

COMMITTEE DRAFT		Reference document: SC 27 N18808	
ISO/IEC 2nd CD 18045 (revision)			
Date: 2019-01-04		Supersedes document N18075	
THIS DOCUMENT IS STILL UNDER STUDY AND SUBJECT TO CHANGE. IT SHOULD NOT BE USED FOR REFERENCE PURPOSES.			
ISO/IEC JTC 1/SC 27 Information technology - Security techniques Secretariat: Germany (DIN)	Circulated to P- and O-members, and to technical committees and organizations in liaison for comments by: 2019-03-01 Please submit your comments via the online balloting application by the due date indicated.		
ISO/IEC 2nd CD 18045 (revision)			
Title: IT security techniques — Evaluation criteria for IT security — Methodology for IT security evaluation			
Project: 1.27.36 (ISO/IEC 18045, revision)			
Explanatory Report			
Status	SC 27 Decision	Reference documents	
		Input	Output
ISO/IEC NP 18045	53 rd WG 3 meeting, Oct. 2016, Recommendations 6, 15 (N16800 = WG 5 N600).	Expert contr. (WG 3 N1368, N1371, N1373).	SP report (WG 3 N1363); Call f. editor (WG 3 N1387 = N16886); Liaisons to: CCDB (WG 3 N1330); The Open Group (WG 3 N1332); Text f. NWIP (N16884).
ISO/IEC NP 18045 1st WD	54th WG 3 meeting, April 2017, Recommendations 5,10 11, 14 (N17041 = WG 3 N1413).	SoV (N17030).	Liaisons to: CCDB (WG 3 N1391); The Open Group (WG 3 N1394); ISO/TC 22/SC 32 (N17373); Text f. 1st WD (WG 3 N1440).
ISO/IEC 18045 2nd WD	55th WG 3 meeting, October / November 2017, Recommendations 8, 10, 15 (N17666 = WG 3 N1494).	SoCom (WG 3 N1476); Draft DoC (WG 3 N1501).	Liaison to ISO/TC 22/SC 32/WG 11 (N18103); Status (WG 3 N1465); DoC (WG 3 N1462); Text f. 2 nd WD (WG 3 N1478).
ISO/IEC 18045 1st CD	56th WG 3 meeting, April 2018, Recommendations 10, 12 / 30 th SC 27 Plenary, April 2018, Resolution 6 (N18710) (N18471 = WG 3 N1557).	SoCom (WG 3 N1536); Late Com (WG 3 N1567); Draft DoC (WG 3 N15).	DoC (WG 3 N1527); Text f. 1 st CD (N18705).
ISO/IEC 18045 2nd CD	57th WG 3 meeting / CRM for WG 3 projects, Sep / Oct 2018, Recommendations 11, 14, 15 (N18820 = WG 3 N1610).	SoV (N18860).	Liaison to CCDB (WG 3 N1619); DoC (N18802); Text f. 2 nd CD (N18808).
2nd CD Consideration			
In accordance with Recommendation 14 (see SC 27 N18820 = WG 3 N1610) of the 57 th SC 27/WG 3 meeting held in Gjøvik, Norway, 2018-09-30/10-04 the hereby attached document is being circulated for a 8-week 2 nd CD letter ballot closing by			
2019-03-01			
Medium: http://isotc.iso.org/livelink/livelink/open/jtc1sc27			
No. of pages: 2 + 395			

Secretariat, ISO/IEC JTC 1/SC 27 -

DIN Deutsches Institut für Normung e.V., Saatwinkler Damm 42/43, D-13627 [D-10772 postal] Berlin , Germany

Telephone: + 49 2601-2652; Facsimile: + 49 2601-4-2652; E-mail: krvstyna.passia@din.de

<http://www.din.de/go/jtc1sc27>

Explanatory Report (2nd page)			
Status	SC 27 Decision	Reference documents	
		Input	Output
Study Period IT security testing, evaluation and assurance standards and techniques	51 st WG 3 meeting, Oct. 2015, Recommendations 5, 6 (N15594 = WG 3 N1251).		Terms of Reference (WG 5 N1258); 1 st /2 nd call f. contr. (WG 3 N1259 /1317)..
	52 nd WG 3 meeting, April 2016, Recommendation 5, 7 (N16026 = WG 3 N1296).	Expert contr. (WG 3 N1299, 1301).	3 rd call f. contr. (WG 3 N1377); Rapporteur's report (WG 3 N1320); Liaison to: PRIPARE (WG 5 N = N16266).

1
2
3
4
5
6
7
8
9
10
11

ISO/IEC JTC 1/SC 27 N18808

ISO/IEC JTC 1/SC 27/WG 3 N1654

Date: 2018-12-24

ISO/IEC CD 18045:####(EN)

ISO/IEC JTC 1/SC 27 IT Security techniques

Secretariat: DIN

IT security techniques — Evaluation criteria for IT security — Methodology for IT security evaluation

Techniques de sécurité IT — Critères d'évaluation pour a sécurité des technologies de l'information — Méthodologie pour l'évaluation de sécurité TI

CD stage

12
13
14
15
16
17
18

Warning for WDs and CDs

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

19

© ISO 2017, Published in Switzerland

20

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

21

22

23

24

ISO copyright office

25

Ch. de Blandonnet 8 • CP 401

26

CH-1214 Vernier, Geneva, Switzerland

27

Tel. + 41 22 749 01 11

28

Fax + 41 22 749 09 47

29

copyright@iso.org

30

www.iso.org

31

32

Legal Notice:

The text for the legal notice agreed between ISO/IEC and the CCDB will be included here.

33

34	Contents		Page
35	1	Scope	1
36	2	Normative references	1
37	3	Terms and definitions	1
38	4	Symbols and abbreviated terms	2
39	5	Overview	2
40	5.1	Organisation of this International Standard	2
41	6	Document Conventions	2
42	6.1	Terminology	2
43	6.2	Verb usage	2
44	6.3	General evaluation guidance	3
45	6.4	Relationship between ISO/IEC 15408 and ISO/IEC 18045 structures	3
46	7	Evaluation process and related tasks	4
47	7.1	Introduction	4
48	7.2	Evaluation process overview	4
49	7.2.1	Objectives	4
50	7.2.2	Responsibilities of the roles	4
51	7.2.3	Relationship of roles	5
52	7.2.4	General evaluation model	5
53	7.2.5	Evaluator verdicts	5
54	7.3	Evaluation input task	7
55	7.3.1	Objectives	7
56	7.3.2	Application notes	7
57	7.3.3	Management of evaluation evidence sub-task	8
58	7.4	Evaluation sub-activities	8
59	7.5	Evaluation output task	8
60	7.5.1	Objectives	8
61	7.5.2	Management of evaluation outputs	9
62	7.5.3	Application notes	9
63	7.5.4	Write OR sub-task	9
64	7.5.5	Write ETR sub-task	9
65	8	Class APE: Protection Profile evaluation	15
66	8.1	Introduction	15
67	8.2	Application notes	15
68	8.2.1	Re-using the evaluation results of certified PPs	15
69	8.3	PP introduction (APE_INT)	16
70	8.3.1	Evaluation of sub-activity (APE_INT.1)	16
71	8.4	Conformance claims (APE_CCL)	17
72	8.4.1	Evaluation of sub-activity (APE_CCL.1)	17
73	8.5	Security problem definition (APE_SPD)	26
74	8.5.1	Evaluation of sub-activity (APE_SPD.1)	26
75	8.6	Security objectives (APE_OBJ)	28
76	8.6.1	Evaluation of sub-activity (APE_OBJ.1)	28
77	8.6.2	Evaluation of sub-activity (APE_OBJ.2)	28
78	8.7	Extended components definition (APE_ECD)	31
79	8.7.1	Evaluation of sub-activity (APE_ECD.1)	31
80	8.8	Security requirements (APE_REQ)	35
81	8.8.1	Evaluation of sub-activity (APE_REQ.1)	35
82	8.8.2	Evaluation of sub-activity (APE_REQ.2)	41
83	9	Class ACE: Protection Profile Configuration evaluation	46
84	9.1	Introduction	46

85	9.2	PP-Module introduction (ACE_INT)	47
86	9.2.1	Evaluation of sub-activity (ACE_INT.1)	47
87	9.3	PP-Module conformance claims (ACE_CCL)	48
88	9.3.1	Evaluation of sub-activity (ACE_CCL.1)	48
89	9.4	PP-Module Security problem definition (ACE_SPD)	50
90	9.4.1	Evaluation of sub-activity (ACE_SPD.1)	50
91	9.5	PP-Module Security objectives (ACE_OBJ)	50
92	9.5.1	Evaluation of sub-activity (ACE_OBJ.1)	50
93	9.5.2	Evaluation of sub-activity (ACE_OBJ.2)	51
94	9.6	PP-Module extended components definition (ACE_ECD)	51
95	9.6.1	Evaluation of sub-activity (ACE_ECD.1)	51
96	9.7	PP-Module security requirements (ACE_REQ)	51
97	9.7.1	Evaluation of sub-activity (ACE_REQ.1)	51
98	9.7.2	Evaluation of sub-activity (ACE_REQ.1)	51
99	9.8	PP-Module consistency (ACE_MCO)	51
100	9.8.1	Evaluation of sub-activity (ACE_MCO.1)	51
101	9.9	PP-Configuration consistency (ACE_CCO)	53
102	9.9.1	Evaluation of sub-activity (ACE_CCO.1)	53
103	10	Class ASE: Security Target evaluation	56
104	10.1	Introduction	56
105	10.2	Application notes	56
106	10.2.1	Re-using the evaluation results of certified PPs	56
107	10.3	ST introduction (ASE_INT)	57
108	10.3.1	Evaluation of sub-activity (ASE_INT.1)	57
109	10.4	Conformance claims (ASE_CCL)	60
110	10.4.1	Evaluation of sub-activity (ASE_CCL.1)	60
111	10.5	Security problem definition (ASE_SPD)	72
112	10.5.1	Evaluation of sub-activity (ASE_SPD.1)	72
113	10.6	Security objectives (ASE_OBJ)	73
114	10.6.1	Evaluation of sub-activity (ASE_OBJ.1)	73
115	10.6.2	Evaluation of sub-activity (ASE_OBJ.2)	74
116	10.7	Extended components definition (ASE_ECD)	76
117	10.7.1	Evaluation of sub-activity (ASE_ECD.1)	76
118	10.8	Security requirements (ASE_REQ)	80
119	10.8.1	Evaluation of sub-activity (ASE_REQ.1)	80
120	10.8.2	Evaluation of sub-activity (ASE_REQ.2)	86
121	10.9	TOE summary specification (ASE_TSS)	91
122	10.9.1	Evaluation of sub-activity (ASE_TSS.1)	91
123	10.9.2	Evaluation of sub-activity (ASE_TSS.2)	92
124	10.10	Consistency of composite product Security Target (ASE_COMP)	93
125	10.10.1	Evaluation of sub-activity (ASE_COMP.1)	94
126	11	Class ADV: Development	99
127	11.1	Introduction	99
128	11.2	Application notes	99
129	11.3	Security Architecture (ADV_ARC)	100
130	11.3.1	Evaluation of sub-activity (ADV_ARC.1)	100
131	11.4	Functional specification (ADV_FSP)	104
132	11.4.1	Evaluation of sub-activity (ADV_FSP.1)	104
133	11.4.2	Evaluation of sub-activity (ADV_FSP.2)	108
134	11.4.3	Evaluation of sub-activity (ADV_FSP.3)	112
135	11.4.4	Evaluation of sub-activity (ADV_FSP.4)	117
136	11.4.5	Evaluation of sub-activity (ADV_FSP.5)	123
137	11.4.6	Evaluation of sub-activity (ADV_FSP.6)	129
138	11.5	Implementation representation (ADV_IMP)	129
139	11.5.1	Evaluation of sub-activity (ADV_IMP.1)	129
140	11.5.2	Evaluation of sub-activity (ADV_IMP.2)	131
141	11.6	TSF internals (ADV_INT)	134
142	11.6.1	Evaluation of sub-activity (ADV_INT.1)	134
143	11.6.2	Evaluation of sub-activity (ADV_INT.2)	137

144	11.6.3	Evaluation of sub-activity (ADV_INT.3).....	139
145	11.7	Security policy modelling (ADV_SPM)	142
146	11.7.1	Evaluation of sub-activity (ADV_SPM.1)	142
147	11.8	TOE design (ADV_TDS)	146
148	11.8.1	Evaluation of sub-activity (ADV_TDS.1)	146
149	11.8.2	Evaluation of sub-activity (ADV_TDS.2)	150
150	11.8.3	Evaluation of sub-activity (ADV_TDS.3)	155
151	11.8.4	Evaluation of sub-activity (ADV_TDS.4)	164
152	11.8.5	Evaluation of sub-activity (ADV_TDS.5)	174
153	11.8.6	Evaluation of sub-activity (ADV_TDS.6)	181
154	11.9	Composite design compliance (ADV_COMP)	181
155	11.9.1	Evaluation of sub-activity (ADV_COMP.1)	182
156	12	Class AGD: Guidance documents.....	184
157	12.1	Introduction	184
158	12.2	Application notes	184
159	12.3	Operational user guidance (AGD_OPE)	184
160	12.3.1	Evaluation of sub-activity (AGD_OPE.1).....	184
161	12.4	Preparative procedures (AGD_PRE).....	187
162	12.4.1	Evaluation of sub-activity (AGD_PRE.1).....	187
163	13	Class ALC: Life-cycle support	189
164	13.1	Introduction	189
165	13.2	CM capabilities (ALC_CMC).....	190
166	13.2.1	Evaluation of sub-activity (ALC_CMC.1)	190
167	13.2.2	Evaluation of sub-activity (ALC_CMC.2)	191
168	13.2.3	Evaluation of sub-activity (ALC_CMC.3)	192
169	13.2.4	Evaluation of sub-activity (ALC_CMC.4)	196
170	13.2.5	Evaluation of sub-activity (ALC_CMC.5)	202
171	13.3	CM scope (ALC_CMS).....	209
172	13.3.1	Evaluation of sub-activity (ALC_CMS.1)	209
173	13.3.2	Evaluation of sub-activity (ALC_CMS.2)	209
174	13.3.3	Evaluation of sub-activity (ALC_CMS.3)	210
175	13.3.4	Evaluation of sub-activity (ALC_CMS.4)	211
176	13.3.5	Evaluation of sub-activity (ALC_CMS.5)	212
177	13.4	Delivery (ALC_DEL).....	214
178	13.4.1	Evaluation of sub-activity (ALC_DEL.1)	214
179	13.5	Development security (ALC_DVS)	215
180	13.5.1	Evaluation of sub-activity (ALC_DVS.1).....	215
181	13.5.2	Evaluation of sub-activity (ALC_DVS.2).....	218
182	13.6	Flaw remediation (ALC_FLR)	221
183	13.6.1	Evaluation of sub-activity (ALC_FLR.1)	221
184	13.6.2	Evaluation of sub-activity (ALC_FLR.2)	223
185	13.6.3	Evaluation of sub-activity (ALC_FLR.3)	226
186	13.7	Life-cycle definition (ALC_LCD).....	232
187	13.7.1	Evaluation of sub-activity (ALC_LCD.1)	232
188	13.7.2	Evaluation of sub-activity (ALC_LCD.2)	233
189	13.8	TOE Development Artifacts (ALC_TDA)	235
190	13.8.1	Evaluation of sub-activity (ALC_TDA.1)	235
191	13.9	Tools and techniques (ALC_TAT)	235
192	13.9.1	Evaluation of sub-activity (ALC_TAT.1).....	235
193	13.9.2	Evaluation of sub-activity (ALC_TAT.2).....	237
194	13.9.3	Evaluation of sub-activity (ALC_TAT.3).....	239
195	13.10	Integration of composition parts and consistency check of delivery procedures (ALC_COMP)	242
196	13.10.1	Evaluation of sub-activity (ALC_COMP.1)	242
197	14	Class ATE: Tests	243
198	14.1	Introduction	243
199	14.2	Application notes	244
200	14.2.1	Understanding the expected behaviour of the TOE.....	244
201	14.2.2	Testing vs. alternate approaches to verify the expected behaviour of functionality.....	245

202	14.2.3	Verifying the adequacy of tests	245
203	14.3	Coverage (ATE_COV)	246
204	14.3.1	Evaluation of sub-activity (ATE_COV.1)	246
205	14.3.2	Evaluation of sub-activity (ATE_COV.2)	246
206	14.3.3	Evaluation of sub-activity (ATE_COV.3)	248
207	14.4	Depth (ATE_DPT)	250
208	14.4.1	Evaluation of sub-activity (ATE_DPT.1)	250
209	14.4.2	Evaluation of sub-activity (ATE_DPT.2)	252
210	14.4.3	Evaluation of sub-activity (ATE_DPT.3)	255
211	14.4.4	Evaluation of sub-activity (ATE_DPT.4)	257
212	14.5	Functional tests (ATE_FUN)	257
213	14.5.1	Evaluation of sub-activity (ATE_FUN.1)	257
214	14.5.2	Evaluation of sub-activity (ATE_FUN.2)	261
215	14.6	Independent testing (ATE_IND)	264
216	14.6.1	Evaluation of sub-activity (ATE_IND.1)	264
217	14.6.2	Evaluation of sub-activity (ATE_IND.2)	268
218	14.6.3	Evaluation of sub-activity (ATE_IND.3)	274
219	14.7	Composite functional testing (ATE_COMP)	274
220	14.7.1	Evaluation of sub-activity (ATE_COMP.1)	274
221	15	Class AVA: Vulnerability assessment	275
222	15.1	Introduction	275
223	15.1.1	Evaluation of sub-activity (AVA_VAN.1)	275
224	15.1.2	Evaluation of sub-activity (AVA_VAN.2)	281
225	15.1.3	Evaluation of sub-activity (AVA_VAN.3)	287
226	15.1.4	Evaluation of sub-activity (AVA_VAN.4)	296
227	15.1.5	Evaluation of sub-activity (AVA_VAN.5)	303
228	15.2	Composite vulnerability assessment (AVA_COMP)	311
229	15.2.1	Evaluation of sub-activity (AVA_COMP.1)	312
230	16	Class ACO: Composition	313
231	16.1	Introduction	313
232	16.2	Application notes	313
233	16.3	Composition rationale (ACO_COR)	314
234	16.3.1	Evaluation of sub-activity (ACO_COR.1)	314
235	16.4	Development evidence (ACO_DEV)	320
236	16.4.1	Evaluation of sub-activity (ACO_DEV.1)	320
237	16.4.2	Evaluation of sub-activity (ACO_DEV.2)	321
238	16.4.3	Evaluation of sub-activity (ACO_DEV.3)	323
239	16.5	Reliance of dependent component (ACO_REL)	326
240	16.5.1	Evaluation of sub-activity (ACO_REL.1)	326
241	16.5.2	Evaluation of sub-activity (ACO_REL.2)	328
242	16.6	Composed TOE testing (ACO_CTT)	330
243	16.6.1	Evaluation of sub-activity (ACO_CTT.1)	330
244	16.6.2	Evaluation of sub-activity (ACO_CTT.2)	333
245	16.7	Composition vulnerability analysis (ACO_VUL)	336
246	16.7.1	Evaluation of sub-activity (ACO_VUL.1)	336
247	16.7.2	Evaluation of sub-activity (ACO_VUL.2)	339
248	16.7.3	Evaluation of sub-activity (ACO_VUL.3)	343
249	Annex A	(informative) General evaluation guidance	348
250	A.1	Objectives	348
251	A.2	Sampling	348
252	A.3	Dependencies	350
253	A.3.1	Dependencies between activities	350
254	A.3.2	Dependencies between sub-activities	350
255	A.3.3	Dependencies between actions	350
256	A.4	Site Visits	351
257	A.4.1	Introduction	351
258	A.4.2	General Approach	351
259	A.4.3	Orientation Guide for the Preparation of the Check List	352
260	A.4.4	Example of a checklist	354

261	A.5	Scheme Responsibilities	356
262	Annex B	(informative) Vulnerability Assessment (AVA)	358
263	B.1	What is Vulnerability Analysis	358
264	B.2	Evaluator construction of a Vulnerability Analysis	358
265	B.2.1	Generic vulnerability guidance	359
266	B.2.2	Identification of Potential Vulnerabilities	366
267	B.3	When attack potential is used	370
268	B.3.1	Developer	370
269	B.3.2	Evaluator	370
270	B.4	Calculating attack potential	371
271	B.4.1	Application of attack potential	371
272	B.4.2	Characterising attack potential	372
273	B.5	Example calculation for direct attack	378
274	Annex C	Evaluation Techniques and Tools (informative)	380
275	C.1	Semiformal and formal methods	380
276	C.1.1	Description of styles	380
277	C.1.2	Security policy models and styles	384
278			

279 Editor Note

280 Experts in SC27/WG3 agree with the editors that, since this document needs to reflect the evaluation
 281 requirements arising from the various parts of ISO/IEC 15408 the CD for which have only just been completed ,
 282 it is inevitable that the 18045 draft will lag behind the 15408 parts and that some editing will be needed when the
 283 other parts are complete.

284 The aim expressed at WG3 meetings is to have the whole set of documents clearly and comfortably support the co-
 285 existence of the different ways of using the criteria for evaluations. In particular the document set should support
 286 without conflict, contradiction, or interference, ways of providing assurance that accommodate both detailed
 287 specification with transparent, conformance checking (generally the iTC/cPP route), and also the investigative,
 288 judgement-based examination. Evaluations generally combine both approaches to different extents, and different
 289 balances are currently preferred by different groups of users and schemes.

290 This document is intended to meet that aim.

291 Notes for CD1

292 A new element ACE_CCO.1.6C has been added in this version of 18045 (in order to more clearly specify the
 293 requirement for its related work units – previously these were attached to ACE_CCO.1.3C but the connection was
 294 not clear or convincing). This new element therefore needs to be added to 15408-3.

295
 296 Optional SFRs have been removed from 18045 (but will be replaced if discussion on other parts necessitates that
 297 step)

298
 299 **Note**
 300 ISO/IEC 15408-3 CD1 needs to reflect the updated ASE_REQ.1.9C
 301 The security requirements rationale shall demonstrate that the SFRs (in conjunction with the security objectives
 302 for the environment) counter all threats for the TOE.

303
 304 Edits have introduced a new ACE_CCO.1.6C and then renumbered ACE_CCO.1-3a and ACE_CCO.1-3b as ACE_CCO.1-
 305 6 and ACE_CCO.1-7 as its work units. This means that the old ACE_CCO.1-6 is now renumbered as ACE_CCO.1-8.

306
 307 This requires a future update to part 3 to introduce ACE_CCO.1.6C.

308
 309 APE_CCL.1.13C addresses only the identification of allowed PP-modules whereas the related Work Unit
 310 APE_CCL.1-17 covers more, i.e. subject PP's conformance statement / aspect 'allowed with' other base-PPs.

311 ~~~~This seems to be a mismatch, i.e. in APE_CCL.1.13C the goal and content of Work Unit APE_CCL.1-17 is not
312 covered. – expert text awaited

314 Clarification to what a PP may claim conformance. Corresponding update of ISO/IEC 15408-1, ISO/IEC 15408-3
315 and / or ISO/IEC 18045. – deferred to incorporate updates

317 Check as proposed the new subchapters for AVA_VAN.5, ADV_SPM.1, ADV_TDS.5, ADV_IMP.2, ADV_INT.3,
318 ATE_COV.3 and ATE_FUN.2 for consistency to ISO/IEC 15408-1, ISO/IEC 15408-3 and ISO/IEC 18045 (for the
319 latter one check against the other already existing subchapters in AVA, ADV and ATE). Corresponding update of
320 the subchapters where necessary. – expert check awaited (from authors of AIS 34 in particular)

322 Notes for CD2

323 As in comment 003 on CD1 the action elements regarding ASE_COMP, ALC_COMP, ADV_COMP, ATE_COMP,
324 AVA_COMP implemented from Appendix 1.1 of [JIL Composite product evaluation for Smart Cards and similar
325 devices] (version 1.5.1 May 2018) were incorporated. The structure of that appendix was kept broadly the same
326 Certification specific items were omitted. This section needs a close check by relevant experts.

327 There were also a number of changes that could not be made without reference to CD2 versions of other parts and
328 will need to be incorporated in the next round of editing actions. Appropriate detailed comment/changes are
329 invited.

330 ***Deleted: ATE_MTK, ATE_MTT, ADV_MTC_ASE_AMA***

331 ***Added: ACE_OBJ.2, ACE_REQ.2, ASE_COMP, ADV_COMP, ALC_TDA, ALC_COMP, ATE_COMP, AVA_COMP***

332 ***Re-located: ASE_COMP, ADV_COMP, ALC_TDA, ALC_COMP, ATE_COMP, AVA_COMP (to sync up with 15408-3)***

333 ***Need additional contribution or text: ALC_TDA, ALC_COMP, ATE_COMP***

334 ***Resolve the existing cross-reference label issue with TOE design (ADV_TDS), Independent testing (ATE_IND),***
335 ***Depth (ATE_DPT), Class ACO: Composition, TSF internals (ADV_INT)***

336 ***December 24 update***

337 - Item 1) Updated to sync up with the 15408-3 (complete for the part
338 of the class most done but need remaining few chapters such as APE, ACE, ALC,
339 ATE, AVA, ACO)

340 - Item 2) Resolved the existing cross-reference label issue with TOE
341 design (ADV_TDS), Independent testing (ATE_IND), Depth (ATE_DPT),
342 Class ACO: Composition, TSF internals (ADV_INT)

343 - Item 3) Reformatting the *_COMP
344

345 Foreword

346 ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies
347 (ISO member bodies). The work of preparing International Standards is normally carried out through ISO
348 technical committees. Each member body interested in a subject for which a technical committee has been
349 established has the right to be represented on that committee. International organizations, governmental and non-
350 governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International
351 Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

352 The procedures used to develop this document and those intended for its further maintenance are described in
353 the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO
354 documents should be noted. This document was drafted in accordance with the editorial rules of the
355 ISO/IEC Directives, Part 2 (see www.iso.org/directives).

356 Attention is drawn to the possibility that some of the elements of this document may be the subject of patent
357 rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights
358 identified during the development of the document will be in the Introduction and/or on the ISO list of patent
359 declarations received (see www.iso.org/patents).

360 Any trade name used in this document is information given for the convenience of users and does not constitute
361 an endorsement.

362 For an explanation on the voluntary nature of standards, the meaning of ISO specific terms and expressions
363 related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization
364 (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

365 ISO/IEC 18045 was prepared by Joint Technical Committee ISO/IEC JTC 1, Information technology, Subcommittee
366 SC 27, IT Security techniques. The identical text of ISO/IEC 15408 is published by the Common Criteria Project
367 Sponsoring Organisations as Common Criteria for Information Technology Security Evaluation. The common XML
368 source for both publications can be found at <http://www.commoncriteriaportal.org/cc/>

369 This **fourth** edition cancels and replaces the **third** edition (ISO/IEC 18045:-2009), which has been technically
370 revised.

371 The main changes compared to the previous edition are as follows:

- 372 — The document has been revised to comply with ISO/IEC Directives
- 373 — Technical changes have been introduced:
 - 374 ○ New security assurance components have been introduced

375

376

377

378 **Introduction**

379 The target audience for this International Standard is primarily evaluators applying ISO/IEC 15408 and certifiers
380 confirming evaluator actions; evaluation sponsors, developers, PP/ST authors and other parties interested in IT
381 security may be a secondary audience.

382 This International Standard recognises that not all questions concerning IT security evaluation will be answered
383 herein and that further interpretations will be needed. Individual schemes will determine how to handle such
384 interpretations, although these may be subject to mutual recognition agreements. A list of methodology-related
385 activities that may be handled by individual schemes can be found in Annex A.

386 **IT security techniques — Evaluation criteria for IT**
 387 **securityInformation technology — Security techniques —**
 388 **Methodology for IT security evaluation**

389 **1 Scope**

390 This International Standard is a companion document to the “Evaluation criteria for IT security”,
 391 ISO/IEC 15408. This International Standard defines the minimum actions to be performed by an
 392 evaluator in order to conduct an ISO/IEC 15408 evaluation, using the criteria and evaluation
 393 evidence defined in ISO/IEC 15408.

394 This International Standard defines evaluator actions for ISO/IEC 15408 components where there
 395 is agreed guidance. Evaluation activities defined using ISO/IEC 15408-4 may be used in place of
 396 work units within this document provided that this is made clear within the Security Target or
 397 Protection Profile, and the evaluation and certification reports.

398 **2 Normative references**

399 The following referenced documents are indispensable for the application of this document. For
 400 dated references, only the edition cited applies. For undated references, the latest edition of the
 401 referenced document (including any amendments) applies.

402 ISO/IEC 15408-1:20XX, *IT Security techniques — Evaluation criteria for IT security — Part 1:*
 403 *Introduction and general model*

404 ISO/IEC 15408-2:20XX, *IT Security techniques — Evaluation criteria for IT security — Part 2:*
 405 *Security functional components*

406 ISO/IEC 15408-3:20XX, *IT Security techniques — Evaluation criteria for IT security — Part 3:*
 407 *Security assurance components*

408 ISO/IEC 15408-4:20XX, *IT Security techniques — Evaluation criteria for IT security — Part 4:*
 409 *Framework for the specification of evaluation methods and activities*

410 ISO/IEC 15408-5:20XX, *IT Security techniques — Evaluation criteria for IT security — Part 5: Pre-*
 411 *defined packages of security requirements*

412 **3 Terms and definitions**

413 For the purposes of this document, the terms, definitions, symbols and abbreviated terms given in
 414 ISO/IEC 15408-1 apply.

415 ISO and IEC maintain terminological databases for use in standardization at the following
 416 addresses:

417 — IEC Electropedia: available at <http://www.electropedia.org/>

418 — ISO Online browsing platform: available at <http://www.iso.org/obp>

419 **Terms and definitions previously located in this clause of 18045 are now found in ISO/IEC 15408**

420 **4 Symbols and abbreviated terms**

421 Symbols and abbreviations have been moved to part 1

422 **5 Overview**

423 **5.1 Organisation of this International Standard**

424 Clause 6 defines the conventions used in this International Standard.

425 Clause 7 describes general evaluation tasks with no verdicts associated with them as they do not
426 map to ISO/IEC 15408 evaluator action elements.

427 Clauses 8 to 10 address the work necessary for reaching an evaluation result on a PP.

428 Clauses 10 to 16 define the evaluation activities, organised by Assurance Classes.

429 Annex A covers the basic evaluation techniques used to provide technical evidence of evaluation
430 results.

431 Annex B provides an explanation of the Vulnerability Analysis criteria and examples of their
432 application

433 **6 Document Conventions**

434 **6.1 Terminology**

435 Unlike ISO/IEC 15408, where each element maintains the last digit of its identifying symbol for all
436 components within the family, this International Standard may introduce new work units when an
437 ISO/IEC 15408 evaluator action element changes from sub-activity to sub-activity; as a result, the
438 last digit of the work unit's identifying symbol may change although the work unit remains
439 unchanged.

440 Any methodology-specific evaluation work required that is not derived directly from ISO/IEC
441 15408 requirements is termed *task* or *sub-task*.

442 **6.2 Verb usage**

443 All work unit and sub-task verbs are preceded by the auxiliary verb *shall* and by presenting both
444 the verb and the *shall* in ***bold italic*** type face. The auxiliary verb *shall* is used only when the
445 provided text is mandatory and therefore only within the work units and sub-tasks. The work units
446 and sub-tasks contain mandatory activities that the evaluator must perform in order to assign
447 verdicts.

448 Guidance text accompanying work units and sub-tasks gives further explanation on how to apply
449 ISO/IEC 15408 words in an evaluation. The verb usage is in accordance with ISO definitions for
450 these verbs. The auxiliary verb *should* is used when the described method is strongly preferred. All
451 other auxiliary verbs, including *may*, are used where the described method(s) is allowed but is
452 neither recommended nor strongly preferred; it is merely explanation.

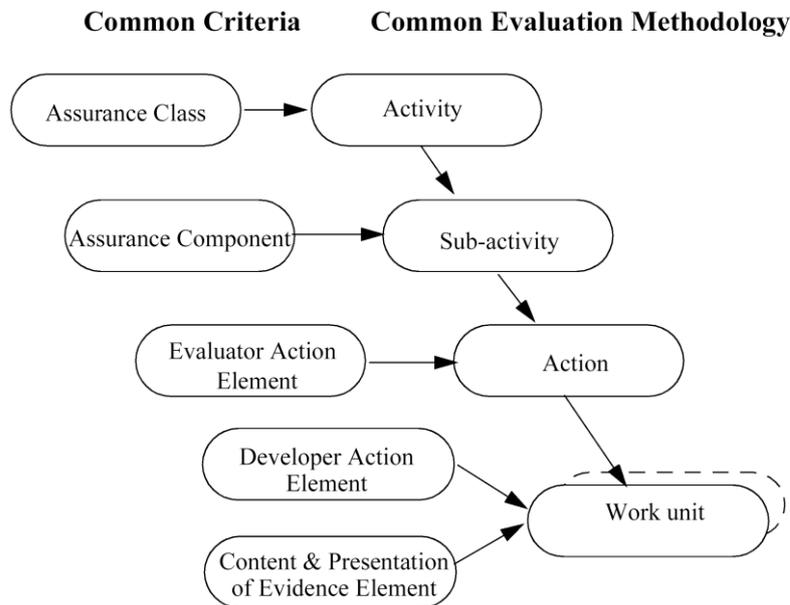
453 The verbs *check*, *examine*, *report* and *record* are used with a precise meaning within this part of this
454 International Standard and the Clause 3 should be referenced for their definitions.

455 **6.3 General evaluation guidance**

456 Material that has applicability to more than one sub-activity is collected in one place. Guidance
 457 whose applicability is widespread (across activities and EALs) has been collected into Annex A.
 458 Guidance that pertains to multiple sub-activities within a single activity has been provided in the
 459 introduction to that activity. If guidance pertains to only a single sub-activity, it is presented within
 460 that sub-activity.

461 **6.4 Relationship between ISO/IEC 15408 and ISO/IEC 18045 structures**

462 There are direct relationships between ISO/IEC 15408 structure (i.e. class, family, component and
 463 element) and the structure of this International Standard. Figure 1 illustrates the correspondence
 464 between ISO/IEC 15408 constructs of class, family and evaluator action elements and evaluation
 465 methodology activities, sub-activities and actions. However, several evaluation methodology work
 466 units may result from the requirements noted in ISO/IEC 15408 developer action and content and
 467 presentation elements. Evaluation activities defined in conformance with part 4 of ISO/IEC15408
 468 may be used in place of work units within this document provided that this is made clear within
 469 the evaluation and certification reports



470

471 **Figure 1 — Mapping of ISO/IEC 15408 and ISO/IEC 18045 structures**

472 **7 Evaluation process and related tasks**

473 **7.1 Introduction**

474 This clause provides an overview of the evaluation process and defines the tasks an evaluator is
475 intended to perform when conducting an evaluation.

476 Each evaluation, whether of a PP or TOE (including ST), follows the same process, and has four
477 evaluator tasks in common: the input task, the output task, the evaluation sub-activities, and the
478 demonstration of the technical competence to the evaluation authority task.

479 The input task and the output tasks, which are related to management of evaluation evidence and
480 to report generation, are entirely described in this clause. Each task has associated sub-tasks that
481 apply to, and are normative for all ISO/IEC 15408 evaluations (evaluation of a PP or a TOE).

482 The evaluation sub-activities are only introduced in this clause, and fully described in the following
483 clauses.

484 In contrast to the evaluation sub-activities, input and output tasks have no verdicts associated with
485 them as they do not map to ISO/IEC 15408 evaluator action elements; they are performed in order
486 to ensure conformance with the universal principles and to comply with this International
487 Standard.

488 The demonstration of the technical competence to the evaluation authority task may be fulfilled by
489 the evaluation authority analysis of the output tasks results, or may include the demonstration by
490 the evaluators of their understanding of the inputs for the evaluation sub-activities. This task has
491 no associated evaluator verdict, but has an evaluator authority verdict. The detailed criteria to pass
492 this task are left to the discretion of the evaluation authority, as noted in Annex A.5.

493 **7.2 Evaluation process overview**

494 **7.2.1 Objectives**

495 This subclause presents the general model of the methodology and identifies:

- 496 a) roles and responsibilities of the parties involved in the evaluation process;
- 497 b) the general evaluation model.

498 **7.2.2 Responsibilities of the roles**

499 The general model defines the following roles: sponsor, developer, evaluator and evaluation
500 authority.

501 The sponsor is responsible for requesting and supporting an evaluation. This means that the
502 sponsor establishes the different agreements for the evaluation (e.g. commissioning the
503 evaluation). Moreover, the sponsor is responsible for ensuring that the evaluator is provided with
504 the evaluation evidence.

505 The developer produces the TOE and is responsible for providing the evidence required for the
506 evaluation (e.g. training, design information), on behalf of the sponsor.

507 The evaluator performs the evaluation tasks required in the context of an evaluation: the evaluator
508 receives the evaluation evidence from the developer on behalf of the sponsor or directly from the
509 sponsor, performs the evaluation sub-activities and provides the results of the evaluation
510 assessment to the evaluation authority.

511 The evaluation authority establishes and maintains the scheme, monitors the evaluation conducted
 512 by the evaluator, and issues certification/validation reports as well as certificates based on the
 513 evaluation results provided by the evaluator.

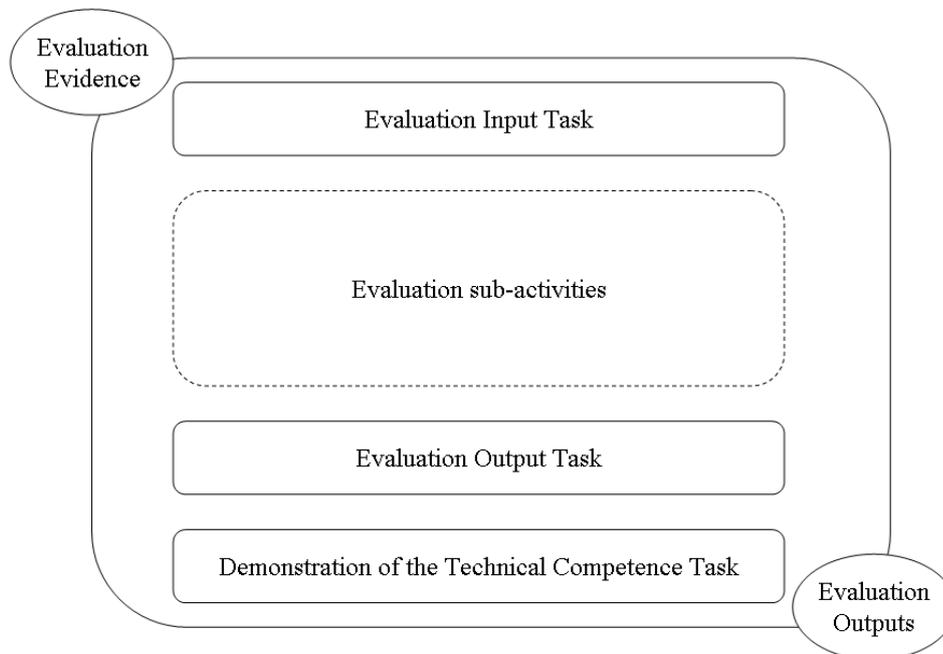
514 7.2.3 Relationship of roles

515 To prevent undue influence from improperly affecting an evaluation, some separation of roles is
 516 required. This implies that the roles described above are fulfilled by different entities, except that
 517 the roles of developer and sponsor may be satisfied by a single entity.

518 Moreover, some evaluations (e.g. EAL1 evaluation) may not require the developer to be involved in
 519 the project. In this case, it is the sponsor who provides the TOE to the evaluator and who generates
 520 the evaluation evidence.

521 7.2.4 General evaluation model

522 The evaluation process consists of the evaluator performing the evaluation input task, the
 523 evaluation output task and the evaluation sub-activities. Figure 2 provides an overview of the
 524 relationship between these tasks and sub-activities.



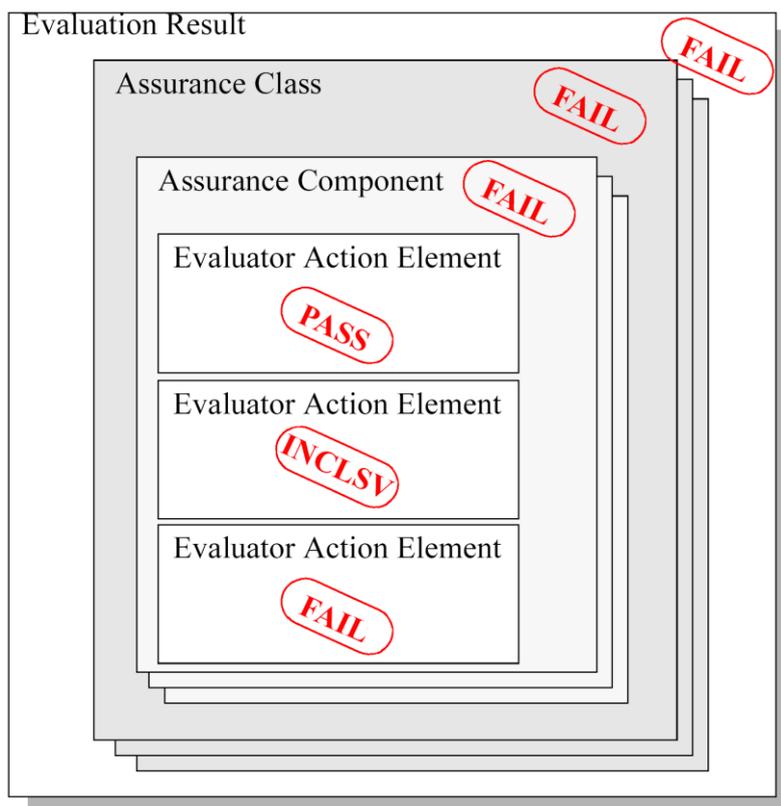
525

526 Figure 2 — Generic evaluation model

527 The evaluation process may be preceded by a preparation phase where initial contact is made
 528 between the sponsor and the evaluator. The work that is performed and the involvement of the
 529 different roles during this phase may vary. It is typically during this step that the evaluator
 530 performs a feasibility analysis to assess the likelihood of a successful evaluation.

531 7.2.5 Evaluator verdicts

532 The evaluator assigns verdicts to the requirements of ISO/IEC 15408 and not to those of this
 533 International Standard. The most granular ISO/IEC 15408 structure to which a verdict is assigned
 534 is the evaluator action element (explicit or implied). A verdict is assigned to an applicable ISO/IEC
 535 15408 evaluator action element as a result of performing the corresponding evaluation
 536 methodology action and its constituent work units. Finally, an evaluation result is assigned, as
 537 described in ISO/IEC 15408-1, Clause 9, Evaluation results.



538

539 **Figure 3 — Example of the verdict assignment rule**

540 This International Standard recognises three mutually exclusive verdict states:

- 541 a) Conditions for a *pass* verdict are defined as an evaluator completion of ISO/IEC 15408
 542 evaluator action element and determination that the requirements for the PP, ST or TOE
 543 under evaluation are met. The conditions for passing the element are defined as:
- 544 1) the constituent work units of the related evaluation methodology action, and;
 - 545 2) all evaluation evidence required for performing these work units is coherent, that is it can
 546 be fully and completely understood by the evaluator, and
 - 547 3) all evaluation evidence required for performing these work units does not have any
 548 obvious internal inconsistencies or inconsistencies with other evaluation evidence. Note
 549 that obvious means here that the evaluator discovers this inconsistency while performing
 550 the work units: the evaluator should not undertake a full consistency analysis across the
 551 entire evaluation evidence every time a work unit is performed.
- 552 b) Conditions for a *fail* verdict are defined as an evaluator completion of ISO/IEC 15408
 553 evaluator action element and determination that the requirements for the PP, ST, or TOE
 554 under evaluation are not met, or that the evidence is incoherent, or an obvious
 555 inconsistency in the evaluation evidence has been found;
- 556 c) All verdicts are initially *inconclusive* and remain so until either a *pass* or *fail* verdict is
 557 assigned.

558 The overall verdict is *pass* if and only if all the constituent verdicts are also *pass*. In the example
 559 illustrated in Figure 3, if the verdict for one evaluator action element is *fail* then the verdicts for the
 560 corresponding assurance component, assurance class, and overall verdict are also *fail*.

561 7.3 Evaluation input task

562 7.3.1 Objectives

563 The objective of this task is to ensure that the evaluator has available the correct version of the
564 evaluation evidence necessary for the evaluation and that it is adequately protected. Otherwise, the
565 technical accuracy of the evaluation cannot be assured, nor can it be assured that the evaluation is
566 being conducted in a way to provide repeatable and reproducible results.

567 7.3.2 Application notes

568 The responsibility to provide all the required evaluation evidence lies with the sponsor. However,
569 most of the evaluation evidence is likely to be produced and supplied by the developer, on behalf of
570 the sponsor.

571 Since the assurance requirements apply to the entire TOE, all evaluation evidence pertaining to all
572 parts of the TOE is to be made available to the evaluator. The scope and required content of such
573 evaluation evidence is independent of the level of control that the developer has over each of the
574 parts of the TOE. For example, if design is required, then the TOE design (ADV_TDS) requirements
575 will apply to all subsystems that are part of the TSF. In addition, assurance requirements that call
576 for procedures to be in place (for example, CM capabilities (ALC_CMC) and Delivery (ALC_DEL))
577 will also apply to the entire TOE (including any part produced by another developer).

578 It is recommended that the evaluator, in conjunction with the sponsor, produce an index to
579 required evaluation evidence. This index may be a set of references to the documentation. This
580 index should contain enough information (e.g. a brief summary of each document, or at least an
581 explicit title, indication of the subclauses of interest) to help the evaluator to find easily the
582 required evidence.

583 It is the information contained in the evaluation evidence that is required, not any particular
584 document structure. Evaluation evidence for a sub-activity may be provided by separate
585 documents, or a single document may satisfy several of the input requirements of a sub-activity.

586 The evaluator requires stable and formally-issued versions of evaluation evidence. However, draft
587 evaluation evidence may be provided during an evaluation, for example, to help an evaluator make
588 an early, informal assessment, but is not used as the basis for verdicts. It may be helpful for the
589 evaluator to see draft versions of particular appropriate evaluation evidence, such as:

- 590 a) test documentation, to allow the evaluator to make an early assessment of tests and test
591 procedures;
- 592 b) design documents, to provide the evaluator with background for understanding the TOE
593 design;
- 594 c) source code or hardware drawings, to allow the evaluator to assess the application of the
595 developer's standards.

596 Draft evaluation evidence is more likely to be encountered where the evaluation of a TOE is
597 performed concurrently with its development. However, it may also be encountered during the
598 evaluation of an already-developed TOE where the developer has had to perform additional work
599 to address a problem identified by the evaluator (e.g. to correct an error in design or
600 implementation) or to provide evaluation evidence of security that is not provided in the existing
601 documentation (e.g. in the case of a TOE not originally developed to meet the requirements of
602 ISO/IEC 15408).

603 **7.3.3 Management of evaluation evidence sub-task**

604 **7.3.3.1 Configuration control**

605 The evaluator *shall perform* configuration control of the evaluation evidence.

606 ISO/IEC 15408 implies that the evaluator is able to identify and locate each item of evaluation
607 evidence after it has been received and is able to determine whether a specific version of a
608 document is in the evaluator's possession.

609 The evaluator *shall protect* the evaluation evidence from alteration or loss while it is in the
610 evaluator's possession.

611 **7.3.3.2 Disposal**

612 Schemes may wish to control the disposal of evaluation evidence at the conclusion of an evaluation.
613 The disposal of the evaluation evidence should be achieved by one or more of:

- 614 a) returning the evaluation evidence;
- 615 b) archiving the evaluation evidence;
- 616 c) destroying the evaluation evidence.

617 **7.3.3.3 Confidentiality**

618 An evaluator may have access to sponsor and developer commercially-sensitive information (e.g.
619 TOE design information, specialist tools), and may have access to nationally-sensitive information
620 during the course of an evaluation. Schemes may wish to impose requirements for the evaluator to
621 maintain the confidentiality of the evaluation evidence. The sponsor and evaluator may mutually
622 agree to additional requirements as long as these are consistent with the scheme.

623 Confidentiality requirements affect many aspects of evaluation work, including the receipt,
624 handling, storage and disposal of evaluation evidence.

625 **7.4 Evaluation sub-activities**

626 The evaluation sub-activities vary depending on whether it is a PP or a TOE evaluation. Moreover,
627 in the case of a TOE evaluation, the sub-activities depend upon the selected assurance
628 requirements.

629 **7.5 Evaluation output task**

630 **7.5.1 Objectives**

631 The objective of this subclause is to describe the Observation Report (OR) and the Evaluation
632 Technical Report (ETR). Schemes may require additional evaluator reports such as reports on
633 individual units of work, or may require additional information to be contained in the OR and the
634 ETR. This International Standard does not preclude the addition of information into these reports
635 as this International Standard specifies only the minimum information content.

636 Consistent reporting of evaluation results facilitates the achievement of the universal principle of
637 repeatability and reproducibility of results. The consistency covers the type and the amount of
638 information reported in the ETR and OR. The consistency of ETR and OR among different
639 evaluations is the responsibility of the evaluation authority.

640 The evaluator performs the two following sub-tasks in order to meet the requirements of this
641 International Standard for the information content of reports:

642 a) write OR sub-task (if needed in the context of the evaluation);

643 b) write ETR sub-task.

644 7.5.2 Management of evaluation outputs

645 The evaluator delivers the ETR to the evaluation authority, as well as any ORs as they become
646 available. Requirements for controls on handling the ETR and ORs are established by the scheme
647 which may include delivery to the sponsor or developer. The ETR and ORs may include sensitive or
648 proprietary information and may need to be sanitised before they are given to the sponsor.

649 7.5.3 Application notes

650 In this version of this International Standard, the requirements for the provision of evaluator
651 evidence to support re-evaluation and re-use have not been explicitly stated. Where information
652 for re-evaluation or re-use is required by the sponsor, the scheme under which the evaluation is
653 being performed should be consulted.

654 7.5.4 Write OR sub-task

655 ORs provide the evaluator with a mechanism to request a clarification (e.g. from the evaluation
656 authority on the application of a requirement) or to identify a problem with an aspect of the
657 evaluation.

658 In the case of a fail verdict, the evaluator **shall provide** an OR to reflect the evaluation result.
659 Otherwise, the evaluator may use ORs as one way of expressing clarification needs.

660 For each OR, the evaluator **shall report** the following:

661 a) the identifier of the PP or TOE evaluated;

662 b) the evaluation task/sub-activity during which the observation was generated;

663 c) the observation;

664 d) the assessment of its severity (e.g. implies a fail verdict, holds up progress on the
665 evaluation, requires a resolution prior to evaluation being completed);

666 e) the identification of the organisation responsible for resolving the issue;

667 f) the recommended timetable for resolution;

668 g) the assessment of the impact on the evaluation of failure to resolve the observation.

669 The intended audience of an OR and procedures for handling the report depend on the nature of
670 the report's content and on the scheme. Schemes may distinguish different types of ORs or define
671 additional types, with associated differences in required information and distribution (e.g.
672 evaluation ORs to evaluation authorities and sponsors).

673 7.5.5 Write ETR sub-task

674 7.5.5.1 Objectives

675 The evaluator **shall provide** an ETR to present technical justification of the verdicts.

676 This International Standard defines the ETR's minimum content requirement; however, schemes
677 may specify additional content and specific presentational and structural requirements. For

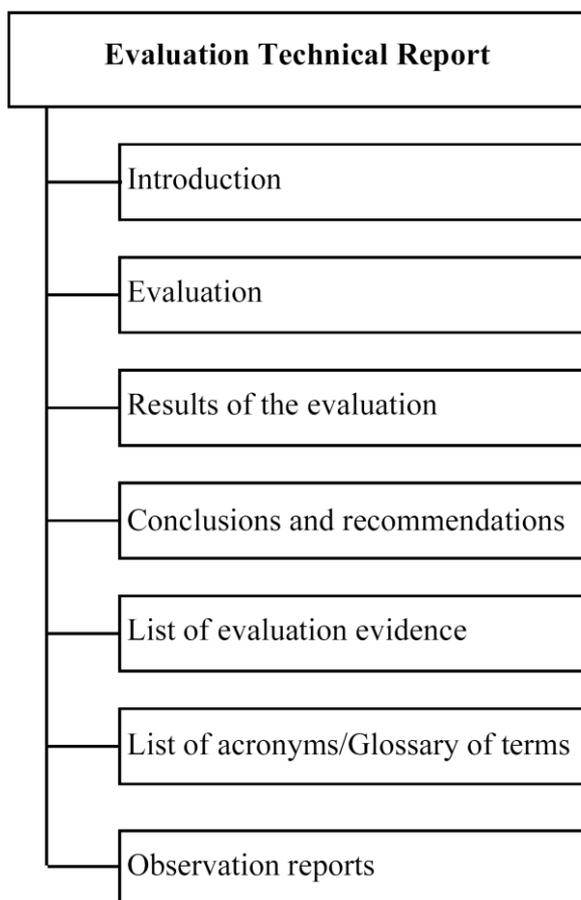
678 instance, schemes may require that certain introductory material (e.g. disclaimers and copyright
679 Clauses) be reported in the ETR.

680 The reader of the ETR is assumed to be familiar with general concepts of information security,
681 ISO/IEC 15408, this International Standard, evaluation approaches and IT.

682 The ETR supports the evaluation authority to confirm that the evaluation was done to the required
683 standard, but it is anticipated that the documented results may not provide all of the necessary
684 information, so additional information specifically requested by the scheme may be necessary. This
685 aspect is outside the scope of this International Standard.

686 **7.5.5.2 ETR for a PP Evaluation**

687 This Subclause describes the minimum content of the ETR for a PP evaluation. The contents of the
688 ETR are portrayed in Figure 4; this figure may be used as a guide when constructing the structural
689 outline of the ETR document.



690

691 **Figure 4 —ETR information content for a PP evaluation**

692

692 **7.5.5.2.1 Introduction**

693 The evaluator **shall report** evaluation scheme identifiers.

694 Evaluation scheme identifiers (e.g. logos) are the information required to unambiguously identify
695 the scheme responsible for the evaluation oversight.

696 The evaluator **shall report** ETR configuration control identifiers.

- 697 The ETR configuration control identifiers contain information that identifies the ETR (e.g. name,
698 date and version number).
- 699 The evaluator **shall report** PP configuration control identifiers.
- 700 PP configuration control identifiers (e.g. name, date and version number) are required to identify
701 what is being evaluated in order for the evaluation authority to verify that the verdicts have been
702 assigned correctly by the evaluator.
- 703 The evaluator **shall report** the identity of the developer.
- 704 The identity of the PP developer is required to identify the party responsible for producing the PP.
- 705 The evaluator **shall report** the identity of the sponsor.
- 706 The identity of the sponsor is required to identify the party responsible for providing evaluation
707 evidence to the evaluator.
- 708 The evaluator **shall report** the identity of the evaluator.
- 709 The identity of the evaluator is required to identify the party performing the evaluation and
710 responsible for the evaluation verdicts.
- 711 **7.5.5.2.2 Evaluation**
- 712 The evaluator **shall report** the evaluation methods, techniques, tools and standards used.
- 713 The evaluator references the evaluation criteria, methodology and interpretations used to evaluate
714 the PP.
- 715 The evaluator **shall report** any constraints on the evaluation, constraints on the handling of
716 evaluation results and assumptions made during the evaluation that have an impact on the
717 evaluation results.
- 718 The evaluator may include information in relation to legal or statutory aspects, organisation,
719 confidentiality, etc.
- 720 **7.5.5.2.3 Results of the evaluation**
- 721 The evaluator **shall report** a verdict and a supporting rationale for each assurance component that
722 constitutes an APE activity, as a result of performing the corresponding evaluation methodology
723 action and its constituent work units.
- 724 The rationale justifies the verdict using ISO/IEC 15408, this International Standard, any
725 interpretations and the evaluation evidence examined and shows how the evaluation evidence
726 does or does not meet each aspect of the criteria. It contains a description of the work performed,
727 the method used, and any derivation of results. The rationale may provide detail to the level of an
728 evaluation methodology work unit.
- 729 **7.5.5.2.4 Conclusions and recommendations**
- 730 The evaluator **shall report** the conclusions of the evaluation, in particular the overall verdict as
731 defined in ISO/IEC 15408-1 Clause 12, Evaluation results, and determined by application of the
732 verdict assignment described in 7.2.5.
- 733 The evaluator provides recommendations that may be useful for the evaluation authority. These
734 recommendations may include shortcomings of the PP discovered during the evaluation or
735 mention of features which are particularly useful.

736 **7.5.5.2.5 List of evaluation evidence**

737 The evaluator *shall report* for each item of evaluation evidence the following information:

738 — the issuing body (e.g. the developer, the sponsor);

739 — the title;

740 — the unique reference (e.g. issue date and version number).

741 **7.5.5.2.6 List of acronyms/Glossary of terms**

742 The evaluator *shall report* any acronyms or abbreviations used in the ETR.

743 Glossary definitions already defined by ISO/IEC 15408 or by this International Standard need not
744 be repeated in the ETR.

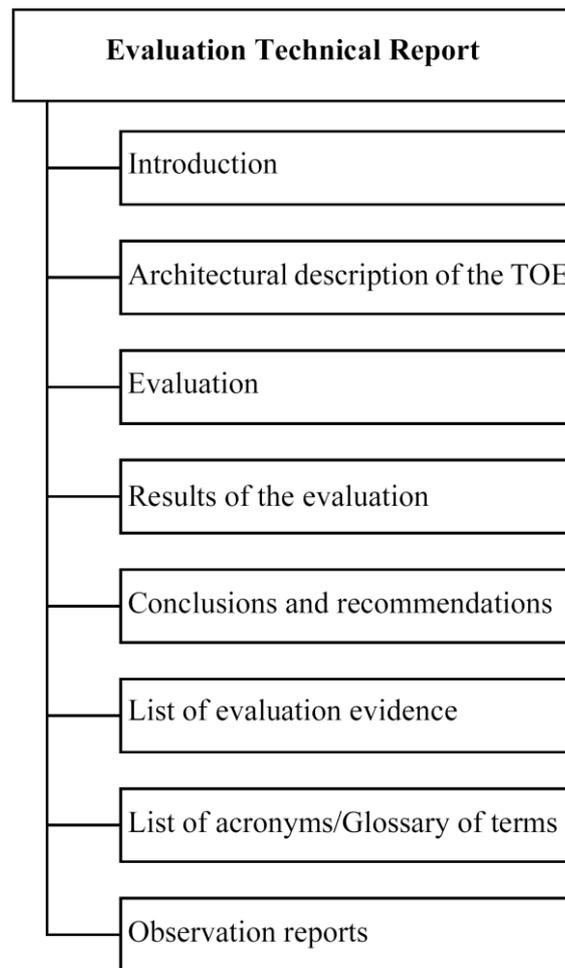
745 **7.5.5.2.7 Observation reports**

746 The evaluator *shall report* a complete list that uniquely identifies the ORs raised during the
747 evaluation and their status.

748 For each OR, the list should contain its identifier as well as its title or a brief summary of its content.

749 **7.5.5.3 ETR for a TOE Evaluation**

750 This Subclause describes the minimum content of the ETR for a TOE evaluation. The contents of the
751 ETR are portrayed in Figure 5; this figure may be used as a guide when constructing the structural
752 outline of the ETR document.



753

754

Figure 5 — ETR information content for a TOE evaluation

755

7.5.5.3.1 Introduction

756

The evaluator **shall report** evaluation scheme identifiers.

757

Evaluation scheme identifiers (e.g. logos) are the information required to unambiguously identify the scheme responsible for the evaluation oversight.

758

759

The evaluator **shall report** ETR configuration control identifiers.

760

The ETR configuration control identifiers contain information that identifies the ETR (e.g. name, date and version number).

761

762

The evaluator **shall report** ST and TOE configuration control identifiers.

763

ST and TOE configuration control identifiers identify what is being evaluated in order for the evaluation authority to verify that the verdicts have been assigned correctly by the evaluator.

764

765

If the ST claims that the TOE conforms to the requirements of one or more PPs, the ETR shall report the reference of the corresponding PPs.

766

767

The PPs reference contains information that uniquely identifies the PPs (e.g. title, date, and version number).

768

- 769 The evaluator **shall report** the identity of the developer.
- 770 The identity of the TOE developer is required to identify the party responsible for producing the
771 TOE.
- 772 The evaluator **shall report** the identity of the sponsor.
- 773 The identity of the sponsor is required to identify the party responsible for providing evaluation
774 evidence to the evaluator.
- 775 The evaluator **shall report** the identity of the evaluator.
- 776 The identity of the evaluator is required to identify the party performing the evaluation and
777 responsible for the evaluation verdicts.
- 778 **7.5.5.3.2 Architectural description of the TOE**
- 779 The evaluator **shall report** a high level description of the TOE and its major components based on
780 the evaluation evidence described in ISO/IEC 15408 assurance family entitled TOE design
781 (ADV_TDS), where applicable.
- 782 The intent of this Subclause is to characterise the degree of architectural separation of the major
783 components. If there is no TOE design (ADV_TDS) requirement in the ST, this is not applicable and
784 is considered to be satisfied.
- 785 **7.5.5.3.3 Evaluation**
- 786 The evaluator **shall report** the evaluation methods, techniques, tools and standards used.
- 787 The evaluator may reference the evaluation criteria, methodology and interpretations used to
788 evaluate the TOE or the devices used to perform the tests.
- 789 The evaluator **shall report** any constraints on the evaluation, constraints on the distribution of
790 evaluation results and assumptions made during the evaluation that have an impact on the
791 evaluation results.
- 792 The evaluator may include information in relation to legal or statutory aspects, organisation,
793 confidentiality, etc.
- 794 **7.5.5.3.4 Results of the evaluation**
- 795 For each activity on which the TOE is evaluated, the evaluator **shall report**:
- 796 — the title of the activity considered;
- 797 — a verdict and a supporting rationale for each assurance component that constitutes this
798 activity, as a result of performing the corresponding evaluation methodology action and its
799 constituent work units.
- 800 The rationale justifies the verdict using ISO/IEC 15408, this International Standard, any
801 interpretations and the evaluation evidence examined and shows how the evaluation evidence
802 does or does not meet each aspect of the criteria. It contains a description of the work performed,
803 the method used, and any derivation of results. The rationale may provide detail to the level of an
804 evaluation methodology work unit.
- 805 The evaluator **shall report** all information specifically required by a work unit.

806 For the AVA and ATE activities, work units that identify information to be reported in the ETR have
807 been defined.

808 **7.5.5.3.5 Conclusions and recommendations**

809 The evaluator *shall report* the conclusions of the evaluation, which will relate to whether the TOE
810 has satisfied its associated ST, in particular the overall verdict as defined in ISO/IEC 15408-1
811 Clause 9, **Evaluation results**, and determined by application of the verdict assignment described in
812 7.2.5.

813 The evaluator provides recommendations that may be useful for the evaluation authority. These
814 recommendations may include shortcomings of the IT product discovered during the evaluation or
815 mention of features which are particularly useful.

816 **7.5.5.3.6 List of evaluation evidence**

817 The evaluator *shall report* for each item of evaluation evidence the following information:

818 — the issuing body (e.g. the developer, the sponsor);

819 — the title;

820 — the unique reference (e.g. issue date and version number).

821 **7.5.5.3.7 List of acronyms/Glossary of terms**

822 The evaluator *shall report* any acronyms or abbreviations used in the ETR.

823 Glossary definitions already defined by ISO/IEC 15408 or by this International Standard need not
824 be repeated in the ETR.

825 **7.5.5.3.8 Observation reports**

826 The evaluator *shall report* a complete list that uniquely identifies the ORs raised during the
827 evaluation and their status.

828 For each OR, the list should contain its identifier as well as its title or a brief summary of its content.

829 **8 Class APE: Protection Profile evaluation**

830 **8.1 Introduction**

831 This Clause describes the evaluation of a PP. The requirements and methodology for PP evaluation
832 are identical for each PP evaluation, regardless of the EAL (or other set of assurance requirements)
833 that is claimed in the PP. The evaluation methodology in this Clause is based on the requirements
834 on the PP as specified in ISO/IEC 15408-3 class APE.

835 This Clause should be used in conjunction with Annexes **A**, **B** and **C, Guidance for Operations** in
836 ISO/IEC 15408-1, as these Annexes clarify the concepts here and provide many examples.

837 **8.2 Application notes**

838 **8.2.1 Re-using the evaluation results of certified PPs**

839 While evaluating a PP that is based on one or more certified PPs, it may be possible to re-use the
840 fact that these PPs were certified. The potential for re-use of the result of a certified PP is greater if
841 the PP under evaluation does not add threats, OSPs, security objectives and/or security

842 requirements to those of the PP that conformance is being claimed to. If the PP under evaluation
843 contains much more than the certified PP, re-use may not be useful at all.

844 The evaluator is allowed to re-use the PP evaluation results by doing certain analyses only partially
845 or not at all if these analyses or parts thereof were already done as part of the PP evaluation. While
846 doing this, the evaluator should assume that the analyses in the PP were performed correctly.

847 An example would be where the PP that conformance is being claimed to contain a set of security
848 requirements, and these were determined to be internally consistent during its evaluation. If the
849 PP under evaluation uses the exact same requirements, the consistency analysis does not have to
850 be repeated during the PP evaluation. If the PP under evaluation adds one or more requirements,
851 or performs operations on these requirements, the analysis will have to be repeated. However, it
852 may be possible to save work in this consistency analysis by using the fact that the original
853 requirements are internally consistent. If the original requirements are internally consistent, the
854 evaluator only has to determine that:

855 a) the set of all new and/or changed requirements is internally consistent, and

856 b) the set of all new and/or changed requirements is consistent with the original
857 requirements.

858 The evaluator notes in the ETR each case where analyses are not done or only partially done for
859 this reason.

860 **8.3 PP introduction (APE_INT)**

861 **8.3.1 Evaluation of sub-activity (APE_INT.1)**

862 **8.3.1.1 Objectives**

863 The objective of this sub-activity is to determine whether the PP is correctly identified, and
864 whether the PP reference and TOE overview are consistent with each other.

865 **8.3.1.2 Input**

866 The evaluation evidence for this sub-activity is:

867 a) the PP.

868 **8.3.1.3 Action APE_INT.1.1E**

869 ISO/IEC 15408-3 APE_INT.1.1C: *The PP introduction shall contain a PP reference and a TOE*
870 *overview.*

871 **8.3.1.3.1 Work unit APE_INT.1-1**

872 The evaluator **shall check** that the PP introduction contains a PP reference and a TOE overview.

873 ISO/IEC 15408-3 APE_INT.1.2C: *The PP reference shall uniquely identify the PP.*

874 **8.3.1.3.2 Work unit APE_INT.1-2**

875 The evaluator **shall examine** the PP reference to determine that it uniquely identifies the PP.

876 The evaluator determines that the PP reference identifies the PP itself, so that it may be easily
877 distinguished from other PPs, and that it also uniquely identifies each version of the PP, e.g. by
878 including a version number and/or a date of publication.

879 The PP should have some referencing system that is capable of supporting unique references (e.g.
880 use of numbers, letters or dates).

881 ISO/IEC 15408-3 APE_INT.1.3C: *The TOE overview shall summarise the usage and major security*
882 *features of the TOE.*

883 **8.3.1.3.3 Work unit APE_INT.1-3**

884 The evaluator **shall examine** the TOE overview to determine that it describes the usage and major
885 security features of the TOE.

886 The TOE overview should briefly (i.e. several paragraphs) describe the usage and major security
887 features expected of the TOE. The TOE overview should enable consumers and potential TOE
888 developers to quickly determine whether the PP is of interest to them.

889 The evaluator determines that the overview is clear enough for TOE developers and consumers,
890 and sufficient to give them a general understanding of the intended usage and major security
891 features of the TOE.

892 ISO/IEC 15408-3 APE_INT.1.4C: *The TOE overview shall identify the TOE type.*

893 **8.3.1.3.4 Work unit APE_INT.1-4**

894 The evaluator **shall check** that the TOE overview identifies the TOE type.

895 ISO/IEC 15408-3 APE_INT.1.5C: *The TOE overview shall identify any non-TOE*
896 *hardware/software/firmware available to the TOE.*

897 **8.3.1.3.5 Work unit APE_INT.1-5**

898 The evaluator **shall examine** the TOE overview to determine that it identifies any non-TOE
899 hardware/software/firmware available to the TOE.

900 While some TOEs may run stand-alone, other TOEs (notably software TOEs) need additional
901 hardware, software or firmware to operate. In this subclause of the PP, the PP author lists all
902 hardware, software, and/or firmware that will be available for the TOE to run on.

903 This identification should be detailed enough for potential consumers and TOE developers to
904 determine whether their TOE may operate with the listed hardware, software and firmware.

905 **8.4 Conformance claims (APE_CCL)**

906 **8.4.1 Evaluation of sub-activity (APE_CCL.1)**

907 **8.4.1.1 Objectives**

908 The objective of this sub-activity is to determine the validity of various conformance claims. These
909 describe how the PP conforms to ISO/IEC 15408, other PPs and packages.

910 **8.4.1.2 Input**

911 The evaluation evidence for this sub-activity is:

- 912 a) the PP
- 913 b) the content of the PP configuration
- 914 c) the package(s) that the PP claims conformance to.

915 **8.4.1.3 Action APE_CCL.1.1E**

916 ISO/IEC 15408-3 APE_CCL.1.1C: *The conformance claim shall contain an ISO/IEC 15408*
917 *conformance claim that identifies the version of ISO/IEC 15408 to which the PP claims conformance.*

918 **8.4.1.3.1 Work unit APE_CCL.1-1**

919 The evaluator **shall check** that the conformance claim contains an ISO/IEC 15408 conformance
920 claim that identifies the version of ISO/IEC 15408 to which the PP claims conformance.

921 The evaluator determines that ISO/IEC 15408 conformance claim identifies the version of ISO/IEC
922 15408 that was used to develop this PP. This should include the version number of ISO/IEC 15408
923 and, unless the International English version of ISO/IEC 15408 was used, the language of the
924 version of ISO/IEC 15408 that was used.

925 ISO/IEC 15408-3 APE_CCL.1.2C: *ISO/IEC 15408 conformance claim shall describe the conformance of*
926 *the PP to ISO/IEC 15408-2 as either ISO/IEC 15408-2 conformant or ISO/IEC 15408-2 extended.*

927 **8.4.1.3.2 Work unit APE_CCL.1-2**

928 The evaluator **shall check** that ISO/IEC 15408 conformance claim states a claim of either ISO/IEC
929 15408-2 conformant or ISO/IEC 15408-2 extended for the PP.

930 ISO/IEC 15408-3 APE_CCL.1.3C: *ISO/IEC 15408 conformance claim shall describe the conformance of*
931 *the PP to ISO/IEC 15408-3 as either ISO/IEC 15408-3 conformant or ISO/IEC 15408-3 extended.*

932 **8.4.1.3.3 Work unit APE_CCL.1-3**

933 The evaluator **shall check** that ISO/IEC 15408 conformance claim states a claim of either ISO/IEC
934 15408-3 conformant or ISO/IEC 15408-3 extended for the PP.

935 ISO/IEC 15408-3 APE_CCL.1.4C: *ISO/IEC 15408 conformance claim shall be consistent with the*
936 *extended components definition.*

937 **8.4.1.3.4 Work unit APE_CCL.1-4**

938 The evaluator **shall examine** ISO/IEC 15408 conformance claim for ISO/IEC 15408-2 to determine
939 that it is consistent with the extended components definition.

940 If ISO/IEC 15408 conformance claim contains ISO/IEC 15408-2 conformant, the evaluator
941 determines that the extended components definition does not define functional components.

942 If ISO/IEC 15408 conformance claim contains ISO/IEC 15408-2 extended, the evaluator determines
943 that the extended components definition defines at least one extended functional component.

944 **8.4.1.3.5 Work unit APE_CCL.1-5**

945 The evaluator **shall examine** ISO/IEC 15408 conformance claim for ISO/IEC 15408-3 to determine
946 that it is consistent with the extended components definition.

947 If ISO/IEC 15408 conformance claim contains ISO/IEC 15408-3 conformant, the evaluator
948 determines that the extended components definition does not define assurance components.

949 If ISO/IEC 15408 conformance claim contains ISO/IEC 15408-3 extended, the evaluator determines
950 that the extended components definition defines at least one extended assurance component.

951 ISO/IEC 15408-3 APE_CCL.1.5C: *The conformance claim shall identify all PPs and security*
952 *requirement packages to which the PP claims conformance.*

953 **8.4.1.3.6 Work unit APE_CCL.1-6**

954 The evaluator **shall check** that the conformance claim contains a PP claim that identifies all PPs for
955 which the PP claims conformance.

956 If the PP does not claim conformance to another PP, this work unit is not applicable and therefore
957 considered to be satisfied.

958 The evaluator determines that any referenced PPs are unambiguously identified (e.g. by title and
959 version number, or by the identification included in the introduction of that PP).

960 The evaluator is reminded that claims of partial conformance to a PP are not permitted.

961 The evaluator **shall check** that, for each other PP to which the PP being evaluated claims
962 conformance, the conformance statement of that other PP requires strict or demonstrable
963 conformance.

964 If the PP does not claim conformance to another PP, this work unit is not applicable and therefore
965 considered to be satisfied.

966 If the PP requires exact conformance, the evaluator ensures that it does not claim conformance to
967 any other PPs.

968 If the PP claims conformance to another PP, the evaluator ensures that all PPs it is claiming
969 conformance to, require either strict or demonstrable conformance.

970 **8.4.1.3.7 Work unit APE_CCL.1-7**

971 The evaluator shall check, for each identified functional package, that the package definition is
972 complete.

973 If the PP does not claim conformance to a package, this work unit is not applicable and therefore
974 considered to be satisfied.

975 The evaluator determines that the package definition is conformant to the requirements from
976 ISO/IEC 15408-1, clause 8 "Packages" by checking that the functional package includes:

977 a) A functional package identification, giving a unique name, short name, version, date,
978 sponsor, and the ISO/IEC 15408 edition;

979 b) A functional package overview, giving a narrative description of the security functionality;

980 c) A functional package conformance claim, giving the conformance claim to ISO/IEC 15408-
981 2 and ISO/IEC 15408-3;

982 d) If the package defines an SPD then it shall also either

983 i. if using the Direct Rationale approach: include a security functional
984 requirements rationale that maps all threats, OSPs and assumptions in
985 the SPD directly to the SFRs and Security Objectives for the operational
986 environment; or else

987 ii. if not using the Direct Rationale approach: include Security Objectives for
988 the TOE and the operational environment and the Security Objectives
989 rationale;

990 e) The functional package SFRs, and **shall** also include a security requirements rationale if the
991 package includes any Security Objectives for the TOE.

992

993 ISO/IEC 15408-3 APE_CCL.1.6C: *The conformance claim shall describe any conformance of the PP to*
994 *a package as either package-conformant or package-augmented.*

995 **8.4.1.3.8 Work unit APE_CCL.1-8**

996 The evaluator **shall check** that, for each identified package, the conformance claim states a claim of
997 either package-name conformant or package-name augmented.

998 If the PP does not claim conformance to a package, this work unit is not applicable and therefore
999 considered to be satisfied.

1000 If the package conformance claim contains package-name conformant, the evaluator determines
1001 that:

1002 a) If the package is an assurance package, then the PP contains all SARs included in the
1003 package, but no additional SARs.

1004 b) If the package is a functional package, then all assumptions, threats, OSPs, security
1005 objectives and SFRs included in the package are included in identical form in the PP (after
1006 allowing for iteration, refinement, assignments and selections from the package to be
1007 completed as required by the PP).

1008 If the package conformance claim contains package-name augmented, the evaluator determines
1009 that:

1010 a) If the package is an assurance package, then the PP contains all SARs included in the package,
1011 and at least one additional SAR or at least one SAR that is hierarchical to a SAR in the package.

1012 b) If the package is a functional package, then all assumptions, threats, OPSs, Security Objectives,
1013 and SFRs included in the package are included in identical form in the PP (after allowing for
1014 iteration, refinement, assignments and selections from the package to be completed as
1015 required by the PP) except that the PP shall have at least one additional SFR or one SFR that is
1016 hierarchically higher than an SFR in the functional package.

1017 ISO/IEC 15408-3 APE_CCL.1.7C: *The conformance claim rationale shall demonstrate that the TOE*
1018 *type is consistent with the TOE type in the PPs for which conformance is being claimed.*

1019 **8.4.1.3.9 Work unit APE_CCL.1-9**

1020 The evaluator **shall examine** the conformance claim rationale to determine that the TOE type of
1021 the TOE is consistent with all TOE types of the PPs.

1022 If the PP does not claim conformance to another PP, this work unit is not applicable and therefore
1023 considered to be satisfied.

1024 The relation between the types may be simple: a firewall PP claiming conformance to another
1025 firewall PP, or more complex: a smart card PP claiming conformance to a number of other PPs at
1026 the same time: a PP for the integrated circuit, a PP for the smart card OS, and two PPs for two
1027 applications on the smart card.

1028 ISO/IEC 15408-3 APE_CCL.1.8C: *The conformance claim rationale shall demonstrate that the*
1029 *statement of the security problem definition is consistent with the statement of the security problem*
1030 *definition in the PPs for which conformance is being claimed.*

1031 **8.4.1.3.10 Work unit APE_CCL.1-10**

1032 The evaluator **shall examine** the conformance claim rationale to determine that it demonstrates
1033 that the statement of security problem definition is consistent, as defined by the conformance

- 1034 statement of the PP, with the statements of security problem definition stated in the PPs to which
1035 conformance is being claimed.
- 1036 If the PP under evaluation does not claim conformance with another PP, this work unit is not
1037 applicable and therefore considered to be satisfied.
- 1038 If the PP to which conformance is being claimed does not have a statement of security problem
1039 definition, this work unit is not applicable and therefore considered to be satisfied.
- 1040 If the PP to which conformance is being claimed contains functional packages, the evaluator
1041 determines that the security problem definition of the PP under evaluation consists of all
1042 assumptions, threats and OSPs of all functional packages.
- 1043 The terms exact, strict and demonstrable conformance are defined in ISO/IEC 15408 Part 1.
- 1044 If packages are used, the rules defined in the following paragraphs concerning exact, strict and
1045 demonstrable conformance also hold for the SPD descriptions taken from the packages.
- 1046 Note that since a PP can only claim conformance to another PP whose conformance statement
1047 requires strict or demonstrable conformance, those are the only cases covered in the following
1048 paragraphs. If strict conformance is required by the PP to which conformance is being claimed, no
1049 conformance claim rationale is required. Instead, the evaluator determines whether:
- 1050 a) the threats in the PP under evaluation are a superset of or identical to the threats in the
1051 PP to which conformance is being claimed;
- 1052 b) the OSPs in the PP under evaluation are a superset of or identical to the OSPs in the PP to
1053 which conformance is being claimed;
- 1054 c) the assumptions in the PP claiming conformance are identical to the assumptions in the
1055 PP to which conformance is being claimed, with two possible exceptions described in the
1056 following two bullet points;
- 1057 — an assumption (or part of an assumption) from the PP to which conformance is claimed, can be
1058 omitted, if all security objectives for the operational environment addressing this assumption
1059 (or part of an assumption) are replaced by security objectives for the TOE;
- 1060 — an assumption can be added to the assumptions defined in the PP to which conformance is
1061 claimed, if a justification is given, why the new assumption neither mitigates a threat (or a part
1062 of a threat) meant to be addressed by security objectives for the TOE in the PP to which
1063 conformance is claimed, nor fulfils an OSP (or part of an OSP) meant to be addressed by
1064 security objectives for the TOE in the PP to which conformance is claimed.
- 1065 For items “a” and “b”, it is allowed to omit SPD elements associated with optional requirements
1066 that are not included in the ST and still claim exact conformance.
- 1067 When examining a PP, which omits assumptions from another PP to which conformance is claimed,
1068 or adds new assumptions, the evaluator shall carefully determine, if the conditions given above are
1069 fulfilled. The following discussion gives some motivation and examples for these cases:
- 1070 — Example for omitting an assumption: A PP to which conformance is claimed, may contain an
1071 assumption stating that the operational environment prevents unauthorized modification or
1072 interception of data sent to an external interface of the TOE. This may be the case if the TOE
1073 accepts data in clear text and without integrity protection at this interface and is assumed to be
1074 located in a secure operational environment, which will prevent attackers from accessing these
1075 data. The assumption will then be mapped in the PP, to which conformance is claimed, to some
1076 objective for the operational environment stating that the data interchanged at this interface
1077 are protected by adequate measures in the operational environment. If a PP claiming this PP,

1078 defines a more secure TOE, which has an additional security objective stating that the TOE
 1079 itself protects these data, for example by providing a secure channel for encryption and
 1080 integrity protection of all data transferred via this interface, the corresponding objective and
 1081 assumption for the operational environment can be omitted from the PP claiming conformance.
 1082 This is also called re-assigning of the objective, since the objective is re-assigned from the
 1083 operational environment to the TOE. Note, that this TOE is still secure in an operational
 1084 environment fulfilling the omitted assumption and therefore still fulfils the PP to which
 1085 conformance is claimed.

1086 — Example for adding an assumption: In this example, the PP to which conformance is claimed, is
 1087 designed to specify requirements for a TOE of type "Firewall" and the author of another PP
 1088 wishes to claim conformance to this PP for a TOE, which implements a firewall, but
 1089 additionally provides the functionality of a virtual private network (VPN) component. For the
 1090 VPN functionality, the TOE needs cryptographic keys and these keys may also have to be
 1091 handled securely by the operational environment (e. g. if symmetric keys are used to secure
 1092 the network connection and therefore need to be provided in some secure way to other
 1093 components in the network). In this case, it is acceptable to add an assumption that the
 1094 cryptographic keys used by the VPN are handled securely by the operational environment.
 1095 This assumption does not address threats or OSPs of the PP to which conformance is claimed,
 1096 and therefore fulfils the conditions stated above.

1097 — Counterexample for adding an assumption: In a variant of the first example a PP to which
 1098 conformance is claimed, may already contain an objective for the TOE to provide a secure
 1099 channel for one of its interfaces, and this objective is mapped to a threat of unauthorized
 1100 modification or reading of the data on this interface. In this case, it is clearly not allowed for
 1101 another PP claiming this PP, to add an assumption for the operational environment, which
 1102 assumes that the operational environment protects data on this interface against modification
 1103 or unauthorized reading of the data. This assumption would reduce a threat, which is meant to
 1104 be addressed by the TOE. Therefore, a TOE fulfilling a PP with this added assumption would
 1105 not automatically fulfil the PP to which conformance is claimed, anymore and this addition is
 1106 therefore not allowed.

1107 — Second counterexample for adding an assumption: In the example above of a TOE
 1108 implementing a firewall it would not be admissible to add a general assumption that the TOE is
 1109 only connected to trusted devices, because this would obviously remove essential threats
 1110 relevant for a firewall (namely that there is untrusted IP traffic, which needs to be filtered).
 1111 Therefore, this addition would not be allowed.

1112 If demonstrable conformance is required by the PP to which conformance is being claimed, the
 1113 evaluator examines the conformance claim rationale to determine that it demonstrates that the
 1114 statement of security problem definition of the PP under evaluation is equivalent or more
 1115 restrictive than the statement of security problem definition in the PP to which conformance is
 1116 being claimed.

1117 For this, the conformance claim rationale needs to demonstrate that the security problem
 1118 definition in the PP claiming conformance is equivalent (or more restrictive) than the security
 1119 problem definition in the PP to which conformance is claimed. This means that:

1120 — all TOEs that would meet the security problem definition in the PP claiming conformance also
 1121 meet the security problem definition in the PP to which conformance is claimed. This can also
 1122 be shown indirectly by demonstrating that every event, which realizes a threat defined in the
 1123 PP to which conformance is claimed, or violates an OSP defined in the PP to which
 1124 conformance is claimed, would also realize a threat stated in the PP claiming conformance or
 1125 violate an OSP defined in the PP claiming conformance. Note that fulfilling an OSP stated in the
 1126 PP claiming conformance may avert a threat stated in the PP to which conformance is claimed,
 1127 or that averting a threat stated in the PP claiming conformance may fulfil an OSP stated in the
 1128 PP to which conformance is claimed, so threats and OSPs can substitute each other;

- 1129 — all operational environments that would meet the security problem definition in the PP to
 1130 which conformance is claimed, would also meet the security problem definition in the PP
 1131 claiming conformance (with one exception in the next bullet);
- 1132 — besides a set of assumptions in the PP claiming conformance needed to demonstrate
 1133 conformance to the SPD of the PP to which conformance is claimed, an PP claiming
 1134 conformance may specify further assumptions, but only if these additional assumptions are
 1135 independent of and do not affect the security problem definition as defined in the PP to which
 1136 conformance is claimed. More detailed, there are no assumptions in the PP claiming
 1137 conformance that exclude threats to the TOE that need to be countered by the TOE according
 1138 to the PP to which conformance is claimed. Similarly, there are no assumptions in the PP
 1139 claiming conformance that realize aspects of an OSP stated in the PP to which conformance is
 1140 claimed, which are meant to be fulfilled by the TOE according to the PP to which conformance
 1141 is claimed.
- 1142 ISO/IEC 15408-3 APE_CCL.1.9C: *The conformance claim rationale shall demonstrate that the*
 1143 *statement of security objectives is consistent with the statement of security objectives in the PPs for*
 1144 *which conformance is being claimed.*
- 1145 **8.4.1.3.11 Work unit APE_CCL.1-11**
- 1146 The evaluator ***shall examine*** the conformance claim rationale to determine that the statement of
 1147 security objectives is consistent, as defined by the conformance statement of the PPs, with the
 1148 statement of security objectives in the PPs.
- 1149 If the PP does not claim conformance to another PP, this work unit is not applicable and therefore
 1150 considered to be satisfied.
- 1151 If the PP to which conformance is being claimed contains functional packages, the evaluator
 1152 determines that the security objectives of the PP under evaluation consist of all security objectives
 1153 of all functional packages.
- 1154 If packages are used, the rules defined in the following paragraphs concerning exact, strict and
 1155 demonstrable conformance also hold for the security objectives taken from the packages.
- 1156 Note that since a PP can only claim conformance to another PP whose conformance statement
 1157 requires strict or demonstrable conformance, those are the only cases covered in the following
 1158 paragraphs. If strict conformance is required by the PP to which conformance is being claimed, no
 1159 conformance claim rationale is required. Instead, the evaluator determines whether:
- 1160 — The PP under evaluation contains all security objectives for the TOE of the PP to which
 1161 conformance is being claimed. Note that it is allowed for the PP under evaluation to have
 1162 additional security objectives for the TOE;
- 1163 — The security objectives for the operational environment in the PP claiming conformance are
 1164 identical to the security objectives for the operational environment in the PP to which
 1165 conformance is being claimed, with two possible exceptions described in the following two
 1166 bullet points;
- 1167 — a security objective for the operational environment (or part of such security objective) from
 1168 the PP to which conformance is claimed, can be replaced by the same (part of the) security
 1169 objective stated for the TOE;
- 1170 — a security objective for the operational environment can be added to the objectives defined in
 1171 the PP to which conformance is claimed, if a justification is given, why the new objective
 1172 neither mitigates a threat (or a part of a threat) meant to be addressed by security objectives
 1173 for the TOE in the PP to which conformance is claimed, nor fulfils an OSP (or part of an OSP)

1174 meant to be addressed by security objectives for the TOE in the PP to which conformance is
1175 claimed.

1176 When examining a PP claiming another PP which omits security objectives for the operational
1177 environment from the PP to which conformance is claimed, or adds new security objectives for the
1178 operational environment, the evaluator shall carefully determine, if the conditions given above are
1179 fulfilled. The examples given for the case of assumptions in the preceding work unit are also valid
1180 here.

1181 If demonstrable conformance is required by the PP to which conformance is being claimed, the
1182 evaluator examines the conformance claim rationale to determine that it demonstrates that the
1183 statement of security objectives of the PP under evaluation is equivalent or more restrictive than
1184 the statement of security objectives in the PP to which conformance is being claimed.

1185 For this the conformance claim rationale needs to demonstrate that the security objectives in the
1186 PP claiming conformance are equivalent (or more restrictive) than the security objectives in the PP
1187 to which conformance is claimed. This means that:

1188 — all TOEs that would meet the security objectives for the TOE in the PP claiming conformance
1189 also meet the security objectives for the TOE in the PP to which conformance is claimed;

1190 — all operational environments that would meet the security objectives for the operational
1191 environment in the PP to which conformance is claimed, would also meet the security
1192 objectives for the operational environment in the PP claiming conformance (with one
1193 exception in the next bullet);

1194 — besides a set of security objectives for the operational environment in the PP claiming
1195 conformance, which are used to demonstrate conformance to the set of security objectives
1196 defined in the PP to which conformance is claimed, an PP claiming conformance may specify
1197 further security objectives for the operational environment, but only if these security
1198 objectives neither affect the original set of security objectives for the TOE nor the security
1199 objectives for the operational environment as defined in the PP to which conformance is
1200 claimed.

1201 ISO/IEC 15408-3 APE_CCL.1.10C: *The conformance claim rationale shall demonstrate that the*
1202 *statement of security requirements is consistent with the statement of security requirements in the*
1203 *PPs for which conformance is being claimed.*

1204 **8.4.1.3.12 Work unit APE_CCL.1-12**

1205 The evaluator **shall examine** the PP to determine that it is consistent, as defined by the
1206 conformance statement of the PP, with all security requirements in the PPs for which conformance
1207 is being claimed.

1208 If the PP does not claim conformance to another PP, this work unit is not applicable and therefore
1209 considered to be satisfied.

1210 If the PP to which conformance is being claimed contains functional packages, the evaluator
1211 determines that the SFRs of the PP under evaluation consist of all SFRs (or hierarchical SFRs) of all
1212 functional packages.

1213 If packages are used, the rules defined in the following paragraphs concerning exact, strict and
1214 demonstrable conformance also hold for the SFRs taken from the packages.

1215 — Note that since a PP can only claim conformance to another PP whose conformance statement
1216 requires strict or demonstrable conformance, those are the only cases covered in the following
1217 paragraphs.

1218 If strict conformance is required by the PP to which conformance is being claimed, no conformance
 1219 claim rationale is required. Instead, the evaluator determines whether the statement of security
 1220 requirements in the PP under evaluation is a superset of or identical to the statement of security
 1221 requirements in the PP to which conformance is being claimed (for strict conformance).

1222 If demonstrable conformance is required by the PP to which conformance is being claimed, the
 1223 evaluator examines the conformance claim rationale to determine that it demonstrates that the
 1224 statement of security requirements of the PP under evaluation is equivalent or more restrictive
 1225 than the statement of security requirements in the PP to which conformance is being claimed.

1226 For:

1227 — SFRs: The conformance rationale in the PP claiming conformance shall demonstrate that the
 1228 overall set of requirements defined by the SFRs in the PP claiming conformance is equivalent
 1229 (or more restrictive) than the overall set of requirements defined by the SFRs in the PP to
 1230 which conformance is claimed. This means that all TOEs that would meet the requirements
 1231 defined by the set of all SFRs in the PP claiming conformance would also meet the
 1232 requirements defined by the set of all SFRs in the PP to which conformance is claimed;

1233 — SARs: The PP claiming conformance shall contain all SARs in the PP to which conformance is
 1234 claimed, but may claim additional SARs or replace SARs by hierarchically stronger SARs. The
 1235 completion of operations in the PP claiming conformance must be consistent with that in the
 1236 PP to which conformance is claimed; either the same completion will be used in the PP
 1237 claiming conformance as that in the PP to which conformance is claimed or a completion that
 1238 makes the SAR more restrictive (the rules of refinement apply).

1239 ISO/IEC 15408-3 APE_CCL.1.11C: *The conformance statement shall describe the conformance*
 1240 *required of any PPs/STs to the PP as exact-PP, strict-PP or demonstrable-PP conformance.*

1241 **8.4.1.3.13 Work unit APE_CCL.1-13**

1242 The evaluator **shall check** that the PP conformance statement states a claim of exact-PP, strict-PP
 1243 or demonstrable-PP conformance.

1244 ISO/IEC 15408-3 APE_CCL.1.12C: *The conformance statement shall identify the set of other PPs (if*
 1245 *any) to which, in combination with the PP under evaluation, exact conformance is allowed to be*
 1246 *claimed.*

1247 **8.4.1.3.14 Work unit APE_CCL.1-14**

1248 The evaluator **shall check** the conformance statement to determine that it lists the set of PPs to
 1249 which, in combination with the PP being evaluated, an exact conformance claim (in an ST or PP
 1250 Configuration) is allowed.

1251 If the PP does not require exact conformance in its conformance statement, this work unit does not
 1252 apply and is therefore considered satisfied.

1253 If the PP does not allow claims of exact conformance to it in combination with any other PPs, then
 1254 no list of PPs is required and this work unit is considered satisfied.

1255 The evaluator **shall check** that the list of the allowed PPs is consistent to the PP under evaluation,
 1256 and do not contradict the statements in the PP under evaluation by examining the following:

1257 • Verification and examination of all Threats, Assumptions, Objectives, SFRs, SARs taken
 1258 from the allowed PPs with those of the PP in evaluation.

1259 • Assessing that the SPD, the objectives and the SFRs in the allowed PPs do not contradict
 1260 the statements in the PP under evaluation

- 1261 • Checking that the assumptions and objectives for the environment are the same as in the
1262 PP under evaluation or that they are out of the scope of the PP under evaluation

1263 There are no other actions for the evaluator other than determining that the list is present.

1264 **8.4.1.3.15 Work unit APE_CCL.1-15**

1265 *[**This work unit has been deleted and renumbering of later work units may therefore be*
1266 *done in a future draft]*

1267 ISO/IEC 15408-3 APE_CCL.1.13C: *The conformance statement shall identify the set of PP-modules (if*
1268 *any) that are allowed to be used with the PP under evaluation in a PP-Configuration.*

1269 **8.4.1.3.16 Work unit APE_CCL.1-16**

1270 The evaluator shall check the conformance statement to determine that it lists the set of PP-
1271 Modules that can be used with the PP under evaluation in a PP-configuration.

1272 If the PP does not require exact conformance in its conformance statement, this work unit does not
1273 apply and is therefore considered satisfied.

1274 If the PP is not allowed to be used with any PP-Module in a PP-Configuration, then the evaluator
1275 confirms that no PP-modules are listed.

1276 There are no other actions for the evaluator other than determining that the list is present.

1277 **8.5 Security problem definition (APE_SPD)**

1278 **8.5.1 Evaluation of sub-activity (APE_SPD.1)**

1279 **8.5.1.1 Objectives**

1280 The objective of this sub-activity is to determine that the security problem intended to be
1281 addressed by the TOE and its operational environment is clearly defined.

1282 **8.5.1.2 Input**

1283 The evaluation evidence for this sub-activity is:

1284 a) the PP.

1285 **8.5.1.3 Action APE_SPD.1.1E**

1286 ISO/IEC 15408-3 APE_SPD.1.1C: *The security problem definition shall describe the threats.*

1287 **8.5.1.3.1 Work unit APE_SPD.1-1**

1288 The evaluator **shall check** that the security problem definition describes the threats.

1289 If all security objectives are derived from assumptions and/or OSPs only, the statement of threats
1290 need not be present in the PP. In this case, this work unit is not applicable and therefore considered
1291 to be satisfied.

1292 The evaluator determines that the security problem definition describes the threats that must be
1293 countered by the TOE and/or its operational environment.

1294 Note that if optional requirements are defined by the PP, there may be associated threats that are
1295 covered by this work unit.

- 1296 ISO/IEC 15408-3 APE_SPD.1.2C: *All threats shall be described in terms of a threat agent, an asset,*
1297 *and an adverse action.*
- 1298 **8.5.1.3.2 Work unit APE_SPD.1-2**
- 1299 The evaluator ***shall examine*** the security problem definition to determine that all threats are
1300 described in terms of a threat agent, an asset, and an adverse action.
- 1301 If all security objectives are derived from assumptions and OSPs only, the statement of threats
1302 need not be present in the PP. In this case, this work unit is not applicable and therefore considered
1303 to be satisfied.
- 1304 Threat agents may be further described by aspects such as expertise, resource, opportunity, and
1305 motivation.
- 1306 ISO/IEC 15408-3 APE_SPD.1.3C: *The security problem definition shall describe the OSPs.*
- 1307 **8.5.1.3.3 Work unit APE_SPD.1-3**
- 1308 The evaluator ***shall examine*** that the security problem definition describes the OSPs.
- 1309 If all security objectives are derived from assumptions and/or threats only, OSPs need not be
1310 present in the PP. In this case, this work unit is not applicable and therefore considered to be
1311 satisfied.
- 1312 The evaluator determines that OSP statements are made in terms of rules or guidelines that must
1313 be followed by the TOE and/or its operational environment.
- 1314 The evaluator determines that each OSP is explained and/or interpreted in sufficient detail to make
1315 it clearly understandable; a clear presentation of policy statements is necessary to permit tracing
1316 security objectives to them.
- 1317 Note that if optional requirements are defined by the PP, there may be associated threats that are
1318 covered by this work unit.
- 1319 ISO/IEC 15408-3 APE_SPD.1.4C: *The security problem definition shall describe the assumptions*
1320 *about the operational environment of the TOE.*
- 1321 **8.5.1.3.4 Work unit APE_SPD.1-4**
- 1322 The evaluator ***shall examine*** the security problem definition to determine that it describes the
1323 assumptions about the operational environment of the TOE.
- 1324 If there are no assumptions, this work unit is not applicable and is therefore considered to be
1325 satisfied.
- 1326 The evaluator determines that each assumption about the operational environment of the TOE is
1327 explained in sufficient detail to enable consumers to determine that their operational environment
1328 matches the assumption. If the assumptions are not clearly understood, the end result may be that
1329 the TOE is used in an operational environment in which it will not function in a secure manner.

1330 **8.6 Security objectives (APE_OBJ)**

1331 **8.6.1 Evaluation of sub-activity (APE_OBJ.1)**

1332 **8.6.1.1 Objectives**

1333 The objective of this sub-activity is to determine whether the security objectives for the
1334 operational environment are clearly defined.

1335 **8.6.1.2 Input**

1336 The evaluation evidence for this sub-activity is:

1337 a) the PP.

1338 **8.6.1.3 Action APE_OBJ.1.1E**

1339 ISO/IEC 15408-3 APE_OBJ.1.1C: *The statement of security objectives shall describe the security*
1340 *objectives for the operational environment.*

1341 **8.6.1.3.1 Work unit APE_OBJ.1-1**

1342 The evaluator **shall check** that the statement of security objectives defines the security objectives
1343 for the operational environment.

1344 The evaluator checks that the security objectives for the operational environment are identified.

1345 **8.6.2 Evaluation of sub-activity (APE_OBJ.2)**

1346 **8.6.2.1 Objectives**

1347 The objective of this sub-activity is to determine whether the security objectives adequately and
1348 completely address the security problem definition and that the division of this problem between
1349 the TOE and its operational environment is clearly defined.

1350 **8.6.2.2 Input**

1351 The evaluation evidence for this sub-activity is:

1352 a) the PP.

1353 **8.6.2.3 Action APE_OBJ.2.1E**

1354 ISO/IEC 15408-3 APE_OBJ.2.1C: *The statement of security objectives shall describe the security*
1355 *objectives for the TOE and the security objectives for the operational environment.*

1356 **8.6.2.3.1 Work unit APE_OBJ.2-1**

1357 The evaluator **shall check** that the statement of security objectives defines the security objectives
1358 for the TOE and the security objectives for the operational environment.

1359 The evaluator checks that both categories of security objectives are clearly identified and
1360 separated from the other category.

1361 ISO/IEC 15408-3 APE_OBJ.2.2C: *The security objectives rationale shall trace each security objective*
1362 *for the TOE back to threats countered by that security objective and OSPs enforced by that security*
1363 *objective.*

1364 **8.6.2.3.2 Work unit APE_OBJ.2-2**

1365 The evaluator **shall check** that the security objectives rationale traces all security objectives for the
1366 TOE back to threats countered by the objectives and/or OSPs enforced by the objectives.

1367 Each security objective for the TOE may trace back to threats or OSPs, or a combination of threats
1368 and OSPs, but it must trace back to at least one threat or OSP. Optional requirements may require
1369 Threats/OSP to be specified, and security objectives associated with these SPD elements are also
1370 covered by this work unit.

1371 Failure to trace implies that either the security objectives rationale is incomplete, the security
1372 problem definition is incomplete, or the security objective for the TOE has no useful purpose.

1373 ISO/IEC 15408-3 APE_OBJ.2.3C: *The security objectives rationale shall trace each security objective*
1374 *for the operational environment back to threats countered by that security objective, OSPs enforced*
1375 *by that security objective, and assumptions upheld by that security objective.*

1376 **8.6.2.3.3 Work unit APE_OBJ.2-3**

1377 The evaluator **shall check** that the security objectives rationale traces the security objectives for
1378 the operational environment back to threats countered by that security objective, to OSPs enforced
1379 by that security objective, and to assumptions upheld by that security objective.

1380 Each security objective for the operational environment may trace back to threats, OSPs,
1381 assumptions, or a combination of threats, OSPs and/or assumptions, but it must trace back to at
1382 least one threat, OSP or assumption.

1383 Failure to trace implies that either the security objectives rationale is incomplete, the security
1384 problem definition is incomplete, or the security objective for the operational environment has no
1385 useful purpose.

1386 ISO/IEC 15408-3 APE_OBJ.2.4C: *The security objectives rationale shall demonstrate that the security*
1387 *objectives counter all threats.*

1388 **8.6.2.3.4 Work unit APE_OBJ.2-4**

1389 The evaluator **shall examine** the security objectives rationale to determine that it justifies for each
1390 threat that the security objectives are suitable to counter that threat.

1391 If no security objectives trace back to the threat, the evaluator action related to this work unit is
1392 assigned a fail verdict.

1393 The evaluator determines that the justification for a threat shows whether the threat is removed,
1394 diminished or mitigated.

1395 The evaluator determines that the justification for a threat demonstrates that the security
1396 objectives are sufficient: if all security objectives that trace back to the threat are achieved, the
1397 threat is removed, sufficiently diminished, or the effects of the threat are sufficiently mitigated.

1398 Note that the tracings from security objectives to threats provided in the security objectives
1399 rationale may be part of a justification, but do not constitute a justification by themselves. Even in
1400 the case that a security objective is merely a statement reflecting the intent to prevent a particular
1401 threat from being realised, a justification is required, but this justification may be as minimal as
1402 "Security Objective X directly counters Threat Y".

1403 The evaluator also determines that each security objective that traces back to a threat is necessary:
1404 when the security objective is achieved it actually contributes to the removal, diminishing or
1405 mitigation of that threat.

1406 ISO/IEC 15408-3 APE_OBJ.2.5C: *The security objectives rationale shall demonstrate that the security*
1407 *objectives enforce all OSPs.*

1408 **8.6.2.3.5 Work unit APE_OBJ.2-5**

1409 The evaluator **shall examine** the security objectives rationale to determine that for each OSP it
1410 justifies that the security objectives are suitable to enforce that OSP.

1411 If no security objectives trace back to the OSP, the evaluator action related to this work unit is
1412 assigned a fail verdict.

1413 The evaluator determines that the justification for an OSP demonstrates that the security
1414 objectives are sufficient: if all security objectives that trace back to that OSP are achieved, the OSP
1415 is enforced.

1416 The evaluator also determines that each security objective that traces back to an OSP is necessary:
1417 when the security objective is achieved it actually contributes to the enforcement of the OSP.

1418 Note that the tracings from security objectives to OSPs provided in the security objectives rationale
1419 may be part of a justification, but do not constitute a justification by themselves. In the case that a
1420 security objective is merely a statement reflecting the intent to enforce a particular OSP, a
1421 justification is required, but this justification may be as minimal as “Security Objective X directly
1422 enforces OSP Y”.

1423 ISO/IEC 15408-3 APE_OBJ.2.6C: *The security objectives rationale shall demonstrate that the security*
1424 *objectives for the operational environment uphold all assumptions.*

1425 **8.6.2.3.6 Work unit APE_OBJ.2-6**

1426 The evaluator **shall examine** the security objectives rationale to determine that for each
1427 assumption for the operational environment it contains an appropriate justification that the
1428 security objectives for the operational environment are suitable to uphold that assumption.

1429 If no security objectives for the operational environment trace back to the assumption, the
1430 evaluator action related to this work unit is assigned a fail verdict.

1431 The evaluator determines that the justification for an assumption about the operational
1432 environment of the TOE demonstrates that the security objectives are sufficient: if all security
1433 objectives for the operational environment that trace back to that assumption are achieved, the
1434 operational environment upholds the assumption.

1435 The evaluator also determines that each security objective for the operational environment that
1436 traces back to an assumption about the operational environment of the TOE is necessary: when the
1437 security objective is achieved it actually contributes to the operational environment upholding the
1438 assumption.

1439 Note that the tracings from security objectives for the operational environment to assumptions
1440 provided in the security objectives rationale may be a part of a justification, but do not constitute a
1441 justification by themselves. Even in the case that a security objective of the operational
1442 environment is merely a restatement of an assumption, a justification is required, but this
1443 justification may be as minimal as “Security Objective X directly upholds Assumption Y”.

1444 **8.7 Extended components definition (APE_ECD)**1445 **8.7.1 Evaluation of sub-activity (APE_ECD.1)**1446 **8.7.1.1 Objectives**

1447 The objective of this sub-activity is to determine whether extended components have been clearly
 1448 and unambiguously defined, and whether they are necessary, i.e. they may not be clearly expressed
 1449 using existing ISO/IEC 15408-2 or ISO/IEC 15408-3 components.

1450 **8.7.1.2 Input**

1451 The evaluation evidence for this sub-activity is:

1452 a) the PP.

1453 **8.7.1.3 Action APE_ECD.1.1E**

1454 ISO/IEC 15408-3 APE_ECD.1.1C: *The statement of security requirements shall identify all extended*
 1455 *security requirements.*

1456 **8.7.1.3.1 Work unit APE_ECD.1-1**

1457 The evaluator **shall check** that all security requirements in the statement of security requirements
 1458 that are not identified as extended requirements are present in ISO/IEC 15408-2 or in ISO/IEC
 1459 15408-3.

1460 ISO/IEC 15408-3 APE_ECD.1.2C: *The extended components definition shall define an extended*
 1461 *component for each extended security requirement.*

1462 **8.7.1.3.2 Work unit APE_ECD.1-2**

1463 The evaluator **shall check** that the extended components definition defines an extended
 1464 component for each extended security requirement.

1465 If the PP does not contain extended security requirements, this work unit is not applicable and
 1466 therefore considered to be satisfied.

1467 A single extended component may be used to define multiple iterations of an extended security
 1468 requirement, it is not necessary to repeat this definition for each iteration.

1469 ISO/IEC 15408-3 APE_ECD.1.3C: *The extended components definition shall describe how each*
 1470 *extended component is related to the existing ISO/IEC 15408 components, families, and classes.*

1471 **8.7.1.3.3 Work unit APE_ECD.1-3**

1472 The evaluator **shall examine** the extended components definition to determine that it describes
 1473 how each extended component fits into the existing ISO/IEC 15408 components, families, and
 1474 classes.

1475 If the PP does not contain extended security requirements, this work unit is not applicable and
 1476 therefore considered to be satisfied.

1477 The evaluator determines that each extended component is either:

1478 a) a member of an existing ISO/IEC 15408-2 or ISO/IEC 15408-3 family, or

1479 b) a member of a new family defined in the PP.

1480 If the extended component is a member of an existing ISO/IEC 15408-2 or ISO/IEC 15408-3 family,
1481 the evaluator determines that the extended components definition adequately describes why the
1482 extended component should be a member of that family and how it relates to other components of
1483 that family.

1484 If the extended component is a member of a new family defined in the PP, the evaluator confirms
1485 that the extended component is not appropriate for an existing family.

1486 If the PP defines new families, the evaluator determines that each new family is either:

1487 a) a member of an existing ISO/IEC 15408-2 or ISO/IEC 15408-3 class, or

1488 a member of a new class defined in the PP.

1489 If the family is a member of an existing ISO/IEC 15408-2 or ISO/IEC 15408-3 class, the evaluator
1490 determines that the extended components definition adequately describes why the family should
1491 be a member of that class and how it relates to other families in that class.

1492 If the family is a member of a new class defined in the PP, the evaluator confirms that the family is
1493 not appropriate for an existing class.

1494 **8.7.1.3.4 Work unit APE_ECD.1-4**

1495 The evaluator *shall examine* the extended components definition to determine that each definition
1496 of an extended component identifies all applicable dependencies of that component.

1497 If the PP does not contain extended security requirements, this work unit is not applicable and
1498 therefore considered to be satisfied.

1499 The evaluator confirms that no applicable dependencies have been overlooked by the PP author.

1500 ISO/IEC 15408-3 APE_ECD.1.4C: *The extended components definition shall use the existing ISO/IEC*
1501 *15408 components, families, classes, and methodology as a model for presentation.*

1502 **8.7.1.3.5 Work unit APE_ECD.1-5**

1503 The evaluator *shall examine* the extended components definition to determine that each extended
1504 functional component uses the existing ISO/IEC 15408-2 components as a model for presentation.

1505 If the PP does not contain extended SFRs, this work unit is not applicable and therefore considered
1506 to be satisfied.

1507 The evaluator determines that the extended functional component is consistent with ISO/IEC
1508 15408-2 Subclause **6.1.3, Component structure**.

1509 If the extended functional component uses operations, the evaluator determines that the extended
1510 functional component is consistent with ISO/IEC 15408-1 Subclause **7.1, Operations**.

1511 If the extended functional component is hierarchical to an existing functional component, the
1512 evaluator determines that the extended functional component is consistent with ISO/IEC 15408-2
1513 Subclause **6.2.1, Component changes highlighting**.

1514 **8.7.1.3.6 Work unit APE_ECD.1-6**

1515 The evaluator *shall examine* the extended components definition to determine that each definition
1516 of a new functional family uses the existing ISO/IEC 15408 functional families as a model for
1517 presentation.

- 1518 If the PP does not define new functional families, this work unit is not applicable and therefore
1519 considered to be satisfied.
- 1520 The evaluator determines that all new functional families are defined consistent with ISO/IEC
1521 15408-2 Subclause **6.1.2, Family structure**.
- 1522 **8.7.1.3.7 Work unit APE_ECD.1-7**
- 1523 The evaluator *shall examine* the extended components definition to determine that each definition
1524 of a new functional class uses the existing ISO/IEC 15408 functional classes as a model for
1525 presentation.
- 1526 If the PP does not define new functional classes, this work unit is not applicable and therefore
1527 considered to be satisfied.
- 1528 The evaluator determines that all new functional classes are defined consistent with ISO/IEC
1529 15408-2 Subclause **6.1.1, Class structure**.
- 1530 **8.7.1.3.8 Work unit APE_ECD.1-8**
- 1531 The evaluator *shall examine* the extended components definition to determine that each definition
1532 of an extended assurance component uses the existing ISO/IEC 15408-3 components as a model
1533 for presentation.
- 1534 If the PP does not contain extended SARs, this work unit is not applicable and therefore considered
1535 to be satisfied.
- 1536 The evaluator determines that the extended assurance component definition is consistent with
1537 ISO/IEC 15408-3 Subclause **6.1.3, Assurance component structure**.
- 1538 If the extended assurance component uses operations, the evaluator determines that the extended
1539 assurance component is consistent with ISO/IEC 15408-1 Subclause **7.1, Operations**.
- 1540 If the extended assurance component is hierarchical to an existing assurance component, the
1541 evaluator determines that the extended assurance component is consistent with ISO/IEC 15408-3
1542 Subclause **6.1.3, Assurance component structure**.
- 1543 **8.7.1.3.9 Work unit APE_ECD.1-9**
- 1544 The evaluator *shall examine* the extended components definition to determine that, for each
1545 defined extended assurance component, applicable methodology has been provided.
- 1546 If the PP does not contain extended SARs, this work unit is not applicable and therefore considered
1547 to be satisfied.
- 1548 The evaluator determines that, for each evaluator action element of each extended SAR, one or
1549 more work units are provided and that successfully performing all work units for a given evaluator
1550 action element will demonstrate that the element has been achieved.
- 1551 **8.7.1.3.10 Work unit APE_ECD.1-10**
- 1552 The evaluator *shall examine* the extended components definition to determine that each definition
1553 of a new assurance family uses the existing ISO/IEC 15408 assurance families as a model for
1554 presentation.
- 1555 If the PP does not define new assurance families, this work unit is not applicable and therefore
1556 considered to be satisfied.

1557 The evaluator determines that all new assurance families are defined consistent with ISO/IEC
1558 15408-3 Subclause 6.1.2, Assurance family structure.

1559 **8.7.1.3.11 Work unit APE_ECD.1-11**

1560 The evaluator *shall examine* the extended components definition to determine that each definition
1561 of a new assurance class uses the existing ISO/IEC 15408 assurance classes as a model for
1562 presentation.

1563 If the PP does not define new assurance classes, this work unit is not applicable and therefore
1564 considered to be satisfied.

1565 The evaluator determines that all new assurance classes are defined consistent with ISO/IEC
1566 15408-3 Subclause 6.1.1, Assurance class structure.

1567 ISO/IEC 15408-3 APE_ECD.1.5C: *The extended components shall consist of measurable and objective*
1568 *elements such that conformance or nonconformance to these elements can be demonstrated.*

1569 **8.7.1.3.12 Work unit APE_ECD.1-12**

1570 The evaluator *shall examine* the extended components definition to determine that each element
1571 in each extended component is measurable and states objective evaluation requirements, such that
1572 conformance or nonconformance can be demonstrated.

1573 If the PP does not contain extended security requirements, this work unit is not applicable and
1574 therefore considered to be satisfied.

1575 The evaluator determines that elements of extended functional components are stated in such a
1576 way that they are testable, and traceable through the appropriate TSF representations.

1577 The evaluator also determines that elements of extended assurance components avoid the need for
1578 subjective evaluator judgement.

1579 The evaluator is reminded that whilst being measurable and objective is appropriate for all
1580 evaluation criteria, it is acknowledged that no formal method exists to prove such properties.
1581 Therefore the existing ISO/IEC 15408 functional and assurance components are to be used as a
1582 model for determining what constitutes conformance to this requirement.

1583 **8.7.1.4 Action APE_ECD.1.2E**

1584 **8.7.1.4.1 Work unit APE_ECD.1-13**

1585 The evaluator *shall examine* the extended components definition to determine that each extended
1586 component may not be clearly expressed using existing components.

1587 If the PP does not contain extended security requirements, this work unit is not applicable and
1588 therefore considered to be satisfied.

1589 The evaluator should take components from ISO/IEC 15408-2 and ISO/IEC 15408-3, other
1590 extended components that have been defined in the PP, combinations of these components, and
1591 possible operations on these components into account when making this determination.

1592 The evaluator is reminded that the role of this work unit is to preclude unnecessary duplication of
1593 components, that is, components that may be clearly expressed by using other components. The
1594 evaluator should not undertake an exhaustive search of all possible combinations of components
1595 including operations in an attempt to find a way to express the extended component by using
1596 existing components.

1597 **8.8 Security requirements (APE_REQ)**1598 **8.8.1 Evaluation of sub-activity (APE_REQ.1)**1599 **8.8.1.1 Objectives**

1600 The objective of this sub-activity is to determine whether the SFRs and SARs are clear,
 1601 unambiguous and well-defined, whether they are internally consistent, and whether the SFRs
 1602 counter the threats and implement the organisational security policies of the TOE.

1603 **8.8.1.2 Input**

1604 The evaluation evidence for this sub-activity is:

1605 a) the PP.

1606 **8.8.1.3 Action APE_REQ.1.1E**

1607 ISO/IEC 15408-3 APE_REQ.1.1C: *The statement of security requirements shall describe the SFRs and*
 1608 *the SARs.*

1609 **8.8.1.3.1 Work unit APE_REQ.1-1**

1610 The evaluator **shall check** that the statement of security requirements describes the SFRs.

1611 The evaluator determines that each SFR is identified by one of the following means:

1612 a) by reference to an individual component in ISO/IEC 15408-2;

1613 b) by reference to an extended component in the extended components definition of the PP;

1614 c) by reference to a PP that the PP claims to be conformant with including any optional
 1615 requirements defined in the PP;

1616 d) by reference to a security requirements package that the PP claims to be conformant
 1617 with;

1618 e) by reproduction in the PP.

1619 It is not required to use the same means of identification for all SFRs.

1620 **8.8.1.3.2 Work unit APE_REQ.1-2**

1621 The evaluator **shall check** that the statement of security requirements describes the SARs.

1622 The evaluator determines that each SAR is identified by one of the following means:

1623 a) by reference to an individual component in ISO/IEC 15408-3;

1624 b) by reference to an extended component in the extended components definition of the PP;

1625 c) by reference to a PP that the PP claims to be conformant with;

1626 d) by reference to a security requirements package that the PP claims to be conformant
 1627 with;

1628 e) by reproduction in the PP.

1629 It is not required to use the same means of identification for all SARs.
1630 ISO/IEC 15408-3 APE_REQ.1.2C: *All subjects, objects, operations, security attributes, external entities*
1631 *and other terms that are used in the SFRs and the SARs shall be defined.*

1632 **8.8.1.3.3 Work unit APE_REQ.1-3**

1633 The evaluator **shall examine** the PP to determine that all subjects, objects, operations, security
1634 attributes, external entities and other terms that are used in the SFRs and the SARs are defined.

1635 The evaluator determines that the PP defines all:

1636 — (types of) subjects and objects that are used in the SFRs;

1637 — (types of) security attributes of subjects, users, objects, information, sessions and/or resources,
1638 possible values that these attributes may take and any relations between these values (e.g.
1639 top_secret is “higher” than secret);

1640 — (types of) operations that are used in the SFRs, including the effects of these operations;

1641 — (types of) external entities in the SFRs;

1642 — other terms that are introduced in the SFRs and/or SARs by completing operations, if these
1643 terms are not immediately clear, or are used outside their dictionary definition.

1644 The goal of this work unit is to ensure that the SFRs and SARs are well-defined and that no
1645 misunderstanding may occur due to the introduction of vague terms. This work unit should not be
1646 taken into extremes, by forcing the PP author to define every single word. The general audience of
1647 a set of security requirements should be assumed to have a reasonable knowledge of IT, security
1648 and “Evaluation criteria for IT security”.

1649 All of the above may be presented in groups, classes, roles, types or other groupings or
1650 characterisations that allow easy understanding.

1651 The evaluator is reminded that these lists and definitions do not have to be part of the statement of
1652 security requirements, but may be placed (in part or in whole) in different subclauses. This may be
1653 especially applicable if the same terms are used in the rest of the PP.

1654 ISO/IEC 15408-3 APE_REQ.1.3C: *The statement of security requirements shall include a natural*
1655 *language description, part of which describes how the SFRs combine together to provide security*
1656 *functionality in terms of the architecture that is visible to Administrators and other users.*

1657 **8.8.1.3.4 Work unit APE_REQ.1-4**

1658 The evaluator **shall check** that the statement of security requirements includes a natural language
1659 description, part of which describes how the SFRs combine together to provide security
1660 functionality in terms of the architecture that is visible to Administrators and other users.

1661 The description is intended to make clear connections between SFRs and to provide a view of how
1662 they provide security functionality that is recognizable to Administrators and other types of user.
1663 The description in terms of the architecture that is “visible to Administrators and other users”
1664 means that the description must relate the security behavior to visible elements, but the
1665 mechanisms themselves need not be visible. For example: when describing authentication using a
1666 biometric mechanism, the calculation of the match or score might not be visible, but (a) might
1667 relate to a referenced description of a matching algorithm, (b) might be based on specific template
1668 files maintained by the Administrator, and (c) will result in acceptance or rejection of the
1669 authentication attempt – therefore the description might make use of any or all of these items (a) –

1670 (c). No specific format for this information is prescribed, and the description need not all be located
 1671 alongside the SFRs themselves (e.g. some of it might be in the PP Introduction). The intention of the
 1672 requirement is to make the meaning of the SFRs clearer and more easily understood by readers of
 1673 the PP who may not have deep knowledge of the CC but who are familiar with the product type.

1674 The evaluator determines that all operations are identified in each SFR or SAR where such an
 1675 operation is used. This includes both completed operations and uncompleted operations.
 1676 Identification may be achieved by typographical distinctions, or by explicit identification in the
 1677 surrounding text, or by any other distinctive means.

1678 ISO/IEC 15408-3 APE_REQ.1.4C: *The statement of security requirements shall identify all operations*
 1679 *on the security requirements.*

1680 **8.8.1.3.5 Work unit APE_REQ.1-5**

1681 The evaluator ***shall check*** that the statement of security requirements identifies all operations on
 1682 the security requirements.

1683 The evaluator determines that all operations are identified in each SFR or SAR where such an
 1684 operation is used. This includes both completed operations and uncompleted operations.
 1685 Identification may be achieved by typographical distinctions, or by explicit identification in the
 1686 surrounding text, or by any other distinctive means.

1687 *If the PP defines selection-based SFRs, the evaluator determines that the PP clearly identifies the*
 1688 *dependencies between the selection in an SFR and the selection-based SFR(s) to be included in the*
 1689 *PP/ST should that selection be chosen by the PP/ST author.*

1690 ISO/IEC 15408-3 APE_REQ.1.5C: *All operations shall be performed correctly.*

1691 **8.8.1.3.6 Work unit APE_REQ.1-6**

1692 The evaluator ***shall examine*** the statement of security requirements to determine that all
 1693 assignment operations are performed correctly.

1694 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
 1695 **Guidance for Operations.**

1696 **8.8.1.3.7 Work unit APE_REQ.1-7**

1697 The evaluator ***shall examine*** the statement of security requirements to determine that all iteration
 1698 operations are performed correctly.

1699 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
 1700 **Guidance for Operations.**

1701 **8.8.1.3.8 Work unit APE_REQ.1-8**

1702 The evaluator ***shall examine*** the statement of security requirements to determine that all selection
 1703 operations are performed correctly.

1704 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
 1705 **Guidance for Operations.**

1706 **8.8.1.3.9 Work unit APE_REQ.1-9**

1707 The evaluator ***shall examine*** the statement of security requirements to determine that all
 1708 refinement operations are performed correctly.

1709 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
1710 **Guidance for Operations**.

1711 ISO/IEC 15408-3 APE_REQ.1.6C: *Each dependency of the security requirements shall either be*
1712 *satisfied, or the security requirements rationale shall justify the dependency not being satisfied.*

1713 **8.8.1.3.10 Work unit APE_REQ.1-10**

1714 The evaluator **shall examine** the statement of security requirements to determine that each
1715 dependency of the security requirements is either satisfied, or that the security requirements
1716 rationale justifies the dependency not being satisfied.

1717 A dependency is satisfied by the inclusion of the relevant component (or one that is hierarchical to
1718 it) within the statement of security requirements. The component used to satisfy the dependency
1719 should, if necessary, be modified by operations to ensure that it actually satisfies that dependency.

1720 A justification that a dependency is not met should address either:

1721 a) why the dependency is not necessary or useful, in which case no further information is
1722 required; or

1723 b) that the dependency has been addressed by the operational environment of the TOE, in
1724 which case the justification should describe how the security objectives for the
1725 operational environment address this dependency.

1726 ISO/IEC 15408-3 APE_REQ.1.7C: *The security requirements rationale shall trace each SFR back to*
1727 *the threats countered by that SFR and OSPs enforced by that SFR.*

1728 **8.8.1.3.11 Work unit APE_REQ.1-11**

1729 The evaluator **shall check** that the security requirements rationale traces each SFR back to the
1730 threats countered by that SFR and OSPs enforced by that SFR.

1731 The evaluator determines that each SFR is traced back to at least one threat or OSP for the TOE.

1732 Failure to trace implies that either the security requirements rationale is incomplete, the security
1733 objectives for the TOE are incomplete, or the SFR has no useful purpose.

1734 There is no prescribed location for this part of the rationale: for example, the relevant parts may be
1735 located under each threat and OSP in order to help make the security argument clearer and easier
1736 to read.

1737 ISO/IEC 15408-3 APE_REQ.1.8C: *The security requirements rationale shall trace each security*
1738 *objective for the operational environment back to threats countered by that security objective, OSPs*
1739 *enforced by that security objective, and assumptions upheld by that security objective.*

1740 **8.8.1.3.12 Work unit APE_REQ.1-12**

1741 The evaluator **shall check** that the security objectives requirements rationale traces the security
1742 objectives for the operational environment back to threats countered by that security objective, to
1743 OSPs enforced by that security objective, and to assumptions upheld by that security objective.

1744 Each security objective for the operational environment may trace back to threats, OSPs,
1745 assumptions, or a combination of threats, OSPs and/or assumptions, but it must trace back to at
1746 least one threat, OSP or assumption.

- 1747 Failure to trace implies that either the security objectives requirements rationale is incomplete, the
1748 security problem definition is incomplete, or the security objective for the operational
1749 environment has no useful purpose.
- 1750 There is no prescribed location for this part of the rationale: for example, the relevant parts may be
1751 located under each threat, OSP and assumption in order to help make the security argument
1752 clearer and easier to read.
- 1753 ISO/IEC 15408-3 APE_REQ.1.9C: *The security requirements rationale shall demonstrate that the*
1754 *SFRs (in conjunction with the security objectives for the environment) counter all threats for the TOE.*
- 1755 **8.8.1.3.13 Work unit APE_REQ.1-13**
- 1756 The evaluator ***shall examine*** the security requirements rationale to determine that for each threat
1757 it demonstrates that the SFRs are suitable to meet that threat.
- 1758 If no SFRs trace back to a threat, the evaluator action related to this work unit is assigned a fail
1759 verdict.
- 1760 The evaluator determines that the justification for a threat shows whether the threat is removed,
1761 diminished or mitigated.
- 1762 The evaluator determines that the justification for a threat demonstrates that the SFRs are
1763 sufficient: if all SFRs that trace back to the threat are achieved then, in the context of any applicable
1764 OSPs and assumptions, the threat is removed, sufficiently diminished, or the effects of the threat
1765 are sufficiently mitigated.
- 1766 Note that simply listing in the security requirements rationale the SFRs associated with each threat
1767 may be part of a justification, but does not constitute a justification by itself. A descriptive
1768 justification is required, although in simple cases this justification may be as minimal as “SFR X
1769 directly counters Threat Y”.
- 1770 The evaluator also determines that each SFR that traces back to a threat is necessary: when the SFR
1771 is implemented it actually contributes to the removal, diminishing or mitigation of that threat.
- 1772 ISO/IEC 15408-3 APE_REQ.1.10C: *The security requirements rationale shall demonstrate that the*
1773 *SFRs (in conjunction with the security objectives for the environment) enforce all OSPs for the TOE.*
- 1774 **8.8.1.3.14 Work unit APE_REQ.1-14**
- 1775 The evaluator ***shall examine*** the security requirements rationale to determine that for each OSP it
1776 justifies that the SFRs are suitable to enforce that OSP.
- 1777 If no SFRs or security objectives for the operational environment trace back to the OSP, the
1778 evaluator action related to this work unit is assigned a fail verdict.
- 1779 The evaluator determines that the justification for an OSP demonstrates that the security
1780 objectives are sufficient: if all SFRs that trace back to that OSP are achieved then, in the context of
1781 any applicable assumptions, the OSP is enforced.
- 1782 The evaluator also determines that each SFR that traces back to an OSP is necessary: when the SFR
1783 is implemented it actually contributes to the enforcement of the OSP.
- 1784 Note that simply listing in the security requirements rationale the SFRs associated with each OSP
1785 may be part of a justification, but does not constitute a justification by itself. A descriptive
1786 justification is required, although in simple cases this justification may be as minimal as “SFR X
1787 directly enforces OSP Y”.

1788 ISO/IEC 15408-3 APE_REQ.1.11C: The security requirements rationale shall demonstrate that the
1789 SFRs (in conjunction with the security objectives for the environment) enforce all OSPs for the TOE.

1790 **8.8.1.3.15 Work unit APE_REQ.1-15**

1791 The evaluator *shall examine* the security requirements rationale to determine that for each
1792 assumption for the operational environment it contains an appropriate justification that the
1793 security objectives for the operational environment are suitable to uphold that assumption.

1794 If no security objectives for the operational environment trace back to the assumption, the
1795 evaluator action related to this work unit is assigned a fail verdict.

1796 The evaluator determines that the justification for an assumption about the operational
1797 environment of the TOE demonstrates that the security objectives are sufficient: if all security
1798 objectives for the operational environment that trace back to that assumption are achieved, the
1799 operational environment upholds the assumption.

1800 The evaluator also determines that each security objective for the operational environment that
1801 traces back to an assumption about the operational environment of the TOE is necessary: when the
1802 security objective is achieved it actually contributes to the operational environment upholding the
1803 assumption.

1804 Note that simply listing in the security requirements rationale the security objectives for the
1805 operational environment associated with each assumption may be a part of a justification, but does
1806 not constitute a justification by itself. A descriptive justification is required, although in simple
1807 cases this justification may be as minimal as "Security Objective X directly upholds Assumption Y".

1808

1809 ISO/IEC 15408-3 APE_REQ.1.12C: *The statement of security requirements shall be internally*
1810 *consistent.*

1811 **8.8.1.3.16 Work unit APE_REQ.1-16**

1812 The evaluator *shall examine* the statement of security requirements to determine that it is
1813 internally consistent.

1814 The evaluator determines that the combined set of all SFRs and SARs is internally consistent. With
1815 respect to optional requirements, the evaluator determines that:

1816 a) All optional requirements either trace to an SPD element that is itself not optional, or trace
1817 to an SPD element that is clearly associated with that optional SFR;

1818 b) All optional requirements are clearly identified as being required if a conformant TOE
1819 implements the functionality covered by the requirement, or as being "purely optional";
1820 and

1821 c) All optional requirements do not conflict with non-optional requirements (a capability
1822 cannot be both required and optional; however, a base capability can be required with
1823 enhancements to that capability being specified as optional).

1824 The evaluator determines that on all occasions where different security requirements apply to the
1825 same types of developer evidence, events, operations, data, tests to be performed etc. or to "all
1826 objects", "all subjects" etc., that these requirements do not conflict.

1827 Some possible conflicts are:

- 1828 a) an extended SAR specifying that the design of a certain cryptographic algorithm is to be
1829 kept secret, and another extended SAR specifying an open source review;
- 1830 b) **FAU_GEN.1 Audit data generation** specifying that subject identity is to be logged,
1831 **FDP_ACC.1 Subset access control** specifying who has access to these logs, and **FPR_UNO.1**
1832 **Unobservability** specifying that some actions of subjects should be unobservable to other
1833 subjects. If the subject that should not be able to see an activity may access logs of this
1834 activity, these SFRs conflict;
- 1835 c) **FDP_RIP.1 Subset residual information protection** specifying deletion of information no
1836 longer needed, and **FDP_ROL.1 Basic rollback** specifying that a TOE may return to a
1837 previous state. If the information that is needed for the rollback to the previous state has
1838 been deleted, these requirements conflict;
- 1839 d) Multiple iterations of **FDP_ACC.1 Subset access control** especially where some iterations
1840 cover the same subjects, objects, or operations. If one access control SFR allows a subject
1841 to perform an operation on an object, while another access control SFR does not allow
1842 this, these requirements conflict.

1843 **8.8.2 Evaluation of sub-activity (APE_REQ.2)**

1844 **8.8.2.1 Objectives**

1845 The objective of this sub-activity is to determine whether the SFRs and SARs are clear,
1846 unambiguous and well-defined, whether they are internally consistent, and whether the SFRs meet
1847 the security objectives of the TOE.

1848 **8.8.2.2 Input**

1849 The evaluation evidence for this sub-activity is:

- 1850 a) the PP.

1851 **8.8.2.3 Action APE_REQ.2.1E**

1852 ISO/IEC 15408-3 APE_REQ.2.1C: *The statement of security requirements shall describe the SFRs and*
1853 *the SARs.*

1854 **8.8.2.3.1 Work unit APE_REQ.2-1**

1855 The evaluator **shall check** that the statement of security requirements describes the SFRs.

1856 The evaluator determines that each SFR is identified by one of the following means:

- 1857 a) by reference to an individual component in ISO/IEC 15408-2;
- 1858 b) by reference to an extended component in the extended components definition of the PP;
- 1859 c) by reference to an individual component in a PP that the PP claims to be conformant with,
1860 including any optional requirements defined in the PP;
- 1861 d) by reference to an individual component in a security requirements package that the PP
1862 claims to be conformant with;
- 1863 e) by reproduction in the PP.

1864 It is not required to use the same means of identification for all SFRs.

1865 **8.8.2.3.2 Work unit APE_REQ.2-2**

1866 The evaluator **shall check** that the statement of security requirements describes the SARs.

1867 The evaluator determines that each SAR is identified by one of the following means:

- 1868 a) by reference to an individual component in ISO/IEC 15408-3;
- 1869 b) by reference to an extended component in the extended components definition of the PP;
- 1870 c) by reference to an individual component in a PP that the PP claims to be conformant with;
- 1871 d) by reference to an individual component in a security requirements package that the PP
- 1872 claims to be conformant with;
- 1873 e) by reproduction in the PP.

1874 It is not required to use the same means of identification for all SARs.

1875 Note that if optional requirements are defined by the PP, there may be associated threats that are

1876 covered by this work unit.

1877 ISO/IEC 15408-3 APE_REQ.2.2C: *All subjects, objects, operations, security attributes, external entities*

1878 *and other terms that are used in the SFRs and the SARs shall be defined.*

1879 **8.8.2.3.3 Work unit APE_REQ.2-3**

1880 The evaluator **shall examine** the PP to determine that all subjects, objects, operations, security

1881 attributes, external entities and other terms that are used in the SFRs and the SARs are defined.

1882 The evaluator determines that the PP defines all:

- 1883 — (types of) subjects and objects that are used in the SFRs;
- 1884 — (types of) security attributes of subjects, users, objects, information, sessions and/or resources,
- 1885 possible values that these attributes may take and any relations between these values (e.g.
- 1886 top_secret is “higher” than secret);
- 1887 — (types of) operations that are used in the SFRs, including the effects of these operations;
- 1888 — (types of) external entities in the SFRs;
- 1889 — other terms that are introduced in the SFRs and/or SARs by completing operations, if these
- 1890 terms are not immediately clear, or are used outside their dictionary definition.

1891 The goal of this work unit is to ensure that the SFRs and SARs are well-defined and that no

1892 misunderstanding may occur due to the introduction of vague terms. This work unit should not be

1893 taken into extremes, by forcing the PP author to define every single word. The general audience of

1894 a set of security requirements should be assumed to have a reasonable knowledge of IT, security

1895 and “Evaluation criteria for IT security”.

1896 All of the above may be presented in groups, classes, roles, types or other groupings or

1897 characterisations that allow easy understanding.

1898 The evaluator is reminded that these lists and definitions do not have to be part of the statement of

1899 security requirements, but may be placed (in part or in whole) in different subclauses. This may be

1900 especially applicable if the same terms are used in the rest of the PP.

- 1901 ISO/IEC 15408-3 APE_REQ.2.3C: *The statement of security requirements shall identify all operations*
1902 *on the security requirements.*
- 1903 **8.8.2.3.4 Work unit APE_REQ.2-4**
- 1904 The evaluator ***shall check*** that the statement of security requirements identifies all operations on
1905 the security requirements.
- 1906 The evaluator determines that all operations are identified in each SFR or SAR where such an
1907 operation is used. This includes both completed operations and uncompleted operations.
1908 Identification may be achieved by typographical distinctions, or by explicit identification in the
1909 surrounding text, or by any other distinctive means.
- 1910 **If the PP defines *selection-based* SFRs, the evaluator determines that the PP clearly identifies the**
1911 **dependencies between the selection in an SFR and the selection-based SFR(s) to be included in the**
1912 **PP/ST should that selection be chosen by the PP/ST author.**
- 1913 ISO/IEC 15408-3 APE_REQ.2.4C: *All operations shall be performed correctly.*
- 1914 **8.8.2.3.5 Work unit APE_REQ.2-5**
- 1915 The evaluator ***shall examine*** the statement of security requirements to determine that all
1916 assignment operations are performed correctly.
- 1917 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
1918 **Guidance for Operations.**
- 1919 **8.8.2.3.6 Work unit APE_REQ.2-6**
- 1920 The evaluator ***shall examine*** the statement of security requirements to determine that all iteration
1921 operations are performed correctly.
- 1922 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
1923 **Guidance for Operations.**
- 1924 **8.8.2.3.7 Work unit APE_REQ.2-7**
- 1925 The evaluator ***shall examine*** the statement of security requirements to determine that all selection
1926 operations are performed correctly.
- 1927 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
1928 **Guidance for Operations.**
- 1929 **8.8.2.3.8 Work unit APE_REQ.2-8**
- 1930 The evaluator ***shall examine*** the statement of security requirements to determine that all
1931 refinement operations are performed correctly.
- 1932 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
1933 **Guidance for Operations.**
- 1934 ISO/IEC 15408-3 APE_REQ.2.5C: *Each dependency of the security requirements shall either be*
1935 *satisfied, or the security requirements rationale shall justify the dependency not being satisfied.*

1936 **8.8.2.3.9 Work unit APE_REQ.2-9**

1937 The evaluator **shall examine** the statement of security requirements to determine that each
1938 dependency of the security requirements is either satisfied, or that the security requirements
1939 rationale justifies the dependency not being satisfied.

1940 A dependency is satisfied by the inclusion of the relevant component (or one that is hierarchical to
1941 it) within the statement of security requirements. The component used to satisfy the dependency
1942 should, if necessary, be modified by operations to ensure that it actually satisfies that dependency.

1943 A justification that a dependency is not met should address either:

1944 a) why the dependency is not necessary or useful, in which case no further information is
1945 required; or

1946 b) that the dependency has been addressed by the operational environment of the TOE, in
1947 which case the justification should describe how the security objectives for the
1948 operational environment address this dependency.

1949 ISO/IEC 15408-3 APE_REQ.2.6C: *The security requirements rationale shall trace each SFR back to*
1950 *the security objectives for the TOE.*

1951 **8.8.2.3.10 Work unit APE_REQ.2-10**

1952 The evaluator **shall check** that the security requirements rationale traces each SFR back to the
1953 security objectives for the TOE.

1954 Optional requirements may require Threats/OSPs to be specified, and security objectives
1955 associated with these SPD elements are also covered by this work unit.

1956 The evaluator determines that each SFR is traced back to at least one security objective for the TOE.

1957 Failure to trace implies that either the security requirements rationale is incomplete, the security
1958 objectives for the TOE are incomplete, or the SFR has no useful purpose.

1959 ISO/IEC 15408-3 APE_REQ.2.7C: *The security requirements rationale shall demonstrate that the*
1960 *SFRs meet all security objectives for the TOE.*

1961 **8.8.2.3.11 Work unit APE_REQ.2-11**

1962 The evaluator **shall examine** the security requirements rationale to determine that for each
1963 security objective for the TOE it justifies that the SFRs are suitable to meet that security objective
1964 for the TOE.

1965 If no SFRs trace back to the security objective for the TOE, the evaluator action related to this work
1966 unit is assigned a fail verdict.

1967 The evaluator determines that the justification for a security objective for the TOE demonstrates
1968 that the SFRs are sufficient: if all SFRs that trace back to the objective are satisfied, the security
1969 objective for the TOE is achieved.

1970 If the SFRs that trace back to a security objective for the TOE have any uncompleted assignments,
1971 or uncompleted or restricted selections, the evaluator determines that for every conceivable
1972 completion or combination of completions of these operations, the security objective is still met.

1973 The evaluator also determines that each SFR that traces back to a security objective for the TOE is
1974 necessary: when the SFR is satisfied, it actually contributes to achieving the security objective.

- 1975 Note that the tracings from SFRs to security objectives for the TOE provided in the security
 1976 requirements rationale may be a part of the justification, but do not constitute a justification by
 1977 themselves.
- 1978 ISO/IEC 15408-3 APE_REQ.2.8C: *The security requirements rationale shall explain why the SARs*
 1979 *were chosen.*
- 1980 **8.8.2.3.12 Work unit APE_REQ.2-12**
- 1981 The evaluator **shall check** that the security requirements rationale explains why the SARs were
 1982 chosen.
- 1983 The evaluator is reminded that any explanation is correct, as long as it is coherent and neither the
 1984 SARs nor the explanation have obvious inconsistencies with the remainder of the PP.
- 1985 An example of an obvious inconsistency between the SARs and the remainder of the PP would be to
 1986 have threat agents that are very capable, but an AVA_VAN SAR that does not protect against these
 1987 threat agents.
- 1988 ISO/IEC 15408-3 APE_REQ.2.9C: *The statement of security requirements shall be internally*
 1989 *consistent.*
- 1990 **8.8.2.3.13 Work unit APE_REQ.2-13**
- 1991 The evaluator **shall examine** the statement of security requirements to determine that it is
 1992 internally consistent.
- 1993 The evaluator determines that the combined set of all SFRs and SARs is internally consistent. With
 1994 respect to optional requirements, the evaluator determines that:
- 1995 a) All optional requirements either trace to an SPD element that is itself not optional, or trace
 1996 to an SPD element that is clearly associated with that optional SFR;
- 1997 b) All optional requirements are clearly identified as being required if a conformant TOE
 1998 implements the functionality covered by the requirement, or as being “purely optional”;
 1999 and
- 2000 c) All optional requirements do not conflict with non-optional requirements (a capability
 2001 cannot be both required and optional; however, a base capability can be required with
 2002 enhancements to that capability being specified as optional).
- 2003
- 2004 The evaluator determines that on all occasions where different security requirements apply to the
 2005 same types of developer evidence, events, operations, data, tests to be performed etc. or to “all
 2006 objects”, “all subjects” etc., that these requirements do not conflict.
- 2007 Some possible conflicts are:
- 2008 c) an extended SAR specifying that the design of a certain cryptographic algorithm is to be
 2009 kept secret, and another extended SAR specifying an open source review;
- 2010 d) **FAU_GEN.1 Audit data generation** specifying that subject identity is to be logged,
 2011 **FDP_ACC.1 Subset access control** specifying who has access to these logs, and **FPR_UNO.1**
 2012 **Unobservability** specifying that some actions of subjects should be unobservable to other
 2013 subjects. If the subject that should not be able to see an activity may access logs of this
 2014 activity, these SFRs conflict;

2015 e) **FDP_RIP.1 Subset residual information protection** specifying deletion of information no
2016 longer needed, and **FDP_ROL.1 Basic rollback** specifying that a TOE may return to a
2017 previous state. If the information that is needed for the rollback to the previous state has
2018 been deleted, these requirements conflict;

2019 Multiple iterations of **FDP_ACC.1 Subset access control** especially where some iterations
2020 cover the same subjects, objects, or operations. If one access control SFR allows a subject
2021 to perform an operation on an object, while another access control SFR does not allow
2022 this, these requirements conflict.

2023 **9 Class ACE: Protection Profile Configuration evaluation**

2024 **9.1 Introduction**

2025 All Base-PP(s) referenced in the PP-Module must be evaluated before the evaluation of a PP-
2026 Configuration.

2027 One possibility for evaluating a PP-Configuration is to flatten/serialise all the components of the
2028 Base-PP(s) and PP-Modules composing the PP-Configuration, duplicating components as necessary,
2029 and evaluating the resulting PP as a standard PP.

2030 Another possibility for evaluation of a PP-Configuration composed of several PP-Modules proceeds
2031 PP-Module by PP-Module, iteratively. Considering a PP-Configuration composed of the Protection
2032 Profiles P_i and the PP-Modules M_j , evaluation of the PP-Configuration proceeds with the following
2033 steps, illustrated in Figure 6

2034 1) first evaluating independently all Protection Profiles P_i ;

2035 2) evaluating the PP-Configuration C_1 composed of the PP-Module M_1 with the Protection
2036 Profiles P_i ;

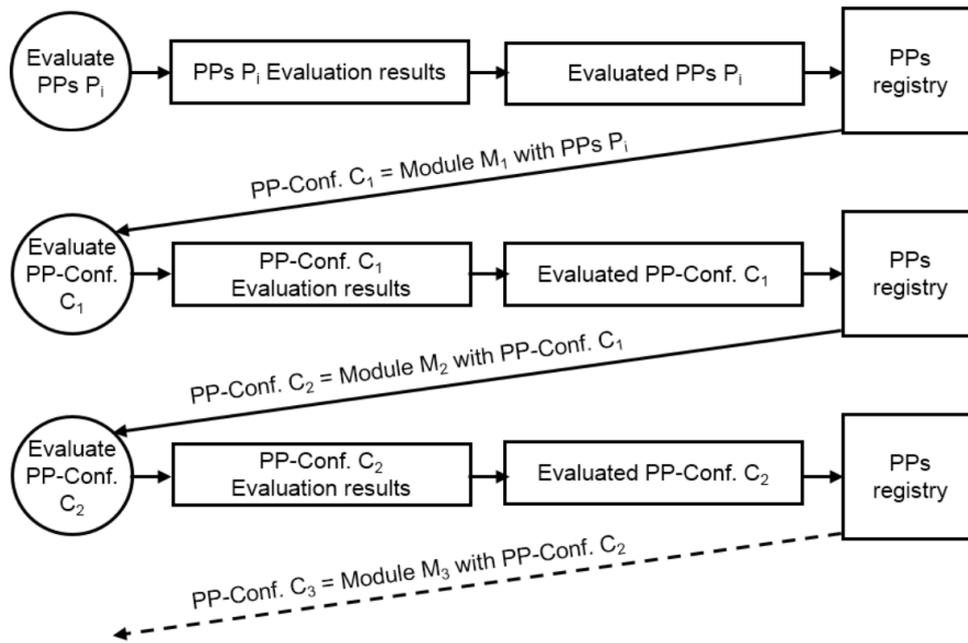
2037 3) evaluating the PP-Configuration C_{i+1} composed of the PP-Module M_{i+1} with the PP-
2038 Configuration C_i considered as a standard PP (cf. Section B.14 in ISO/IEC 15408-1);

2039 4) iterating the step 3) for all the PP-Modules

2040 Steps 2) and 3) are themselves performed in two steps:

2041 a) Evaluation of the PP-Module with its Base-PP(s) (Evaluation of sub-activity (ACE_MCO.1))

2042 b) Extension of the evaluation (consistency assessment) to the other elements of the PP-
2043 Configuration (Evaluation of sub-activity (ACE_CCO.1))



2044

Figure 6 - Evaluation of a PP-Configuration

2045

2046

2047 The ACE evaluation methodology is based on APE's. The common parts are prescribed in each
 2048 "Evaluation of sub-activity" in this document but referred to.

2049 **9.2 PP-Module introduction (ACE_INT)**

2050 **9.2.1 Evaluation of sub-activity (ACE_INT.1)**

2051 **9.2.1.1 Objectives**

2052 The objective of this sub-activity is to determine whether the PP-Module is correctly identified, and
 2053 whether the Base-PP(s) and TOE overview are consistent with each other.

2054 **9.2.1.2 Input**

2055 The evaluation evidence for this sub-activity is:

- 2056 a) the PP-Module;
- 2057 b) its Base-PP(s)

2058 **9.2.1.3 Application notes**

2059 All actions of APE_INT.1.1E hold.

2060 **9.2.1.4 Action ACE_INT.1.1E**

2061 ISO/IEC 15408-3 ACE_INT.1.1C *The PP-Module introduction shall uniquely identify all the Base-PPs*
 2062 *on which the PP-Module relies, including their logical structuring and relationship to the PP-Module*
 2063 *according to ISO/IEC 15408 Part 1, section B.13.3.2.*

2064 **9.2.1.4.1 Work unit ACE_INT.1-1**

2065 *The evaluator shall check that the PP-Module introduction identifies the Base-PP(s) on which the PP-*
2066 *Module relies.*

2067 ISO/IEC 15408-3 ACE_INT.1.2C *The TOE overview shall identify the differences introduced by the PP-*
2068 *Module with respect to the TOE overview of its Base-PP(s).*

2069 **9.2.1.4.2 Work unit ACE_INT.1-2**

2070 The evaluator *shall check* that the TOE overview identifies the differences introduced by the PP-
2071 Module with respect to the TOE overview of its Base-PP(s).

2072 **9.3 PP-Module conformance claims (ACE_CCL)**

2073 **9.3.1 Evaluation of sub-activity (ACE_CCL.1)**

2074 **9.3.1.1 Objectives**

2075 The objective of this sub-activity is to determine the validity of various conformance claims. These
2076 describe how the PP-Module conforms to the ISO/IEC 15408 Part 2 and SFR packages.

2077 **9.3.1.2 Input**

2078 The evaluation evidence for this sub-activity is:

2079 a) the PP-Module;

2080 b) the SFR package(s) that the PP claims conformance to:

2081 c) the PP-Configuration.

2082 **9.3.1.3 Action ACE_CCL.1.1E**

2083 ISO/IEC 15408-3 ACE_CCL.1.1C *The conformance claim shall contain a ISO/IEC 15408*
2084 *conformance claim that identifies the version of the ISO/IEC 15408 to which the PP-Module*
2085 *claims conformance.*

2086 **9.3.1.3.1 Work unit ACE_CCL.1-1**

2087 The evaluator *shall check* that the conformance claim contains a ISO/IEC 15408 conformance
2088 claim that identifies the version of the ISO/IEC 15408 to which the PP-Module claims
2089 conformance.

2090 The evaluator determines that the ISO/IEC 15408 conformance claim identifies the version of
2091 the ISO/IEC 15408 that was used to develop this PP-Module. This should include the version
2092 number of the ISO/IEC 15408 and, unless the International English version of the ISO/IEC
2093 15408 was used, the language of the version of the ISO/IEC 15408 that was used.

2094 ISO/IEC 15408-3 ACE_CCL.1.2C *The ISO/IEC 15408 conformance claim shall describe the*
2095 *conformance of the PP-Module to ISO/IEC 15408 Part 2 as either ISO/IEC 15408 Part 2*
2096 *conformant or ISO/IEC 15408 Part 2 extended.*

2097 **9.3.1.3.2 Work unit ACE_CCL.1-2**

2098 The evaluator *shall check* that the ISO/IEC 15408 conformance claim states a claim of either
2099 ISO/IEC 15408 Part 2 conformant or ISO/IEC 15408 Part 2 extended for the PP-Module.

2100 ISO/IEC 15408-3 ACE_CCL.1.3C *The conformance claim shall identify all security functional*
2101 *requirement packages to which the PP-Module claims conformance.*

2102 **9.3.1.3.3 Work unit ACE_CCL.1-3**

2103 The evaluator **shall check** that, for each identified package, the conformance claim contains a
2104 package claim that identifies all security functional requirement packages to which the PP-
2105 Module claims conformance.

2106 If the PP-Module does not claim conformance to a security functional requirement package,
2107 this work unit is not applicable and therefore considered to be satisfied.

2108 The evaluator determines that any referenced security functional requirement packages are
2109 unambiguously identified (e.g. by title and version number, or by the identification included in
2110 the introduction of that security functional requirement package).

2111 The evaluator is reminded that claims of partial conformance to a security functional
2112 requirement package are not permitted.

2113 ISO/IEC 15408-3 ACE_CCL.1.4C *The ISO/IEC 15408 conformance claim shall be consistent with*
2114 *the extended components definition.*

2115 **9.3.1.3.4 Work unit ACE_CCL.1-4**

2116 The evaluator **shall examine** the ISO/IEC 15408 conformance claim for ISO/IEC 15408 Part 2
2117 to determine that it is consistent with the extended components definition

2118 If the ISO/IEC 15408 conformance claim contains ISO/IEC 15408 Part 2 conformant, the
2119 evaluator determines that the extended components definition does not define functional
2120 components.

2121 If the ISO/IEC 15408 conformance claim contains ISO/IEC 15408 Part 2 extended, the
2122 evaluator determines that the extended components definition defines at least one extended
2123 functional component.

2124 ISO/IEC 15408-3 ACE_CCL.1.5C *The conformance statement shall identify other PP-modules (if*
2125 *any) and PPs (that are not Base-PPs for the PP-Module under evaluation) that, in combination*
2126 *with the module under evaluation, can be used in a PP-configuration.*

2127 **9.3.1.3.5 Work unit ACE_CCL.1-5**

2128 The evaluator **shall check** the conformance statement to determine that it lists the set of other
2129 PP-modules that can be specified in the components statement of a PP-configuration that
2130 includes the PP-module.

2131 If no PPs in the PP-Configuration's component statement require exact conformance in their
2132 conformance statements then this work unit does not apply and is therefore considered
2133 satisfied.

2134 If the PP-module does not allow its use (in a PP-configuration) with other PP-modules, then
2135 there will be no other PP-modules identified in the PP-module's conformance statement, and
2136 the evaluator ensures the PP-configuration contains no other PP-modules in the PP-
2137 configuration's components statement.

2138 If the PP-configuration's components statement does include other PP-modules, then the
2139 evaluator ensures that all PP-modules listed in the PP-configuration's components statement
2140 are identified as allowed with the PP-module in its conformance statement.

2141 **9.3.1.3.6 Work unit ACE_CCL.1-6**

2142 The evaluator shall check the conformance statement to determine that it lists PPs identified
2143 in the PP-Configuration's component statements that are not included in the PP-Module's set
2144 of Base-PPs as identified in the PP-Configuration's component statements.

2145 If a PP in the PP-Configuration's component statement does not require exact conformance in
2146 its conformance statement, this work unit does not apply and is therefore considered satisfied.

2147 If PP-Module does not identify (in its conformance statement) any PPs other than those that
2148 make up the set of Base-PPs for the PP-Module identified in the PP-Configuration's component
2149 statement, the evaluator ensures the PP-configuration contains no other (non-Base-) PPs in
2150 the PP-configuration's components statement.

2151 If the PP-configuration's components statement does include PPs that are not part of the PP-
2152 Module's set of Base-PPs, then the evaluator ensures that all such PPs listed in the PP-
2153 configuration's components statement are identified as allowed with the PP-Module in its
2154 conformance statement.

2155 ISO/IEC 15408-3 ACE_CCL.1.6C *The conformance claim shall describe any conformance of the*
2156 *PP to a package as either package-conformant or package-augmented.*

2157 **9.3.1.3.7 Work unit ACE_CCL.1-7**

2158 The evaluator **shall check** that, for each identified package, the conformance claim states a claim of
2159 either package-name conformant or package-name augmented.

2160 If the PP-Module does not claim conformance to a package, this work unit is not applicable and
2161 therefore considered to be satisfied. PP-Modules can only claim conformance to functional
2162 packages and therefore only this type of package is considered in the description below.

2163 If the functional package conformance claim contains package-name conformant, the evaluator
2164 determines that all assumptions, threats, OSPs, security objectives and SFRs included in the
2165 package are included in identical form by the PP-Module (including via its base-PP(s)).

2166 If the functional package conformance claim contains package-name augmented, the evaluator
2167 determines that all all assumptions, threats, OSPs, security objectives and SFRs included in the
2168 package are included in identical form by the PP-Module except that the PP-Module shall have at
2169 least one additional SFR or one SFR that is hierarchically higher than an SFR in the functional
2170 package.

2171 **9.4 PP-Module Security problem definition (ACE_SPD)**

2172 **9.4.1 Evaluation of sub-activity (ACE_SPD.1)**

2173 **9.4.1.1 Application notes**

2174 All actions of APE_SPD.1.1E hold.

2175 **9.5 PP-Module Security objectives (ACE_OBJ)**

2176 **9.5.1 Evaluation of sub-activity (ACE_OBJ.1)**

2177 **9.5.1.1 Application notes**

2178 If the PP-Configuration uses the Direct Rationale approach (as determined in ACE_CCO.1-2)
2179 then all actions of APE_OBJ.1.1E hold.

- 2180 **9.5.2 Evaluation of sub-activity (ACE_OBJ.2)**
- 2181 **9.5.2.1 Application notes**
- 2182 All actions of APE_OBJ.2.1E hold.
- 2183 **9.6 PP-Module extended components definition (ACE_ECD)**
- 2184 **9.6.1 Evaluation of sub-activity (ACE_ECD.1)**
- 2185 **9.6.1.1 Application notes**
- 2186 All actions of APE_ECD.1.1E and APE_ECD.1.2E hold.
- 2187 **9.7 PP-Module security requirements (ACE_REQ)**
- 2188 **9.7.1 Evaluation of sub-activity (ACE_REQ.1)**
- 2189 **9.7.1.1 Application notes**
- 2190 If the PP-Configuration uses the Direct Rationale approach (as determined in ACE_CCO.1-2)
- 2191 then all actions of APE_REQ.1.1E hold. The SAR part is not considered because it is empty in
- 2192 PP-Modules.
- 2193 **9.7.2 Evaluation of sub-activity (ACE_REQ.1)**
- 2194 **9.7.2.1 Application notes**
- 2195 All actions of APE_REQ.2.1E hold. The SAR part is not considered because it is empty in PP-
- 2196 Modules.
- 2197 **9.8 PP-Module consistency (ACE_MCO)**
- 2198 **9.8.1 Evaluation of sub-activity (ACE_MCO.1)**
- 2199 **9.8.1.1 Objectives**
- 2200 The objective of this sub-activity is to determine the consistency of the PP-Module regarding
- 2201 its Base-PP(s).
- 2202 **9.8.1.2 Input**
- 2203 The evaluation evidence for this sub-activity is:
- 2204 a) the PP-Module;
- 2205 b) its Base-PP(s)
- 2206 **9.8.1.3 Action ACE_MCO.1.1E**
- 2207 ISO/IEC 15408-3 ACE_MCO.1.1C *The consistency rationale shall demonstrate that the TOE type*
- 2208 *of the PP-Module is consistent with the TOE type(s) in the Base-PPs identified in the PP-Module*
- 2209 *introduction.*
- 2210 **9.8.1.3.1 Work unit ACE_MCO.1-1**
- 2211 The evaluator ***shall examine*** the consistency rationale to determine that the TOE type of the
- 2212 PP-Module is consistent with all the TOE types of the Base-PP(s).

2213 The relation between the types may be simple: a PP-Module may consider a TOE that provides
2214 additional security functionality, or more complex: a TOE that provides a given security
2215 functionality in a specific way.

2216 ISO/IEC 15408-3 ACE_MCO.1.2C The consistency rationale shall demonstrate that the
2217 statement of the security problem definition is consistent with the statement of the security
2218 problem definition in the Base-PPs identified in the PP-Module introduction.

2219 **9.8.1.3.2 Work unit ACE_MCO.1-2**

2220 The evaluator *shall examine* the PP-Module consistency rationale to determine that it
2221 demonstrates that the statement of security problem definition of the PP-Module is consistent
2222 with the statements of security problem definition stated in its Base-PPs.

2223 In particular, the evaluator examines the consistency rationale to determine that:

2224 a) the statements of threats, assumptions and OSPs in the PP-Module do not contradict those
2225 from the Base-PP(s).

2226 b) the statement of assumptions in the PP-Module addresses aspects out of scope of the
2227 Base-PP, in which case, the addition of elements is allowed.

2228 ISO/IEC 15408-3 ACE_MCO.1.3C *The consistency rationale shall demonstrate that the*
2229 *statement of security objectives is consistent with the statement of security objectives in the*
2230 *Base-PPs identified in the PP-Module introduction.*

2231 **9.8.1.3.3 Work unit ACE_MCO.1-3**

2232 The evaluator *shall examine* the PP-Module consistency rationale to determine that it
2233 demonstrates that the statement of security objectives of the PP-Module is consistent with the
2234 statement of security objectives of its Base-PP(s).

2235 Where the PP-Module and its Base-PP(s) use the Direct Rationale approach then this work
2236 unit is trivially satisfied for the TOE objectives (because these are not included under the
2237 Direct Rationale approach). If *any* of the PP-Module or its Base-PPs use the Direct Rationale
2238 approach then the PP-Module *and all* of its Base-PPs must use the Direct Rationale approach,
2239 otherwise the evaluator action related to this work unit is assigned a fail verdict.

2240 In particular, the evaluator examines the consistency rationale to determine that:

2241 a) the statements of the security objectives for the TOE and the security objectives for the
2242 operational environment in the PP-Module do not contradict those from the Base-PPs.

2243 b) the statement of the security objectives for the operational environment in the PP-Module
2244 addresses aspects out of scope of the Base-PP, in which case, the addition of elements is
2245 allowed.

2246 ISO/IEC 15408-3 ACE_MCO.1.4C *The consistency rationale shall demonstrate that the*
2247 *statement of security requirements is consistent with the statement of security requirements in*
2248 *the Base-PPs identified in the PP-Module introduction.*

2249 **9.8.1.3.4 Work unit ACE_MCO.1-4**

2250 The evaluator *shall examine* the consistency rationale to determine that the statement of
2251 security requirements of the PP-Module is consistent with the statement of security
2252 requirements of its Base-PPs, that is, the SFRs of the PP-Module either complete or refine the
2253 SFRs of the Base-PP(s) and that no contradiction arises from the whole set of SFRs of the PP-
2254 Module and the Base-PP(s).

2255 **9.9 PP-Configuration consistency (ACE_CCO)**

2256 **9.9.1 Evaluation of sub-activity (ACE_CCO.1)**

2257 **9.9.1.1 Objectives**

2258 The objective of this sub-activity is to determine whether the PP-Configuration and its
2259 components are correctly identified.

2260 The objective of this sub-activity is also to determine the consistency of the PP-Configuration
2261 regarding the whole set of Protection Profiles and PP-Modules.

2262 For the consistency analysis required by this activity, the application notes of ISO/IEC 18045,
2263 Section 10.2.1 (Re-using the evaluation results of certified PPs), is applicable to determine
2264 which parts of the Base-PPs are to be re-evaluated during the evaluation of PP-Configuration.

2265 **9.9.1.2 Input**

2266 The evaluation evidence for this sub-activity is:

- 2267 a) the PP-Configuration reference;
- 2268 b) the PP-Configuration components statement;
- 2269 c) the PP(s) and PP-Modules identified in the components statement.

2270 **9.9.1.3 Action ACE_CCO.1.1E**

2271 ISO/IEC 15408-3 ACE_CCO.1.1C *The PP-Configuration reference shall uniquely identify the PP-*
2272 *Configuration.*

2273 **9.9.1.3.1 Work unit ACE_CCO.1-1**

2274 The evaluator shall examine the PP-Configuration reference to determine that it uniquely
2275 identifies the PP-Configuration.

2276 The evaluator determines that the PP-Configuration reference identifies the PP-Configuration
2277 itself, so that it may be easily distinguished from other PPs, PP-Configurations and PP-
2278 Modules, and that it also uniquely identifies each version of the PP-Configuration, e.g. by
2279 including a version number and/or a date of publication.

2280 The PP-Configuration should have some referencing system that is capable of supporting
2281 unique references (e.g. use of numbers, letters or dates).

2282 ISO/IEC 15408-3 ACE_CCO.1.2C The components statements shall uniquely identify the
2283 Protection Profiles and the PP-Modules that compose the PP-Configuration.

2284 **9.9.1.3.2 Work unit ACE_CCO.1-2**

2285 The evaluator shall examine the PP-Configuration components statement to determine that it
2286 uniquely identifies the Protection Profiles and PP-Modules contained in the PP-Configuration.

2287 The evaluator shall check that if *any* of the Base-PPs or PP-Modules in the PP-Configuration
2288 use the Direct Rationale Approach then *all* Base-PPs and PP-Modules in the PP-Configuration
2289 use the Direct Rationale approach.

2290 The Protection Profiles should have been certified and available for use in security targets.

2291 ISO/IEC 15408-3 ACE_CCO.1.3C *The conformance statement shall specify the required*
2292 *conformance to the PP-Configuration as one of exact, strict, or demonstrable. The conformance*
2293 *claim shall contain a ISO/IEC 15408 conformance claim that identifies the version of the ISO/IEC*
2294 *15408 to which the PP-Configuration and its underlying Protection Profiles and PP-Module claim*
2295 *conformance.*

2296 **9.9.1.3.3 Work unit ACE_CCO.1-3**

2297 The evaluator shall examine the PP-Configuration conformance statement to determine that it
2298 specifies the kind of conformance required: exact, strict, or demonstrable.

2299 The evaluator shall check that the conformance claim contains a ISO/IEC 15408 conformance
2300 claim that identifies the version of the ISO/IEC 15408 to which the PP-Configuration and its
2301 underlying Protection Profile(s) and PP-Module(s) claim conformance.

2302 The evaluator shall examine the PP-Configuration conformance claim to determine the
2303 compatibility between all ISO/IEC 15408 versions that are related to the PP-Configuration
2304 and its underlying Protection Profile(s) and PP-Module(s).

2305 **If at least one of the Protection Profiles identified in the PP-configuration components**
2306 **statement requires exact conformance, then the PP-configuration conformance statement**
2307 **shall also require exact conformance. If none of the PPs identified in the PP-configuration**
2308 **components statement requires exact conformance but** at least one of the Protection Profiles
2309 identified in the PP-Configuration components statement claims strict conformance, then the
2310 PP-Configuration conformance statement shall also require strict conformance also.

2311 ISO/IEC 15408 versions used in a PP-Configuration and its underlying Protection Profile(s)
2312 and PP-Module(s) have to be compatible. If compatibility is not obvious, guidance from the
2313 certification scheme should be asked.

2314 ISO/IEC 15408-3 ACE_CCO.1.4C The SAR statement shall specify the set of SAR or predefined
2315 EAL that applies to this PP-Configuration.

2316 **9.9.1.3.4 Work unit ACE_CCO.1-4**

2317 The evaluator shall examine the PP-Configuration SAR statement to determine that it specifies
2318 a well-formed package of assurance requirements drawn from ISO/IEC 15408-3. The SAR
2319 package can be built with components from ISO/IEC 15408-3 or can refer to a specific SAR
2320 package stated in one of the Protection Profiles composing the PP-Configuration.

2321 If the package comes from ISO/IEC 15408-3 then the evaluator shall check that it is well-
2322 formed: it is closed by dependencies or the SAR statements provide a sound discarding
2323 rationale.

2324 The evaluator shall check that the package of SAR of the PP-Configuration is consistent with
2325 respect to the SARs of each of the Protection Profiles contained in the PP-Configuration: for
2326 any SAR component in each of the Protection Profile, the PP-Configuration provides either the
2327 same component or a higher component in the family hierarchy. If the SAR component in the
2328 Protection Profile is a refinement of a standard component, then the correspondent SAR
2329 component in the PP-Configuration has to include these refinements. If two Protection
2330 Profiles refine the same SAR component, the evaluator shall check that the refinements are
2331 not contradictory and that the corresponding SAR component in the PP-Configuration meets
2332 both.

2333 ISO/IEC 15408-3 ACE_CCO.1.5C *The Base-PP(s) on which the PP-Modules relies shall belong to*
2334 *the Protection Profiles identified in the components statement of the PP-Configuration.*

2335 **9.9.1.3.5 Work unit ACE_CCO.1-5**

2336 The evaluator shall check that the Base-PP(s) of each PP-Module in the PP-Configuration are
2337 included in the set of Protection Profiles identified in the PP-Configuration's component
2338 statement. Where a PP-Module specifies alternative sets of Base-PP(s) then only one of these
2339 sets must be referred to in the PP-Configuration.

2340 ISO/IEC 15408-3 ACE_CCO.1.6C *The conformance statement of each Base-PPs and PP in the*
2341 *components statement of the PP-Configuration shall identify other PP-Modules and PPs that can*
2342 *be used in combination with the PP in a PP-Configuration.*

2343 **9.9.1.3.6 Work unit ACE_CCO.1-6**

2344 For each Protection Profile listed in the PP-Configuration's components statement, the
2345 evaluator shall check the PP's conformance statement to determine that all PP-modules
2346 specified in the PP-Configuration's components statement are listed as allowed to be used
2347 with that PP. If the PP-configuration does not require exact conformance in its conformance
2348 statement, this work unit does not apply and is therefore considered satisfied.

2349 The evaluator checks each PP in the PP-Configuration's components statement. For each PP,
2350 the evaluator determines that each PP-Module listed in the PP-Configuration's components
2351 statement is also listed in the PP's conformance statement as allowed to be used with that PP.

2352 **9.9.1.3.7 Work unit ACE_CCO.1-7**

2353 For each Protection Profile listed in the PP-Configuration's components statement, the
2354 evaluator shall check the PP's conformance statement to determine that all other PPs
2355 specified in the PP-Configuration's components statement are listed as allowed to be used
2356 with that PP.

2357 If the PP-Configuration does not require exact conformance in its conformance statement, this
2358 work unit does not apply and is therefore considered satisfied.

2359 If there is only one PP identified in the PP-Configuration's component statement, then this
2360 work unit does not apply and is therefore considered satisfied.

2361 **9.9.1.4 Action ACE_CCO.1.2E**

2362 **9.9.1.4.1 Work unit ACE_CCO.1-8**

2363 The evaluator shall check that the PP-Configuration made up of all the Protection Profiles and
2364 PP-Modules identified in the components statement of the PP-Configuration is consistent.
2365 That is, the evaluator shall check that no contradiction arises from the whole set of Protection
2366 Profiles and PP-Modules included in the PP-Configuration.

2367 The evaluator can organise this work in many ways; the actual organisation may depend on
2368 the will to derive evaluation results for more than one PP-Configuration at a time

2369 For instance, the evaluator can process in two steps as follows:

- 2370 a) Assess the consistency of the set of Protection Profiles composing the PP-Configuration,
2371 b) Then proceed with the assessment of the PP-Configuration consistency incrementally, by
2372 adding one PP-Module at a time.

2373 An alternative is to proceed incrementally but mixing PPs and PP-Modules or to
2374 flatten/serialise the definition of the PP-Configuration (cf. Annex B in ISO/IEC 15408-1),
2375 duplicating as required, and to assess the consistency of the whole set of elements.

2376 Any incremental consistency analysis step where C is a subset of the PP-Configuration and X is
2377 a PP or a PP-Module that has to be added to C consists in:

- 2378 • assessing that the SPD, the objectives and the SFRs of X do not contradict the statements in
2379 C;
- 2380 • the assumptions and objectives for the environment in X either are the same as in C or
2381 address security aspects that are out of the scope of C.

2382 If the PP-Configuration is a Direct Rationale PP-Configuration (as determined in ACE_CCO.1-2)
2383 then the TOE objectives are not required in the consistency analysis.

2384 Note that if X is a PP-Module, C contains all its Base-PP(s) and Evaluation of sub-activity
2385 (ACE_MCO.1) has succeed for X, then the consistency analysis step has to be performed with
2386 respect to the components of C different from these Base-PP(s) only.

2387 **10 Class ASE: Security Target evaluation**

2388 **10.1 Introduction**

2389 This Clause describes the evaluation of an ST. The ST evaluation should be started prior to any TOE
2390 evaluation sub-activities since the ST provides the basis and context to perform these sub-activities.
2391 The evaluation methodology in this subclause is based on the requirements on the ST as specified
2392 in ISO/IEC 15408-3 class ASE.

2393 This Clause should be used in conjunction with Annexes **A**, **B** and **C, Guidance for Operations** in
2394 ISO/IEC 15408-1, as these Annexes clarify the concepts here and provide many examples.

2395 **10.2 Application notes**

2396 **10.2.1 Re-using the evaluation results of certified PPs**

2397 While evaluating an ST that is based on one or more certified PPs, it may be possible to re-use the
2398 fact that these PPs were certified. The potential for re-use of the result of a certified PP is greater if
2399 the ST does not add threats, OSPs, assumptions, security objectives and/or security requirements
2400 to those of the PP. If the ST contains much more than the certified PP, re-use may not be useful at
2401 all.

2402 The evaluator is allowed to re-use the PP evaluation results by doing certain analyses only partially
2403 or not at all if these analyses or parts thereof were already done as part of the PP evaluation. While
2404 doing this, the evaluator should assume that the analyses in the PP were performed correctly.

2405 An example would be where the PP contains a set of security requirements, and these were
2406 determined to be internally consistent during the PP evaluation. If the ST uses the exact same
2407 requirements, the consistency analysis does not have to be repeated during the ST evaluation. If
2408 the ST adds one or more requirements, or performs operations on these requirements, the analysis
2409 will have to be repeated. However, it may be possible to save work in this consistency analysis by
2410 using the fact that the original requirements are internally consistent. If the original requirements
2411 are internally consistent, the evaluator only has to determine that:

2412 a) the set of all new and/or changed requirements is internally consistent, and

2413 b) the set of all new and/or changed requirements is consistent with the original
2414 requirements.

2415 The evaluator notes in the ETR each case where analyses are not done or only partially done for
2416 this reason.

2417 The same re-use discussion applies to an ST claiming conformance to a certified PP-Configuration.

2418 **10.3 ST introduction (ASE_INT)**

2419 **10.3.1 Evaluation of sub-activity (ASE_INT.1)**

2420 **10.3.1.1 Objectives**

2421 The objective of this sub-activity is to determine whether the ST and the TOE are correctly
2422 identified, whether the TOE is correctly described in a narrative way at three levels of abstraction
2423 (TOE reference, TOE overview and TOE description), and whether these three descriptions are
2424 consistent with each other.

2425 **10.3.1.2 Input**

2426 The evaluation evidence for this sub-activity is:

2427 a) the ST.

2428 **10.3.1.3 Action ASE_INT.1.1E**

2429 ISO/IEC 15408-3 ASE_INT.1.1C: *The ST introduction shall contain an ST reference, a TOE reference, a*
2430 *TOE overview and a TOE description.*

2431 **10.3.1.3.1 Work unit ASE_INT.1-1**

2432 The evaluator **shall check** that the ST introduction contains an ST reference, a TOE reference, a
2433 TOE overview and a TOE description.

2434 ISO/IEC 15408-3 ASE_INT.1.2C: *The ST reference shall uniquely identify the ST.*

2435 **10.3.1.3.2 Work unit ASE_INT.1-2**

2436 The evaluator **shall examine** the ST reference to determine that it uniquely identifies the ST.

2437 The evaluator determines that the ST reference identifies the ST itself, so that it may be easily
2438 distinguished from other STs, and that it also uniquely identifies each version of the ST, e.g. by
2439 including a version number and/or a date of publication.

2440 In evaluations where a CM system is provided, the evaluator may validate the uniqueness of the
2441 reference by checking the configuration list. In the other cases, the ST should have some
2442 referencing system that is capable of supporting unique references (e.g. use of numbers, letters or
2443 dates).

2444 ISO/IEC 15408-3 ASE_INT.1.3C: *The TOE reference shall uniquely identify the TOE.*

2445 **10.3.1.3.3 Work unit ASE_INT.1-3**

2446 The evaluator **shall examine** the TOE reference to determine that it uniquely identifies the TOE.

2447 The evaluator determines that the TOE reference uniquely identifies the TOE, so that it is clear to
2448 which TOE the ST refers, and that it also identifies the version of the TOE, e.g. by including a
2449 version/release/build number, or a date of release.

2450 In the end of the evaluation, the evaluator **shall check** the TOE reference, and any unique
2451 identifiers associated with the TOE physical components are consistent with the identifier(s)
2452 assigned to the TOE evaluated in work units related to ALC_CMC.x.1C and the configuration list
2453 evaluated in work units related to ALC_CMS.x.2C.

2454 **10.3.1.3.4 Work unit ASE_INT.1-4**

2455 The evaluator *shall examine* the TOE reference to determine that it is not misleading.

2456 If the TOE is related to one or more well-known products, it is allowed to reflect this in the TOE
2457 reference. However, this should not be used to mislead consumers and it must be made clear which
2458 part of the product has been evaluated.

2459 When a TOE needs some required non-TOE hardware/software/firmware to run properly, the TOE
2460 reference may include the name of the non-TOE hardware/software/firmware used by the TOE,
2461 however it must be made clear that the non-TOE hardware/software/firmware has not been
2462 evaluated.

2463 ISO/IEC 15408-3 ASE_INT.1.4C: *The TOE overview shall summarise the usage and major security*
2464 *features of the TOE.*

2465 **10.3.1.3.5 Work unit ASE_INT.1-5**

2466 The evaluator *shall examine* the TOE overview to determine that it describes the usage and major
2467 security features of the TOE.

2468 The TOE overview may describe security features that are provided by the product, and/or those
2469 that users may expect in that product type, but it must clearly distinguish those features that are
2470 evaluated and those that are not evaluated.

2471 The TOE overview shall be consistent with information provided in other sections of the Security
2472 Target such as the TOE description, the security objectives, the security functional requirements,
2473 and the TOE summary specification. In addition to ensuring the evaluated security features are
2474 consistently described throughout the ST, this means that any security feature that is not evaluated
2475 is only discussed within the ST introduction, or else is explicitly identified as not evaluated in each
2476 other place where it is mentioned (failure to make this identification means that this work unit is
2477 assigned a fail verdict).

2478 The TOE overview in an ST for a composed TOE should describe the usage and major security
2479 feature of the composed TOE, rather than those of the individual component TOEs.

2480 The evaluator determines that the overview is clear enough for consumers, and sufficient to give
2481 them a general understanding of the intended usage and major security features of the TOE.

2482 ISO/IEC 15408-3 ASE_INT.1.5C: *The TOE overview shall identify the TOE type.*

2483 **10.3.1.3.6 Work unit ASE_INT.1-6**

2484 The evaluator *shall check* that the TOE overview identifies the TOE type.

2485 **10.3.1.3.7 Work unit ASE_INT.1-7**

2486 The evaluator *shall examine* the TOE overview to determine that the TOE type is not misleading.

2487 There are situations where the general consumer would expect certain functionality of the TOE
2488 because of its TOE type. If this functionality is absent in the TOE, the evaluator determines that the
2489 TOE overview adequately discusses this absence.

2490 There are also TOEs where the general consumer would expect that the TOE should be able to
2491 operate in a certain operational environment because of its TOE type. If the TOE is unable to
2492 operate in such an operational environment, the evaluator determines that the TOE overview
2493 adequately discusses this.

2494 ISO/IEC 15408-3 ASE_INT.1.6C: *The TOE overview shall identify any non-TOE*
2495 *hardware/software/firmware required by the TOE.*

2496 **10.3.1.3.8 Work unit ASE_INT.1-8**

2497 The evaluator **shall examine** the TOE overview to determine that it identifies any non-TOE
2498 hardware/software/firmware required by the TOE.

2499 While some TOEs are able to run stand-alone, other TOEs (notably software TOEs) need additional
2500 hardware, software or firmware to operate. If the TOE does not require any hardware, software or
2501 firmware, this work unit is not applicable and therefore considered to be satisfied.

2502 The evaluator determines that the TOE overview identifies any additional hardware, software and
2503 firmware needed by the TOE to operate. This identification does not have to be exhaustive, but
2504 detailed enough for potential consumers of the TOE to determine whether their current hardware,
2505 software and firmware support use of the TOE, and, if this is not the case, which additional
2506 hardware, software and/or firmware is needed.

2507 ISO/IEC 15408-3 ASE_INT.1.7C: *The TOE description shall describe the physical scope of the TOE.*

2508 **10.3.1.3.9 Work unit ASE_INT.1-9**

2509 The evaluator **shall examine** the TOE description to determine that it describes the physical scope
2510 of the TOE.

2511 The evaluator determines that the TOE description lists the hardware, firmware, software and
2512 guidance parts that constitute the TOE and describes them at a level of detail that is sufficient to
2513 give the reader a general understanding of those parts.

2514 As a minimum, the TOE description will cover the following elements:

2515 a) Each separately delivered part of the TOE, which will be identified by its unique identifier
2516 and the current format (binary, wafer, inlay, *.pdf, *.doc, *.chm etc.).

2517 b) The delivery method used by the developer to make available each part to the TOE
2518 consumer (Web site download, courier delivery, etc.)

2519 The physical description will also include some clear statements about the evaluated TOE
2520 configuration. In the case where a product could have multiple physical components, and therefore
2521 multiple configurations, the evaluated configurations must be briefly described and identified.

2522 The evaluator also determines that there is no possible misunderstanding as to whether any
2523 hardware, firmware, software or guidance part is part of the TOE or not.

2524 ISO/IEC 15408-3 ASE_INT.1.8C: *The TOE description shall describe the logical scope of the TOE.*

2525 **10.3.1.3.10 Work unit ASE_INT.1-10**

2526 The evaluator **shall examine** the TOE description to determine that it describes the logical scope of
2527 the TOE.

2528 The evaluator determines that the TOE description discusses the logical security features offered
2529 by the TOE at a level of detail that is sufficient to give the reader a general understanding of those
2530 features.

2531 The evaluator also determines that there is no possible misunderstanding as to whether any logical
2532 security feature is offered by the TOE or not.

2533 An ST for a composed TOE may refer out to the description of the logical scope of the component
2534 TOEs, provided in the component TOE STs to provide the majority of this description for the
2535 composed TOE. However, the evaluator determines that the composed TOE ST clearly discusses
2536 which features of the individual components are not within the composed TOE, and therefore not a
2537 feature of the composed TOE.

2538 **10.3.1.4 Action ASE_INT.1.2E**

2539 **10.3.1.4.1 Work unit ASE_INT.1-11**

2540 The evaluator *shall examine* the TOE reference, TOE overview and TOE description to determine
2541 that they are consistent with each other.

2542 **10.4 Conformance claims (ASE_CCL)**

2543 **10.4.1 Evaluation of sub-activity (ASE_CCL.1)**

2544 **10.4.1.1 Objectives**

2545 The objective of this sub-activity is to determine the validity of various conformance claims. These
2546 describe how the ST and the TOE conform to ISO/IEC 15408 and how the ST conforms to a PP-
2547 Configuration, PPs and packages.

2548 **10.4.1.2 Input**

2549 The evaluation evidence for this sub-activity is:

- 2550 a) the ST;
- 2551 b) the Base-PP(s) that the ST claims conformance to;
- 2552 c) the package(s) that the ST claims conformance to.

2553 **10.4.1.3 Action ASE_CCL.1.1E**

2554 ISO/IEC 15408-3 ASE_CCL.1.1C: *The conformance claim shall contain an ISO/IEC 15408*
2555 *conformance claim that identifies the edition of ISO/IEC 15408 to which the ST and the TOE claim*
2556 *conformance.*

2557 **10.4.1.3.1 Work unit ASE_CCL.1-1**

2558 The evaluator *shall check* that the conformance claim contains an ISO/IEC 15408 conformance
2559 claim that identifies the edition of ISO/IEC 15408 to which the ST and the TOE claim conformance.

2560 The evaluator determines that ISO/IEC 15408 conformance claim identifies the edition of ISO/IEC
2561 15408 that was used to develop this ST. This should include the version number of ISO/IEC 15408
2562 and, unless the International English version of ISO/IEC 15408 was used, the language of the
2563 edition of ISO/IEC 15408 that was used.

2564 For a composed TOE, the evaluator will consider any differences between the edition of ISO/IEC
2565 15408 claimed for a component and the edition of ISO/IEC 15408 claimed for the composed TOE. If
2566 the edition differ the evaluator will assess whether the differences between the versions will lead
2567 to conflicting claims.

2568 For instances where ISO/IEC 15408 conformance claims for the base TOE and dependent TOE are
2569 for different major releases of ISO/IEC 15408 (e.g. one component TOE conformance claim is
2570 ISO/IEC 15408 v2.x and the other component TOE conformance claim is ISO/IEC 15408 v3.x), the
2571 conformance claim for the composed TOE will be the earlier release of ISO/IEC 15408, as ISO/IEC

- 2572 15408 is developed with an aim to provide backwards compatibility (although this may not be
2573 achieved in the strictest sense, it is understood to be achieved in principle).
- 2574 ISO/IEC 15408-3 ASE_CCL.1.2C: *ISO/IEC 15408 conformance claim shall describe the conformance of*
2575 *the ST to ISO/IEC 15408-2 as either ISO/IEC 15408-2 conformant or ISO/IEC 15408-2 extended.*
- 2576 **10.4.1.3.2 Work unit ASE_CCL.1-2**
- 2577 The evaluator **shall check** that ISO/IEC 15408 conformance claim states a claim of either ISO/IEC
2578 15408-2 conformant or ISO/IEC 15408-2 extended for the ST.
- 2579 For a composed TOE, the evaluator will consider whether this claim is consistent not only with
2580 ISO/IEC 15408-2, but also with the claims of conformance to ISO/IEC 15408-2 by each of the
2581 component TOEs. I.e. if one or more component TOEs claims to be ISO/IEC 15408-2 extended, then
2582 the composed TOE should also claim to be ISO/IEC 15408-2 extended.
- 2583 ISO/IEC 15408 conformance claim for the composed TOE may be ISO/IEC 15408-2 extended, even
2584 though the component TOEs are ISO/IEC 15408-2 conformant, in the event that additional SFRs
2585 are claimed for the base TOE (see composed TOE guidance for **ASE_CCL.1.6C**)
- 2586 ISO/IEC 15408-3 ASE_CCL.1.3C: *ISO/IEC 15408 conformance claim shall describe the conformance of*
2587 *the ST to ISO/IEC 15408-3 as either ISO/IEC 15408-3 conformant or ISO/IEC 15408-3 extended.*
- 2588 **10.4.1.3.3 Work unit ASE_CCL.1-3**
- 2589 The evaluator **shall check** that ISO/IEC 15408 conformance claim states a claim of either ISO/IEC
2590 15408-3 conformant or ISO/IEC 15408-3 extended for the ST.
- 2591 ISO/IEC 15408-3 ASE_CCL.1.4C: *ISO/IEC 15408 conformance claim shall be consistent with the*
2592 *extended components definition.*
- 2593 **10.4.1.3.4 Work unit ASE_CCL.1-4**
- 2594 The evaluator **shall examine** ISO/IEC 15408 conformance claim for ISO/IEC 15408-2 to determine
2595 that it is consistent with the extended components definition.
- 2596 If ISO/IEC 15408 conformance claim contains ISO/IEC 15408-2 conformant, the evaluator
2597 determines that the extended components definition does not define functional components.
- 2598 If ISO/IEC 15408 conformance claim contains ISO/IEC 15408-2 extended, the evaluator determines
2599 that the extended components definition defines at least one extended functional component.
- 2600 **10.4.1.3.5 Work unit ASE_CCL.1-5**
- 2601 The evaluator **shall examine** ISO/IEC 15408 conformance claim for ISO/IEC 15408-3 to determine
2602 that it is consistent with the extended components definition.
- 2603 If ISO/IEC 15408 conformance claim contains ISO/IEC 15408-3 conformant, the evaluator
2604 determines that the extended components definition does not define assurance components.
- 2605 If ISO/IEC 15408 conformance claim contains ISO/IEC 15408-3 extended, the evaluator determines
2606 that the extended components definition defines at least one extended assurance component.
- 2607 ISO/IEC 15408-3 ASE_CCL.1.5C: *The conformance claim shall identify a PP-Configuration, or all PPs*
2608 *and security requirement packages to which the ST claims conformance.*

2609 **10.4.1.3.6 Work unit ASE_CCL.1-6**

2610 The evaluator **shall check** that the conformance claim contains a PP claim that identifies all PPs for
2611 which the ST claims conformance.

2612 If the ST does not claim conformance to a PP, this work unit is not applicable and therefore
2613 considered to be satisfied.

2614 The evaluator determines that any referenced PPs are unambiguously identified (e.g. by title and
2615 version number, or by the identification included in the introduction of that PP).

2616 For conformance claims to PPs containing functional packages, the evaluator examines that:

- 2617 - The ST does not include conformance claims to any PP that also claims conformance to any
2618 of the packages to which the ST is also claiming conformance. all dependencies between
2619 the selected packages have been resolved.

2620 The evaluator is reminded that claims of partial conformance to a PP are not permitted. Therefore,
2621 conformance to a PP requiring a composite solution may be claimed in an ST for a composed TOE.
2622 Conformance to such a PP would not have been possible during the evaluation of the component
2623 TOEs, as these components would not have satisfied the composed solution. This is only possible in
2624 the instances where the “composite” PP permits use of the composition evaluation approach (use
2625 of ACO components).

2626 For PPs containing functional packages, partial conformance means that not all packages have been
2627 included in the ST, a functional package has only been partially included into the ST, or a
2628 dependency requirement between functional packages has not been met. Note that exclusion of
2629 optional requirements that the ST either chooses not to, or is not required to, claim does not result
2630 in “partial conformance” to the PP, and so is allowed.

2631 **10.4.1.3.7 Work unit ASE_CCL.1-6a**

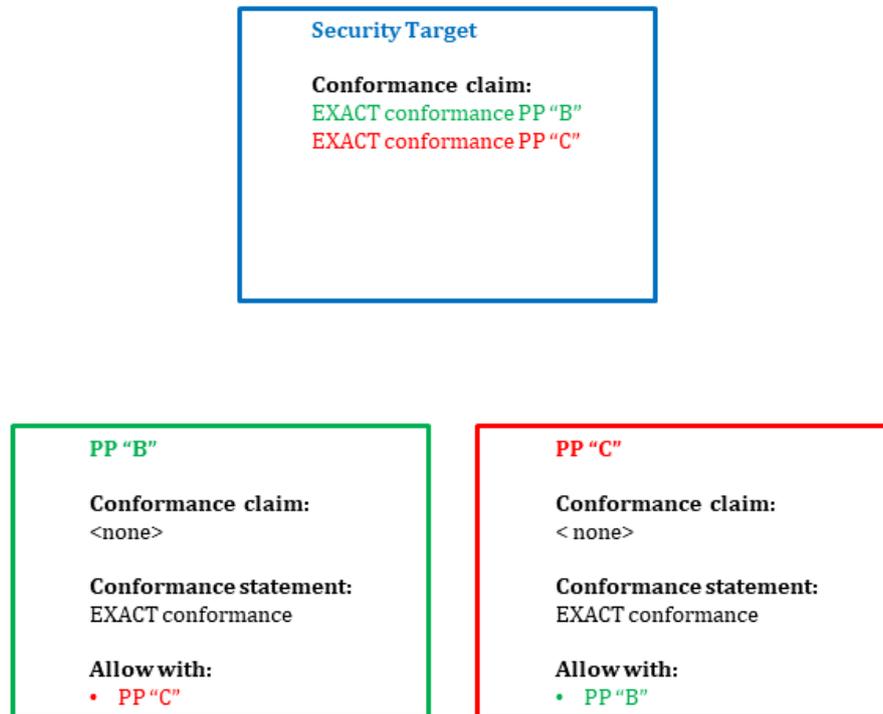
2632 The evaluator **shall check** that, for each PP to which the ST claims conformance, the conformance
2633 statement of that PP allows all other PPs in the conformance claim to be allowed to be claimed with
2634 that PP.

2635 If the ST does not claim conformance to a PP, or claims conformance to only one PP, this work unit
2636 is not applicable and therefore considered to be satisfied.

2637 If the ST is not claiming exact conformance to a PP, this work unit is not applicable and therefore
2638 considered to be satisfied.

2639 The evaluator determines that the conformance statement of the PP to which conformance is being
2640 claimed lists each of the PPs identified in the conformance claim section of the ST as being “allowed
2641 to be claimed with” that PP. Note that this is only applicable in cases where that PP requires exact
2642 conformance and the ST claims exact conformance.

2643 EXAMPLE consider the case where an ST is being evaluated and claims conformance to PPs B and
2644 C; this is depicted in Figure 7. The ST is claiming exact conformance, so all PPs require exact
2645 conformance in their conformance statements. Under this work unit, the evaluator determines that
2646 PP B lists (in its conformance statement) “PP C” as being a PP that can be claimed (by an ST) with
2647 PP B. Likewise, the evaluator determines that PP C lists (in its conformance statement) “PP B” as
2648 being a PP that can be claimed (by an ST) with PP C.



2649

2650 **Figure 7 — Example of exact conformance relationships between an ST and PPs**2651 **10.4.1.3.8 Work unit ASE_CCL.1-6b**

2652 The evaluator shall check that the conformance claim contains a PP-Configuration claim that
2653 identifies the PP-Configuration(s) for which the ST claims conformance.

2654 If the ST does not claim conformance to a PP-Configuration, this work unit is not applicable and
2655 therefore considered to be satisfied.

2656 If the ST claims conformance to multiple PP-Configurations, the evaluator ensures that the
2657 conformance statement for all PP-Configurations is either "strict" or "demonstrable"; an ST cannot
2658 claim exact conformance to multiple PP-Configurations

2659 If the ST claims conformance to a PP-Configuration and a PP (that is not part of the PP-
2660 Configuration), the evaluator ensures that the conformance statement for all PP-Configurations
2661 and the PP is either "strict" or "demonstrable"; an ST cannot claim exact conformance to a PP-
2662 Configurations and a PP that is not part of the PP-Configuration. The evaluator determines that any
2663 referenced PP-Configuration(s) are unambiguously identified (e.g. by title and version number, or
2664 by the identification included in the introduction of that PP).

2665 For conformance claims to PP-Configurations containing functional packages, the evaluator
2666 examines that:

2667 - all dependencies between the selected packages have been resolved.

2668 The evaluator is reminded that claims of partial conformance to a PP are not permitted. For PP-
2669 Configurations containing functional packages, partial conformance means that a functional
2670 package has only been partially included into the ST, or a dependency requirement between
2671 functional packages has not been met.

2672 **10.4.1.3.9 Work unit ASE_CCL.1-7**

2673 The evaluator *shall check* that the conformance claim contains a package claim that identifies all
2674 packages to which the ST claims conformance.

2675 If the ST does not claim conformance to a package, this work unit is not applicable and therefore
2676 considered to be satisfied.

2677 The evaluator determines that any packages to which the ST claims conformance are not also
2678 claimed conformance to by a PP, Base-PP, or PP-Module that the ST is claiming conformance to.

2679 The evaluator also determines that if the ST is claiming exact conformance to a PP or PP-
2680 Configuration, then no packages are claimed conformance to by the ST. The evaluator determines
2681 that the component TOE STs from which the composed TOE is derived are also unambiguously
2682 identified.

2683 The evaluator is reminded that claims of partial conformance to a package are not permitted.

2684 **10.4.1.3.10 Work unit ASE_CCL.1-8**

2685 The evaluator *shall check* that, for each identified package, the conformance claim states a claim of
2686 either package-name conformant or package-name augmented.

2687 If the ST claims conformance to a PP and the PP itself claims conformance to one or more
2688 functional packages then the ST shall not separately make a conformance claim to the same
2689 packages. If the ST does not claim conformance to a package, this work unit is not applicable and
2690 therefore considered to be satisfied.

2691 If the package conformance claim contains package-name conformant, the evaluator determines
2692 that:

2693 a) If the package is an assurance package, then the ST contains all SARs included in the
2694 package, but no additional SARs.

2695 b) If the package is a functional package, then all assumptions, threats, OSPs, security
2696 objectives and SFRs included in the package are identical to those included in the ST
2697 (after allowing any remaining iterations, refinements, assignments or selections from the
2698 package to be made in the ST).

2699 If the package conformance claim contains package-name augmented, the evaluator determines
2700 that:

2701 a) If the package is an assurance package then the ST contains all SARs included in the
2702 package, and at least one additional SAR or at least one SAR that is hierarchical to a SAR
2703 in the package.

2704 b) If the package is a functional package, then the constituent parts (security problem
2705 definition, security objectives, SFRs) of that ST contain all constituent parts (security
2706 problem definition, security objectives, SFRs) of that specific package, but additionally
2707 contain at least one enhancement of the security
2708 functionality defined by that specific package (finally resulting in an additional SFR or
2709 one an SFR that is hierarchically higher than an SFR in the package).

2710 The evaluator determines that, if the ST claims exact conformance to the PPs/PP-Configuration,
2711 only claims of <package name>-conformant are present.

- 2712 ISO/IEC 15408-3 ASE_CCL.1.7C: *The conformance claim rationale shall demonstrate that the TOE*
 2713 *type is consistent with the TOE type in the PP-Configuration or PPs for which conformance is being*
 2714 *claimed.*
- 2715 **10.4.1.3.11 Work unit ASE_CCL.1-9**
- 2716 In this work unit, the term “PP” shall be understood to mean “PP or PP-Configuration component”.
- 2717 The evaluator **shall examine** the conformance claim rationale to determine that the TOE type of
 2718 the TOE is consistent with all TOE types of the PPs.
- 2719 If the ST does not claim conformance to a PP, this work unit is not applicable and therefore
 2720 considered to be satisfied.
- 2721 The relation between the types may be simple: a firewall ST claiming conformance to a firewall PP,
 2722 or more complex: a smart card ST claiming conformance to a number of PPs at the same time (a PP
 2723 for the integrated circuit, a PP for the smart card OS, and two PPs for two applications on the smart
 2724 card).
- 2725 For a composed TOE, the evaluator will determine whether the conformance claim rationale
 2726 demonstrates that the TOE types of the component TOEs are consistent with the composed TOE
 2727 type. This does not mean that both the component and the composed TOE types have to be the
 2728 same, but rather that the component TOEs are suitable for integration to provide the composed
 2729 TOE. It should be made clear in the composed TOE ST which SFRs are only included as a result of
 2730 composition, and were not examined as SFRs in the base and dependent TOE (e.g. EALx) evaluation.
- 2731 ISO/IEC 15408-3 ASE_CCL.1.8C: *The conformance claim rationale shall demonstrate that the*
 2732 *statement of the security problem definition is consistent with the statement of the security problem*
 2733 *definition in the PP-Configuration or PPs for which conformance is being claimed.*
- 2734 **10.4.1.3.12 Work unit ASE_CCL.1-10**
- 2735 In this work unit, the term “PP” shall be understood to mean “PP or PP-Configuration component”.
- 2736 The evaluator **shall examine** the conformance claim rationale to determine that it demonstrates
 2737 that the statement of security problem definition is consistent, as defined by the conformance
 2738 statement of the PP, with the statements of security problem definition stated in the PPs to which
 2739 conformance is being claimed.
- 2740 If the ST does not claim conformance with a PP, this work unit is not applicable and therefore
 2741 considered to be satisfied.
- 2742 If the PP does not have a statement of security problem definition, this work unit is not applicable
 2743 and therefore considered to be satisfied.
- 2744 If the PP contains functional packages, the evaluator determines that the security problem
 2745 definition of the ST consists of all assumptions, threats and OSPs of all functional packages.
- 2746 If packages are used, the rules defined in the following paragraphs concerning exact, strict and
 2747 demonstrable conformance also hold for the SPD descriptions taken from the packages.
- 2748 If exact conformance is required by the PP to which conformance is being claimed, no conformance
 2749 claim rationale is required. Instead, the evaluator determines whether:
- 2750 a) the threats in the ST are identical (no fewer threats, no additional threats) to the threats in
 2751 the PP to which conformance is being claimed. If exact conformance is being claimed to
 2752 more than one PP, then the set of threats in the ST must be identical to the union of the
 2753 threats in all PPs to which conformance is being claimed.

2754 b) the OSPs in the ST are identical (no fewer OSPs, no additional OSPs) to the OSPs in the PP
2755 to which conformance is being claimed. If exact conformance is being claimed to more than
2756 one PP, then the set of OSPs in the ST must be identical to the union of the OSPs in all PPs
2757 to which conformance is being claimed.

2758 c) the assumptions in the ST are identical (no fewer assumptions, no additional assumptions)
2759 to the assumptions in the PP to which conformance is being claimed. If exact conformance
2760 is being claimed to more than one PP, then the set of assumptions in the ST must be
2761 identical to the union of the assumptions in all PPs to which conformance is being claimed,
2762 with the following possible exception;

2763 - an assumption (or part of an assumption) from a PP can be omitted, if all security
2764 objectives for the operational environment addressing this assumption (or part of an
2765 assumption) are replaced by security objectives for the TOE that are identical to
2766 (taken from) another of the PPs to which the ST is claiming conformance;

2767 When examining an ST in these circumstances (assumptions from one PP are replaced by security
2768 objectives on the TOE from one of the other PPs) the evaluator shall carefully determine that the
2769 condition given above is fulfilled. The following discussion gives an example:

2770 - EXAMPLE an ST is claiming exact conformance to two PPs. As determined in
2771 previous work units, both PPs require exact conformance in their conformance
2772 statements, and both PPs list the other as being "allowed with" the PP in a
2773 conformance claim by an ST. One PP to which the ST claims conformance contains an
2774 assumption stating that the operational environment prevents unauthorised
2775 modification or interception of data sent to an external interface of the TOE. This may
2776 be the case if the TOE accepts data in clear text and without integrity protection at
2777 this interface and is assumed to be located in a secure operational environment,
2778 which will prevent attackers from accessing this data. The assumption will then be
2779 mapped in the PP to some objective for the operational environment stating that the
2780 data interchanged at this interface are protected by adequate measures in the
2781 operational environment. Suppose there is another PP that specifies that conformant
2782 TOEs must protect data sent over the TOEs external interfaces, and has appropriate
2783 threats and security objectives addressing this threat. The ST author can then
2784 replace the assumption and security objective for the environment related to the
2785 protection of data over the external interfaces of the TOE from one PP with the
2786 security objective stating that the TOE itself protects these data, for example by
2787 providing a secure channel for encryption and integrity protection of all data
2788 transferred via this interface from the other PP; the corresponding objective and
2789 assumption for the operational environment from the other PP is thus omitted from
2790 the ST. This is also called re-assigning of the objective, since the objective is re-
2791 assigned from the operational environment to the TOE. Note, that this TOE is still
2792 secure in an operational environment fulfilling the omitted assumption and therefore
2793 still fulfils the PP. Further, the set of threats and objectives in the ST is still no
2794 broader than the union of threats and objectives in the PPs to which it is claiming
2795 exact conformance.

2796 If strict conformance is required by the PP to which conformance is being claimed no conformance
2797 claim rationale is required. Instead, the evaluator determines whether:

2798 a) the threats in the ST are a superset of or identical to the threats in the PP to which
2799 conformance is being claimed;

2800 b) the OSPs in the ST are a superset of or identical to the OSPs in the PP to which
2801 conformance is being claimed;

- 2802 c) the assumptions in the ST are identical to the assumptions in the PP to which
 2803 conformance is being claimed, with two possible exceptions described in the following
 2804 two bullet points;
- 2805 — an assumption (or part of an assumption) from the PP can be omitted, if all security objectives
 2806 for the operational environment addressing this assumption (or part of an assumption) are
 2807 replaced by security objectives for the TOE;
- 2808 — an assumption can be added to the assumptions defined in the PP, if a rationale is given, why
 2809 the new assumption neither mitigates a threat (or a part of a threat) meant to be addressed by
 2810 security objectives for the TOE in the PP, nor fulfils an OSP (or part of an OSP) meant to be
 2811 addressed by security objectives for the TOE in the PP.
- 2812 When examining an ST claiming a PP, which omits assumptions from the PP or adds new
 2813 assumptions, the evaluator shall carefully determine, if the conditions given above are fulfilled. The
 2814 following discussion gives some motivation and examples for these cases:
- 2815 — Example for omitting an assumption: A PP may contain an assumption stating that the
 2816 operational environment prevents unauthorised modification or interception of data sent to an
 2817 external interface of the TOE. This may be the case if the TOE accepts data in clear text and
 2818 without integrity protection at this interface and is assumed to be located in a secure
 2819 operational environment, which will prevent attackers from accessing these data. The
 2820 assumption will then be mapped in the PP to some objective for the operational environment
 2821 stating that the data interchanged at this interface are protected by adequate measures in the
 2822 operational environment. If an ST claiming this PP defines a more secure TOE, which has an
 2823 additional security objective stating that the TOE itself protects these data, for example by
 2824 providing a secure channel for encryption and integrity protection of all data transferred via
 2825 this interface, the corresponding objective and assumption for the operational environment
 2826 can be omitted from the ST. This is also called re-assigning of the objective, since the objective
 2827 is re-assigned from the operational environment to the TOE. Note, that this TOE is still secure
 2828 in an operational environment fulfilling the omitted assumption and therefore still fulfils the
 2829 PP.
- 2830 — Example for adding an assumption: In this example, the PP is designed to specify requirements
 2831 for a TOE of type "Firewall" and an ST author wishes to claim this PP for a TOE, which
 2832 implements a firewall, but additionally provides the functionality of a virtual private network
 2833 (VPN) component. For the VPN functionality, the TOE needs cryptographic keys and these keys
 2834 may also have to be handled securely by the operational environment (e. g. if symmetric keys
 2835 are used to secure the network connection and therefore need to be provided in some secure
 2836 way to other components in the network). In this case, it is acceptable to add an assumption
 2837 that the cryptographic keys used by the VPN are handled securely by the operational
 2838 environment. This assumption does not address threats or OSPs of the PP and therefore fulfils
 2839 the conditions stated above.
- 2840 — Counterexample for adding an assumption: In a variant of the first example a PP may already
 2841 contain an objective for the TOE to provide a secure channel for one of its interfaces, and this
 2842 objective is mapped to a threat of unauthorised modification or reading of the data on this
 2843 interface. In this case, it is clearly not allowed for an ST claiming this PP to add an assumption
 2844 for the operational environment, which assumes that the operational environment protects
 2845 data on this interface against modification or unauthorised reading of the data. This
 2846 assumption would reduce a threat, which is meant to be addressed by the TOE. Therefore a
 2847 TOE fulfilling an ST with this added assumption would not automatically fulfil the PP any more
 2848 and this addition is therefore not allowed.
- 2849 — Second counterexample for adding an assumption: In the example above of a TOE
 2850 implementing a firewall it would not be admissible to add a general assumption that the TOE is
 2851 only connected to trusted devices, because this would obviously remove essential threats

2852 relevant for a firewall (namely that there is untrusted IP traffic, which needs to be filtered).
2853 Therefore, this addition would not be allowed.

2854 If demonstrable conformance is required by the PP, the evaluator examines the conformance claim
2855 rationale to determine that it demonstrates that the statement of security problem definition of the
2856 ST is equivalent or more restrictive than the statement of security problem definition in the PP to
2857 which conformance is being claimed.

2858 For this, the conformance claim rationale needs to demonstrate that the security problem
2859 definition in the ST is equivalent (or more restrictive) than the security problem definition in the
2860 PP. This means that:

2861 — all TOEs that would meet the security problem definition in the ST also meet the security
2862 problem definition in the PP. This can also be shown indirectly by demonstrating that every
2863 event, which realises a threat defined in the PP or violates an OSP defined in the PP, would also
2864 realise a threat stated in the ST or violate an OSP defined in the ST. Note that fulfilling an OSP
2865 stated in the ST may avert a threat stated in the PP or that averting a threat stated in the ST
2866 may fulfil an OSP stated in the PP, so threats and OSPs can substitute each other;

2867 — all operational environments that would meet the security problem definition in the PP would
2868 also meet the security problem definition in the ST (with one exception in the next bullet);

2869 — besides a set of assumptions in the ST needed to demonstrate conformance to the SPD of the
2870 PP, an ST may specify further assumptions, but only if these additional assumptions are
2871 independent of and do not affect the security problem definition as defined in the PP. More
2872 detailed, there are no assumptions in the ST that exclude threats to the TOE that need to be
2873 countered by the TOE according to the PP. Similarly, there are no assumptions in the ST that
2874 realise aspects of an OSP stated in the PP, which are meant to be fulfilled by the TOE according
2875 to the PP."

2876 For a composed TOE, the evaluator will consider whether the security problem definition of the
2877 composed TOE is consistent with that specified in the STs for the component TOEs. This is
2878 determined in terms of demonstrable conformance. In particular, the evaluator examines the
2879 conformance claim rationale to determine that:

2880 a) Threat statements and OSPs in the composed TOE ST do not contradict those from the
2881 component STs.

2882 b) Any assumptions made in the component STs are upheld in the composed TOE ST. That is,
2883 either the assumption should also be present in the composed ST, or the assumption
2884 should be positively addressed in the composed ST. The assumption may be positively
2885 addressed through specification of requirements in the composed TOE to provide
2886 functionality fulfilling the concern captured in the assumption.

2887 ISO/IEC 15408-3 ASE_CCL.1.9C: *The conformance claim rationale shall demonstrate that the*
2888 *statement of security objectives is consistent with the statement of security objectives in the PP-*
2889 *Configuration or PPs for which conformance is being claimed.*

2890 **10.4.1.3.13 Work unit ASE_CCL.1-11**

2891 In this work unit, the term "PP" shall be understood to mean "PP or PP-Configuration component".

2892 The evaluator *shall examine* the conformance claim rationale to determine that the statement of
2893 security objectives is consistent, as defined by the conformance statement of the PP, with the
2894 statement of security objectives in the PPs to which conformance is being claimed.

2895 If the ST does not claim conformance to a PP, this work unit is not applicable and therefore
2896 considered to be satisfied.

- 2897 If the PP to which conformance is being claimed contains functional packages, the evaluator
2898 determines that the security objectives of the ST consist of all security objectives of all functional
2899 packages.
- 2900 If packages are used, the rules defined in the following paragraphs concerning exact, strict and
2901 demonstrable conformance also hold for the security objectives taken from the packages.
- 2902 If exact conformance is required by the PP to which conformance is being claimed, no conformance
2903 claim rationale is required. Instead, the evaluator determines whether:
- 2904 a) The ST contains all security objectives for the TOE of the PP to which conformance is being
2905 claimed. Note that in the exact conformance case, it is not allowed for the ST under
2906 evaluation to have additional security objectives for the TOE. If conformance is being
2907 claimed to more than one PP, the set of security objectives for the TOE must be identical to
2908 the union of the security objectives for the TOE in the PPs to which conformance is being
2909 claimed. It should be noted that in the case that optional requirements have associated
2910 SPD elements, exact conformance can still be claimed if objectives associated with the SPD
2911 elements are omitted when the associated optional SFRs are also omitted.
- 2912 b) The security objectives for the operational environment in the ST are identical to the
2913 security objectives for the operational environment in the PP to which conformance is
2914 being claimed. If conformance is being claimed to more than one PP, the set of security
2915 objectives for the operational environment must be identical to the union of the security
2916 objectives for the operational environment in the PPs to which conformance is being
2917 claimed with the possible exception as follows:
- 2918 - a security objective for the operational environment (or part of such security
2919 objective) from one PP can be replaced by the same (part of the) security objective
2920 for the TOE from another PP.
- 2921 If strict conformance is required by the PP to which conformance is being claimed, no conformance
2922 claim rationale is required. Instead, the evaluator determines whether:
- 2923 — The ST contains all security objectives for the TOE of the PP to which conformance is being
2924 claimed. Note that it is allowed for the ST under evaluation to have additional security
2925 objectives for the TOE;
- 2926 — The security objectives for the operational environment in the ST are identical to the security
2927 objectives for the operational environment in the PP to which conformance is being claimed,
2928 with two possible exceptions described in the following two bullet points;
- 2929 — a security objective for the operational environment (or part of such security objective) from
2930 the PP can be replaced by the same (part of the) security objective stated for the TOE;
- 2931 — a security objective for the operational environment can be added to the objectives defined in
2932 the PP, if a justification is given, why the new objective neither mitigates a threat (or a part of a
2933 threat) meant to be addressed by security objectives for the TOE in the PP, nor fulfils an OSP
2934 (or part of an OSP) meant to be addressed by security objectives for the TOE in the PP.
- 2935 When examining an ST claiming a PP, which omits security objectives for the operational
2936 environment from the PP or adds new security objectives for the operational environment, the
2937 evaluator shall carefully determine, if the conditions given above are fulfilled. The examples given
2938 for the case of assumptions in the preceding work unit are also valid here.
- 2939 If demonstrable conformance is required by the PP to which conformance is being claimed, the
2940 evaluator examines the conformance claim rationale to determine that it demonstrates that the
2941 statement of security objectives of the ST is equivalent or more restrictive than the statement of
2942 security objectives in the PP to which conformance is being claimed.

2943 For this the conformance claim rationale needs to demonstrate that the security objectives in the
2944 ST are equivalent (or more restrictive) than the security objectives in the PP. This means that:

2945 — all TOEs that would meet the security objectives for the TOE in the ST also meet the security
2946 objectives for the TOE in the PP;

2947 — all operational environments that would meet the security objectives for the operational
2948 environment in the PP would also meet the security objectives for the operational
2949 environment in the ST (with one exception in the next bullet);

2950 — besides a set of security objectives for the operational environment in the ST, which are used
2951 to demonstrate conformance to the set of security objectives defined in the PP, an ST may
2952 specify further security objectives for the operational environment, but only if these security
2953 objectives neither affect the original set of security objectives for the TOE nor the security
2954 objectives for the operational environment as defined in the PP to which conformance is
2955 claimed."

2956 For a composed TOE, the evaluator will consider whether the security objectives of the composed
2957 TOE are consistent with that specified in the STs for the component TOEs. This is determined in
2958 terms of demonstrable conformance. In particular, the evaluator examines the conformance claim
2959 rationale to determine that:

2960 a) The statement of security objectives in the dependent TOE ST relevant to any IT in the
2961 operational environment are consistent with the statement of security objectives for the
2962 TOE in the base TOE ST. It is not expected that the statement of security objectives for the
2963 environment within in the dependent TOE ST will cover all aspects of the statement of
2964 security objectives for the TOE in the base TOE ST.

2965 b) The statement of security objectives in the composed ST is consistent with the statements
2966 of security objectives in the STs for the component TOEs.

2967 If demonstrable conformance is required by the PP, the evaluator examines the conformance claim
2968 rationale to determine that it demonstrates that the statement of security objectives of the ST is at
2969 least equivalent to the statement of security objectives in the PP, or component TOE ST in the case
2970 of a composed TOE ST.

2971 ISO/IEC 15408-3 ASE_CCL.1.10C: *The conformance claim rationale shall demonstrate that the*
2972 *statement of security requirements is consistent with the statement of security requirements in the*
2973 *PP-Configuration or PPs for which conformance is being claimed.*

2974 **10.4.1.3.14 Work unit ASE_CCL.1-12**

2975 In this work unit, the term "PP" shall be understood to mean "PP or PP-Configuration component".

2976 The evaluator *shall examine* the ST to determine that it is consistent, as defined by the
2977 conformance statement of the PP, with all security requirements in the PPs for which conformance
2978 is being claimed.

2979 If the ST does not claim conformance to a PP, this work unit is not applicable and therefore
2980 considered to be satisfied.

2981 If the PP to which conformance is being claimed contains functional packages, the evaluator
2982 determines that the SFRs of the ST consist of all SFRs (or hierarchical SFRs) of all functional
2983 packages.

2984 If packages are used, the rules defined in the following paragraphs concerning exact, strict and
2985 demonstrable conformance also hold for the SFRs taken from the packages.

2986 If exact conformance is required by the PP to which conformance is being claimed, no conformance
 2987 claim rationale is required. Instead, the evaluator determines that the statement of security
 2988 requirements in the PP to which conformance is being claimed is exactly reproduced in the ST,
 2989 with the following allowances:

- 2990 a) an SFR from the PP may be iterated or refined in the ST,
- 2991 b) all SFRs that are defined in the PP to which conformance is being claimed as selection-
 2992 based upon a particular selection shall be included if and only if that selection on which
 2993 inclusion is based is present in the ST. If a selection is not chosen by the ST author, then
 2994 the selection-based SFRs associated with that selection are not included in the ST.
- 2995 c) There are no additional security requirements (SFRs or SARs) that are included in the ST
 2996 that are not also present in the PP.
- 2997 d) Optional requirements (and associated SPD elements) that 1) the ST wishes to claim
 2998 and/or 2) the ST is required to claim (as stipulated in the PP/PP-Module) due to the TOE
 2999 implementation are included; other optional requirements may be excluded while
 3000 maintaining the exact conformance claim.
- 3001 e) In the case where exact conformance is being claimed to multiple PPs, the evaluator
 3002 determines there are no additional security requirements included in the ST that are not
 3003 in at least one of the PPs, and that all of the requirements (with the allowances described
 3004 above) in all of the PPs have been included in the ST.

3005 If strict conformance is required by the PP to which conformance is being claimed, no
 3006 conformance claim rationale is required. Instead, the evaluator determines whether the
 3007 statement of security requirements in the ST is a superset of or identical to the statement of
 3008 security requirements in the PP to which conformance is being claimed (for strict
 3009 conformance).

3010 If demonstrable conformance is required by the PP to which conformance is being claimed, the
 3011 evaluator examines the conformance claim rationale to determine that it demonstrates that the
 3012 statement of security requirements of the ST is equivalent or more restrictive than the statement of
 3013 security requirements in the PP to which conformance is being claimed.

3014 For:

3015 — SFRs: The conformance rationale in the ST shall demonstrate that the overall set of
 3016 requirements defined by the SFRs in the ST is equivalent (or more restrictive) than the overall
 3017 set of requirements defined by the SFRs in the PP. This means that all TOEs that would meet
 3018 the requirements defined by the set of all SFRs in the ST would also meet the requirements
 3019 defined by the set of all SFRs in the PP;

3020 — SARs: The ST shall contain all SARs in the PP, but may claim additional SARs or replace SARs by
 3021 hierarchically stronger SARs. The completion of operations in the ST must be consistent with
 3022 that in the PP; either the same completion will be used in the ST as that in the PP or a
 3023 completion that makes the SAR more restrictive (the rules of refinement apply).

3024 For a composed TOE, the evaluator will consider whether the security requirements of the
 3025 composed TOE are consistent with that specified in the STs for the component TOEs. This is
 3026 determined in terms of demonstrable conformance. In particular, the evaluator examines the
 3027 conformance rationale to determine that:

- 3028 a) The statement of security requirements in the dependent TOE ST relevant to any IT in the
 3029 operational environment is consistent with the statement of security requirements for
 3030 the TOE in the base TOE ST. It is not expected that the statement of security requirements
 3031 for the environment within in the dependent TOE ST will cover all aspects of the

- 3032 statement of security requirements for the TOE in the base TOE ST, as some SFRs may
3033 need to be added to the statement of security requirements in the composed TOE ST.
3034 However, the statement of security requirements in the base should support the
3035 operation of the dependent component.
- 3036 b) The statement of security objectives in the dependent TOE ST relevant to any IT in the
3037 operational environment is consistent with the statement of security requirements for
3038 the TOE in the base TOE ST. It is not expected that the statement of security objectives for
3039 the environment within in the dependent TOE ST will cover all aspects of the statement
3040 of security requirements for the TOE in the base TOE ST.
- 3041 c) The statement of security requirements in the composed is consistent with the
3042 statements of security requirements in the STs for the component TOEs.
- 3043 If demonstrable conformance is required by the PP to which conformance is being claimed, the
3044 evaluator examines the conformance claim rationale to determine that it demonstrates that the
3045 statement of security requirements of the ST is at least equivalent to the statement of security
3046 requirements in the PP, or component TOE ST in the case of a composed TOE ST.
- 3047 **10.5 Security problem definition (ASE_SPD)**
- 3048 **10.5.1 Evaluation of sub-activity (ASE_SPD.1)**
- 3049 **10.5.1.1 Objectives**
- 3050 The objective of this sub-activity is to determine that the security problem intended to be
3051 addressed by the TOE and its operational environment is clearly defined.
- 3052 **10.5.1.2 Input**
- 3053 The evaluation evidence for this sub-activity is:
- 3054 a) the ST.
- 3055 **10.5.1.3 Action ASE_SPD.1.1E**
- 3056 ISO/IEC 15408-3 ASE_SPD.1.1C: *The security problem definition shall describe the threats.*
- 3057 **10.5.1.3.1 Work unit ASE_SPD.1-1**
- 3058 The evaluator **shall check** that the security problem definition describes the threats.
- 3059 If all security objectives are derived from assumptions and/or OSPs only, the statement of threats
3060 need not be present in the ST. In this case, this work unit is not applicable and therefore considered
3061 to be satisfied.
- 3062 The evaluator determines that the security problem definition describes the threats that must be
3063 countered by the TOE and/or operational environment.
- 3064 ISO/IEC 15408-3 ASE_SPD.1.2C: *All threats shall be described in terms of a threat agent, an asset,
3065 and an adverse action.*
- 3066 **10.5.1.3.2 Work unit ASE_SPD.1-2**
- 3067 The evaluator **shall examine** the security problem definition to determine that all threats are
3068 described in terms of a threat agent, an asset, and an adverse action.

- 3069 If all security objectives are derived from assumptions and/or OSPs only, the statement of threats
3070 need not be present in the ST. In this case, this work unit is not applicable and therefore considered
3071 to be satisfied.
- 3072 Threat agents may be further described by aspects such as expertise, resource, opportunity, and
3073 motivation.
- 3074 ISO/IEC 15408-3 ASE_SPD.1.3C: *The security problem definition shall describe the OSPs.*
- 3075 **10.5.1.3.3 Work unit ASE_SPD.1-3**
- 3076 The evaluator ***shall examine*** that the security problem definition describes the OSPs.
- 3077 If all security objectives are derived from assumptions and threats only, OSPs need not be present
3078 in the ST. In this case, this work unit is not applicable and therefore considered to be satisfied.
- 3079 The evaluator determines that OSP statements are made in terms of rules or guidelines that must
3080 be followed by the TOE and/or its operational environment.
- 3081 The evaluator determines that each OSP is explained and/or interpreted in sufficient detail to make
3082 it clearly understandable; a clear presentation of policy statements is necessary to permit tracing
3083 security objectives to them.
- 3084 ISO/IEC 15408-3 ASE_SPD.1.4C: *The security problem definition shall describe the assumptions*
3085 *about the operational environment of the TOE.*
- 3086 **10.5.1.3.4 Work unit ASE_SPD.1-4**
- 3087 The evaluator ***shall examine*** the security problem definition to determine that it describes the
3088 assumptions about the operational environment of the TOE.
- 3089 If there are no assumptions, this work unit is not applicable and is therefore considered to be
3090 satisfied.
- 3091 The evaluator determines that each assumption about the operational environment of the TOE is
3092 explained in sufficient detail to enable consumers to determine that their operational environment
3093 matches the assumption. If the assumptions are not clearly understood, the end result may be that
3094 the TOE is used in an operational environment in which it will not function in a secure manner.
- 3095 **10.6 Security objectives (ASE_OBJ)**
- 3096 **10.6.1 Evaluation of sub-activity (ASE_OBJ.1)**
- 3097 **10.6.1.1 Objectives**
- 3098 The objective of this sub-activity is to determine whether the security objectives for the
3099 operational environment are clearly defined.
- 3100 **10.6.1.2 Input**
- 3101 The evaluation evidence for this sub-activity is:
- 3102 a) the ST.
- 3103 **10.6.1.3 Action ASE_OBJ.1.1E**
- 3104 ISO/IEC 15408-3 ASE_OBJ.1.1C: *The statement of security objectives shall describe the security*
3105 *objectives for the operational environment.*

3106 **10.6.1.3.1 Work unit ASE_OBJ.1-1**

3107 The evaluator **shall check** that the statement of security objectives defines the security objectives
3108 for the operational environment.

3109 The evaluator checks that the security objectives for the operational environment are identified.

3110 **10.6.2 Evaluation of sub-activity (ASE_OBJ.2)**

3111 **10.6.2.1 Objectives**

3112 The objective of this sub-activity is to determine whether the security objectives adequately and
3113 completely address the security problem definition and that the division of this problem between
3114 the TOE and its operational environment is clearly defined.

3115 **10.6.2.2 Input**

3116 The evaluation evidence for this sub-activity is:

3117 a) the ST.

3118 **10.6.2.3 Action ASE_OBJ.2.1E**

3119 ISO/IEC 15408-3 ASE_OBJ.2.1C: *The statement of security objectives shall describe the security*
3120 *objectives for the TOE and the security objectives for the operational environment.*

3121 **10.6.2.3.1 Work unit ASE_OBJ.2-1**

3122 The evaluator **shall check** that the statement of security objectives defines the security objectives
3123 for the TOE and the security objectives for the operational environment.

3124 The evaluator checks that both categories of security objectives are clearly identified and
3125 separated from the other category.

3126 ISO/IEC 15408-3 ASE_OBJ.2.2C: *The security objectives rationale shall trace each security objective*
3127 *for the TOE back to threats countered by that security objective and OSPs enforced by that security*
3128 *objective.*

3129 **10.6.2.3.2 Work unit ASE_OBJ.2-2**

3130 The evaluator **shall check** that the security objectives rationale traces all security objectives for the
3131 TOE back to threats countered by the objectives and/or OSPs enforced by the objectives.

3132 Each security objective for the TOE may trace back to threats or OSPs, or a combination of threats
3133 and OSPs, but it must trace back to at least one threat or OSP.

3134 Failure to trace implies that either the security objectives rationale is incomplete, the security
3135 problem definition is incomplete, or the security objective for the TOE has no useful purpose.

3136 ISO/IEC 15408-3 ASE_OBJ.2.3C: *The security objectives rationale shall trace each security objective*
3137 *for the operational environment back to threats countered by that security objective, OSPs enforced*
3138 *by that security objective, and assumptions upheld by that security objective.*

3139 **10.6.2.3.3 Work unit ASE_OBJ.2-3**

3140 The evaluator **shall check** that the security objectives rationale traces the security objectives for
3141 the operational environment back to threats countered by that security objective, to OSPs enforced
3142 by that security objective, and to assumptions upheld by that security objective.

- 3143 Each security objective for the operational environment may trace back to threats, OSPs,
3144 assumptions, or a combination of threats, OSPs and/or assumptions, but it must trace back to at
3145 least one threat, OSP or assumption.
- 3146 Failure to trace implies that either the security objectives rationale is incomplete, the security
3147 problem definition is incomplete, or the security objective for the operational environment has no
3148 useful purpose.
- 3149 ISO/IEC 15408-3 ASE_OBJ.2.4C: *The security objectives rationale shall demonstrate that the security*
3150 *objectives counter all threats.*
- 3151 **10.6.2.3.4 Work unit ASE_OBJ.2-4**
- 3152 The evaluator ***shall examine*** the security objectives rationale to determine that it justifies for each
3153 threat that the security objectives are suitable to counter that threat.
- 3154 If no security objectives trace back to the threat, the evaluator action related to this work unit is
3155 assigned a fail verdict.
- 3156 The evaluator determines that the justification for a threat shows whether the threat is removed,
3157 diminished or mitigated.
- 3158 The evaluator determines that the justification for a threat demonstrates that the security
3159 objectives are sufficient: if all security objectives that trace back to the threat are achieved, the
3160 threat is removed, sufficiently diminished, or the effects of the threat are sufficiently mitigated.
- 3161 Note that the tracings from security objectives to threats provided in the security objectives
3162 rationale may be part of a justification, but do not constitute a justification by themselves. Even in
3163 the case that a security objective is merely a statement reflecting the intent to prevent a particular
3164 threat from being realised, a justification is required, but this justification may be as minimal as
3165 "Security Objective X directly counters Threat Y".
- 3166 The evaluator also determines that each security objective that traces back to a threat is necessary:
3167 when the security objective is achieved it actually contributes to the removal, diminishing or
3168 mitigation of that threat.
- 3169 ISO/IEC 15408-3 ASE_OBJ.2.5C: *The security objectives rationale shall demonstrate that the security*
3170 *objectives enforce all OSPs.*
- 3171 **10.6.2.3.5 Work unit ASE_OBJ.2-5**
- 3172 The evaluator ***shall examine*** the security objectives rationale to determine that for each OSP it
3173 justifies that the security objectives are suitable to enforce that OSP.
- 3174 If no security objectives trace back to the OSP, the evaluator action related to this work unit is
3175 assigned a fail verdict.
- 3176 The evaluator determines that the justification for an OSP demonstrates that the security
3177 objectives are sufficient: if all security objectives that trace back to that OSP are achieved, the OSP
3178 is enforced.
- 3179 The evaluator also determines that each security objective that traces back to an OSP is necessary:
3180 when the security objective is achieved it actually contributes to the enforcement of the OSP.
- 3181 Note that the tracings from security objectives to OSPs provided in the security objectives rationale
3182 may be part of a justification, but do not constitute a justification by themselves. In the case that a
3183 security objective is merely a statement reflecting the intent to enforce a particular OSP, a

3184 justification is required, but this justification may be as minimal as “Security Objective X directly
3185 enforces OSP Y”.

3186 ISO/IEC 15408-3 ASE_OBJ.2.6C: *The security objectives rationale shall demonstrate that the security*
3187 *objectives for the operational environment uphold all assumptions.*

3188 **10.6.2.3.6 Work unit ASE_OBJ.2-6**

3189 The evaluator **shall examine** the security objectives rationale to determine that for each
3190 assumption for the operational environment it contains an appropriate justification that the
3191 security objectives for the operational environment are suitable to uphold that assumption.

3192 If no security objectives for the operational environment trace back to the assumption, the
3193 evaluator action related to this work unit is assigned a fail verdict.

3194 The evaluator determines that the justification for an assumption about the operational
3195 environment of the TOE demonstrates that the security objectives are sufficient: if all security
3196 objectives for the operational environment that trace back to that assumption are achieved, the
3197 operational environment upholds the assumption.

3198 The evaluator also determines that each security objective for the operational environment that
3199 traces back to an assumption about the operational environment of the TOE is necessary: when the
3200 security objective is achieved it actually contributes to the operational environment upholding the
3201 assumption.

3202 Note that the tracings from security objectives for the operational environment to assumptions
3203 provided in the security objectives rationale may be a part of a justification, but do not constitute a
3204 justification by themselves. Even in the case that a security objective of the operational
3205 environment is merely a restatement of an assumption, a justification is required, but this
3206 justification may be as minimal as “Security Objective X directly upholds Assumption Y”.

3207 **10.7 Extended components definition (ASE_ECD)**

3208 **10.7.1 Evaluation of sub-activity (ASE_ECD.1)**

3209 **10.7.1.1 Objectives**

3210 The objective of this sub-activity is to determine whether extended components have been clearly
3211 and unambiguously defined, and whether they are necessary, i.e. they may not be clearly expressed
3212 using existing ISO/IEC 15408-2 or ISO/IEC 15408-3 components.

3213 **10.7.1.2 Input**

3214 The evaluation evidence for this sub-activity is:

3215 a) the ST.

3216 **10.7.1.3 Action ASE_ECD.1.1E**

3217 ISO/IEC 15408-3 ASE_ECD.1.1C: *The statement of security requirements shall identify all extended*
3218 *security requirements.*

3219 **10.7.1.3.1 Work unit ASE_ECD.1-1**

3220 The evaluator **shall check** that all security requirements in the statement of security requirements
3221 that are not identified as extended requirements are present in ISO/IEC 15408-2 or in ISO/IEC
3222 15408-3.

- 3223 ISO/IEC 15408-3 ASE_ECD.1.2C: *The extended components definition shall define an extended*
3224 *component for each extended security requirement.*
- 3225 **10.7.1.3.2 Work unit ASE_ECD.1-2**
- 3226 The evaluator **shall check** that the extended components definition defines an extended
3227 component for each extended security requirement.
- 3228 If the ST does not contain extended security requirements, this work unit is not applicable and
3229 therefore considered to be satisfied.
- 3230 A single extended component may be used to define multiple iterations of an extended security
3231 requirement, it is not necessary to repeat this definition for each iteration.
- 3232 ISO/IEC 15408-3 ASE_ECD.1.3C: *The extended components definition shall describe how each*
3233 *extended component is related to the existing ISO/IEC 15408 components, families, and classes.*
- 3234 **10.7.1.3.3 Work unit ASE_ECD.1-3**
- 3235 The evaluator **shall examine** the extended components definition to determine that it describes
3236 how each extended component fits into the existing ISO/IEC 15408 components, families, and
3237 classes.
- 3238 If the ST does not contain extended security requirements, this work unit is not applicable and
3239 therefore considered to be satisfied.
- 3240 The evaluator determines that each extended component is either:
- 3241 a) a member of an existing ISO/IEC 15408-2 or ISO/IEC 15408-3 family, or
- 3242 b) a member of a new family defined in the ST.
- 3243 If the extended component is a member of an existing ISO/IEC 15408-2 or ISO/IEC 15408-3 family,
3244 the evaluator determines that the extended components definition adequately describes why the
3245 extended component should be a member of that family and how it relates to other components of
3246 that family.
- 3247 If the extended component is a member of a new family defined in the ST, the evaluator confirms
3248 that the extended component is not appropriate for an existing family.
- 3249 If the ST defines new families, the evaluator determines that each new family is either:
- 3250 a) a member of an existing ISO/IEC 15408-2 or ISO/IEC 15408-3 class, or
- 3251 b) a member of a new class defined in the ST.
- 3252 If the family is a member of an existing ISO/IEC 15408-2 or ISO/IEC 15408-3 class, the evaluator
3253 determines that the extended components definition adequately describes why the family should
3254 be a member of that class and how it relates to other families in that class.
- 3255 If the family is a member of a new class defined in the ST, the evaluator confirms that the family is
3256 not appropriate for an existing class.
- 3257 **10.7.1.3.4 Work unit ASE_ECD.1-4**
- 3258 The evaluator **shall examine** the extended components definition to determine that each definition
3259 of an extended component identifies all applicable dependencies of that component.

- 3260 If the ST does not contain extended security requirements, this work unit is not applicable and
3261 therefore considered to be satisfied.
- 3262 The evaluator confirms that no applicable dependencies have been overlooked by the ST author.
- 3263 ISO/IEC 15408-3 ASE_ECD.1.4C: *The extended components definition shall use the existing ISO/IEC*
3264 *15408 components, families, classes, and methodology as a model for presentation.*
- 3265 **10.7.1.3.5 Work unit ASE_ECD.1-5**
- 3266 The evaluator **shall examine** the extended components definition to determine that each extended
3267 functional component uses the existing ISO/IEC 15408-2 components as a model for presentation.
- 3268 If the ST does not contain extended SFRs, this work unit is not applicable and therefore considered
3269 to be satisfied.
- 3270 The evaluator determines that the extended functional component is consistent with ISO/IEC
3271 15408-2 Subclause **6.1.3, Component structure.**
- 3272 If the extended functional component uses operations, the evaluator determines that the extended
3273 functional component is consistent with ISO/IEC 15408-1 Subclause **7.1, Operations.**
- 3274 If the extended functional component is hierarchical to an existing functional component, the
3275 evaluator determines that the extended functional component is consistent with ISO/IEC 15408-2
3276 Subclause **6.2.1, Component changes highlighting.**
- 3277 **10.7.1.3.6 Work unit ASE_ECD.1-6**
- 3278 The evaluator **shall examine** the extended components definition to determine that each definition
3279 of a new functional family uses the existing ISO/IEC 15408 functional families as a model for
3280 presentation.
- 3281 If the ST does not define new functional families, this work unit is not applicable and therefore
3282 considered to be satisfied.
- 3283 The evaluator determines that all new functional families are defined consistent with ISO/IEC
3284 15408-2 Subclause **6.1.2, Family structure.**
- 3285 **10.7.1.3.7 Work unit ASE_ECD.1-7**
- 3286 The evaluator **shall examine** the extended components definition to determine that each definition
3287 of a new functional class uses the existing ISO/IEC 15408 functional classes as a model for
3288 presentation.
- 3289 If the ST does not define new functional classes, this work unit is not applicable and therefore
3290 considered to be satisfied.
- 3291 The evaluator determines that all new functional classes are defined consistent with ISO/IEC
3292 15408-2 Subclause **6.1.1, Class structure.**
- 3293 **10.7.1.3.8 Work unit ASE_ECD.1-8**
- 3294 The evaluator **shall examine** the extended components definition to determine that each definition
3295 of an extended assurance component uses the existing ISO/IEC 15408-3 components as a model
3296 for presentation.
- 3297 If the ST does not contain extended SARs, this work unit is not applicable and therefore considered
3298 to be satisfied.

- 3299 The evaluator determines that the extended assurance component definition is consistent with
3300 ISO/IEC 15408-3 Subclause **6.1.3, Assurance component structure.**
- 3301 If the extended assurance component uses operations, the evaluator determines that the extended
3302 assurance component is consistent with ISO/IEC 15408-1 Subclause **7.1, Operations.**
- 3303 If the extended assurance component is hierarchical to an existing assurance component, the
3304 evaluator determines that the extended assurance component is consistent with ISO/IEC 15408-3
3305 Subclause **6.1.3, Assurance component structure.**
- 3306 **10.7.1.3.9 Work unit ASE_ECD.1-9**
- 3307 The evaluator **shall examine** the extended components definition to determine that, for each
3308 defined extended assurance component, applicable methodology has been provided.
- 3309 If the ST does not contain extended SARs, this work unit is not applicable and therefore considered
3310 to be satisfied.
- 3311 The evaluator determines that, for each evaluator action element of each extended SAR, one or
3312 more work units are provided and that successfully performing all work units for a given evaluator
3313 action element will demonstrate that the element has been achieved.
- 3314 **10.7.1.3.10 Work unit ASE_ECD.1-10**
- 3315 The evaluator **shall examine** the extended components definition to determine that each definition
3316 of a new assurance family uses the existing ISO/IEC 15408 assurance families as a model for
3317 presentation.
- 3318 If the ST does not define new assurance families, this work unit is not applicable and therefore
3319 considered to be satisfied.
- 3320 The evaluator determines that all new assurance families are defined consistent with ISO/IEC
3321 15408-3 Subclause **6.1.2, Assurance family structure.**
- 3322 **10.7.1.3.11 Work unit ASE_ECD.1-11**
- 3323 The evaluator **shall examine** the extended components definition to determine that each definition
3324 of a new assurance class uses the existing ISO/IEC 15408 assurance classes as a model for
3325 presentation.
- 3326 If the ST does not define new assurance classes, this work unit is not applicable and therefore
3327 considered to be satisfied.
- 3328 The evaluator determines that all new assurance classes are defined consistent with ISO/IEC
3329 15408-3 Subclause **6.1.1, Assurance class structure.**
- 3330 ISO/IEC 15408-3 ASE_ECD.1.5C: *The extended components shall consist of measurable and objective*
3331 *elements such that conformance or nonconformance to these elements can be demonstrated.*
- 3332 **10.7.1.3.12 Work unit ASE_ECD.1-12**
- 3333 The evaluator **shall examine** the extended components definition to determine that each element
3334 in each extended component is measurable and states objective evaluation requirements, such that
3335 conformance or nonconformance can be demonstrated.
- 3336 If the ST does not contain extended security requirements, this work unit is not applicable and
3337 therefore considered to be satisfied.

3338 The evaluator determines that elements of extended functional components are stated in such a
3339 way that they are testable, and traceable through the appropriate TSF representations.

3340 The evaluator also determines that elements of extended assurance components avoid the need for
3341 subjective evaluator judgement.

3342 The evaluator is reminded that whilst being measurable and objective is appropriate for all
3343 evaluation criteria, it is acknowledged that no formal method exists to prove such properties.
3344 Therefore the existing ISO/IEC 15408 functional and assurance components are to be used as a
3345 model for determining what constitutes conformance with this requirement.

3346 **10.7.1.4 Action ASE_ECD.1.2E**

3347 **10.7.1.4.1 Work unit ASE_ECD.1-13**

3348 The evaluator *shall examine* the extended components definition to determine that each extended
3349 component can not be clearly expressed using existing components.

3350 If the ST does not contain extended security requirements, this work unit is not applicable and
3351 therefore considered to be satisfied.

3352 The evaluator should take components from ISO/IEC 15408-2 and ISO/IEC 15408-3, other
3353 extended components that have been defined in the ST, combinations of these components, and
3354 possible operations on these components into account when making this determination.

3355 The evaluator is reminded that the role of this work unit is to preclude unnecessary duplication of
3356 components, that is, components that may be clearly expressed by using other components. The
3357 evaluator should not undertake an exhaustive search of all possible combinations of components
3358 including operations in an attempt to find a way to express the extended component by using
3359 existing components.

3360 **10.8 Security requirements (ASE_REQ)**

3361 **10.8.1 Evaluation of sub-activity (ASE_REQ.1)**

3362 **10.8.1.1 Objectives**

3363 The objective of this sub-activity is to determine whether the SFRs and SARs are clear,
3364 unambiguous and well-defined, whether they are internally consistent, and whether the SFRs
3365 counter the threats and implement the organisational security policies of the TOE..

3366 **10.8.1.2 Input**

3367 The evaluation evidence for this sub-activity is:

3368 a) the ST.

3369 **10.8.1.3 Action ASE_REQ.1.1E**

3370 ISO/IEC 15408-3 ASE_REQ.1.1C: *The statement of security requirements shall describe the SFRs and*
3371 *the SARs.*

3372 **10.8.1.3.1 Work unit ASE_REQ.1-1**

3373 The evaluator *shall check* that the statement of security requirements describes the SFRs.

3374 The evaluator determines that each SFR is identified by one of the following means:

- 3375 a) by reference to an individual component in ISO/IEC 15408-2;
- 3376 b) by reference to an extended component in the extended components definition of the ST;
- 3377 c) by reference to a PP that the ST claims to be conformant with, including any optional
3378 requirements defined in the PP;
- 3379 d) by reference to a security requirements package that the ST claims to be conformant with;
- 3380 e) by reproduction in the ST.

3381 It is not required to use the same means of identification for all SFRs.

3382 **10.8.1.3.2 Work unit ASE_REQ.1-2**

3383 The evaluator **shall check** that the statement of security requirements describes the SARs.

3384 The evaluator determines that each SAR is identified by one of the following means:

- 3385 a) by reference to an individual component in ISO/IEC 15408-3;
- 3386 b) by reference to an extended component in the extended components definition of the ST;
- 3387 c) by reference to a PP that the ST claims to be conformant with;
- 3388 d) by reference to a security requirements package that the ST claims to be conformant with;
- 3389 e) by reproduction in the ST.

3390 It is not required to use the same means of identification for all SARs.

3391 Note that if optional requirements are defined by the PP, there may be associated threats that are
3392 covered by this work unit.

3393 ISO/IEC 15408-3 ASE_REQ.1.2C: *All subjects, objects, operations, security attributes, external entities*
3394 *and other terms that are used in the SFRs and the SARs shall be defined.*

3395 **10.8.1.3.3 Work unit ASE_REQ.1-3**

3396 The evaluator **shall examine** the ST to determine that all subjects, objects, operations, security
3397 attributes, external entities and other terms that are used in the SFRs and the SARs are defined.

3398 The evaluator determines that the ST defines all:

- 3399 — (types of) subjects and objects that are used in the SFRs;
- 3400 — (types of) security attributes of subjects, users, objects, information, sessions and/or resources,
3401 possible values that these attributes may take and any relations between these values (e.g.
3402 top_secret is "higher" than secret);
- 3403 — (types of) operations that are used in the SFRs, including the effects of these operations;
- 3404 — (types of) external entities in the SFRs;
- 3405 — other terms that are introduced in the SFRs and/or SARs by completing operations, if these
3406 terms are not immediately clear, or are used outside their dictionary definition.

3407 The goal of this work unit is to ensure that the SFRs and SARs are well-defined and that no
3408 misunderstanding may occur due to the introduction of vague terms. This work unit should not be
3409 taken into extremes, by forcing the ST author to define every single word. The general audience of
3410 a set of security requirements should be assumed to have a reasonable knowledge of IT, security
3411 and "Evaluation criteria for IT security".

3412 All of the above may be presented in groups, classes, roles, types or other groupings or
3413 characterisations that allow easy understanding.

3414 The evaluator is reminded that these lists and definitions do not have to be part of the statement of
3415 security requirements, but may be placed (in part or in whole) in different subclauses. This may be
3416 especially applicable if the same terms are used in the rest of the ST.

3417 ISO/IEC 15408-3 ASE_REQ.1.3C: *The statement of security requirements shall include a natural*
3418 *language description, part of which describes how the SFRs combine together to provide security*
3419 *functionality in terms of the architecture that is visible to Administrators and other users.*

3420 **10.8.1.3.4 Work unit ASE_REQ.1-4**

3421 The evaluator **shall check** that the statement of security requirements includes a natural language
3422 description, part of which describes how the SFRs combine together to provide security
3423 functionality in terms of the architecture that is visible to Administrators and other users.

3424 The description is intended to make clear connections between SFRs and to provide a view of how
3425 they provide security functionality that is recognizable to Administrators and other types of
3426 user. The description in terms of the architecture that is "visible to Administrators and other
3427 users" means that the description must relate the security behavior to visible elements, but
3428 the mechanisms themselves need not be visible. For example: when describing authentication
3429 using a biometric mechanism, the calculation of the match or score might not be visible, but
3430 (a) might relate to a referenced description of a matching algorithm, (b) might be based on
3431 specific template files maintained by the Administrator, and (c) will result in acceptance or
3432 rejection of the authentication attempt – therefore the description might make use of any or
3433 all of these items (a) – (c). No specific format for this information is prescribed, and the
3434 description need not all be located alongside the SFRs themselves (e.g. some of it might be in
3435 the ST Introduction and/or in the TSS). The intention of the requirement is to make the
3436 meaning of the SFRs clearer and more easily understood by readers of the ST who may not
3437 have deep knowledge of the CC but who are familiar with the product type.

3438 The evaluator determines that all operations are identified in each SFR or SAR where such an
3439 operation is used. This includes both completed operations and uncompleted operations.
3440 Identification may be achieved by typographical distinctions, or by explicit identification in the
3441 surrounding text, or by any other distinctive means.

3442 ISO/IEC 15408-3 ASE_REQ.1.4C: *The statement of security requirements shall identify all operations*
3443 *on the security requirements.*

3444 **10.8.1.3.5 Work unit ASE_REQ.1-5**

3445 The evaluator **shall check** that the statement of security requirements identifies all operations on
3446 the security requirements.

3447 The evaluator determines that all operations are identified in each SFR or SAR where such an
3448 operation is used. Identification may be achieved by typographical distinctions, or by explicit
3449 identification in the surrounding text, or by any other distinctive means.

3450 ISO/IEC 15408-3 ASE_REQ.1.5C: *All operations shall be performed correctly.*

3451 **10.8.1.3.6 Work unit ASE_REQ.1-6**

3452 The evaluator *shall examine* the statement of security requirements to determine that all
3453 assignment operations are performed correctly.

3454 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
3455 **Guidance for Operations.**

3456 **10.8.1.3.7 Work unit ASE_REQ.1-7**

3457 The evaluator *shall examine* the statement of security requirements to determine that all iteration
3458 operations are performed correctly.

3459 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
3460 **Guidance for Operations.**

3461 **10.8.1.3.8 Work unit ASE_REQ.1-8**

3462 The evaluator *shall examine* the statement of security requirements to determine that all selection
3463 operations are performed correctly.

3464 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
3465 **Guidance for Operations.**

3466 **10.8.1.3.9 Work unit ASE_REQ.1-9**

3467 The evaluator *shall examine* the statement of security requirements to determine that all
3468 refinement operations are performed correctly.

3469 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
3470 **Guidance for Operations.**

3471 ISO/IEC 15408-3 ASE_REQ.1.6C: *Each dependency of the security requirements shall either be*
3472 *satisfied, or the security requirements rationale shall justify the dependency not being satisfied.*

3473 **10.8.1.3.10 Work unit ASE_REQ.1-10**

3474 The evaluator *shall examine* the statement of security requirements to determine that each
3475 dependency of the security requirements is either satisfied, or that a security requirements
3476 rationale is provided which justifies the dependency not being satisfied.

3477 A dependency is satisfied by the inclusion of the relevant component (or one that is hierarchical to
3478 it) within the statement of security requirements. The component used to satisfy the dependency
3479 should, if necessary, be modified by operations to ensure that it actually satisfies that dependency.

3480 A justification that a dependency is not met should address either:

3481 a) why the dependency is not necessary or useful, in which case no further information is
3482 required; or

3483 b) that the dependency has been addressed by the operational environment of the TOE, in
3484 which case the justification should describe how the security objectives for the
3485 operational environment address this dependency.

3486 ISO/IEC 15408-3 ASE_REQ.1.7C: The security requirements rationale shall trace each SFR back to
3487 the threats countered by that SFR and OSPs enforced by that SFR.

3488 **10.8.1.3.11 Work unit ASE_REQ.1-11**

3489 The evaluator **shall check** that the security requirements rationale traces each SFR back to the
3490 threats countered by that SFR and OSPs enforced by that SFR.

3491 The evaluator determines that each SFR is traced back to at least one threat or OSP for the TOE.

3492 Failure to trace implies that either the security requirements rationale is incomplete, the security
3493 objectives for the TOE are incomplete, or the SFR has no useful purpose.

3494 There is no prescribed location for this part of the rationale: for example, the relevant parts may be
3495 located under each threat and OSP in order to help make the security argument clearer and easier
3496 to read.

3497 Optional requirements may require Threats/OSPs to be specified, and security objectives
3498 associated with these SPD elements are also covered by this work unit.

3499 ISO/IEC 15408-3 ASE_REQ.1.8C: *The security requirements rationale shall trace each security
3500 objective for the operational environment back to threats countered by that security objective, OSPs
3501 enforced by that security objective, and assumptions upheld by that security objective.*

3502 **10.8.1.3.12 Work unit ASE_REQ.1-12**

3503 The evaluator **shall check** that the security objectives requirements rationale traces the security
3504 objectives for the operational environment back to threats countered by that security objective, to
3505 OSPs enforced by that security objective, and to assumptions upheld by that security objective.

3506 Each security objective for the operational environment may trace back to threats, OSPs,
3507 assumptions, or a combination of threats, OSPs and/or assumptions, but it must trace back to at
3508 least one threat, OSP or assumption.

3509 Failure to trace implies that either the security objectives requirements rationale is incomplete, the
3510 security problem definition is incomplete, or the security objective for the operational
3511 environment has no useful purpose.

3512 There is no prescribed location for this part of the rationale: for example, the relevant parts may be
3513 located under each threat, OSP and assumption in order to help make the security argument
3514 clearer and easier to read.

3515 ISO/IEC 15408-3 ASE_REQ.1.9C: *The security requirements rationale shall demonstrate that the
3516 SFRs (in conjunction with the security objectives for the environment) counter all threats for the
3517 TOE.*

3518 **10.8.1.3.13 Work unit ASE_REQ.1-13**

3519 The evaluator **shall examine** the security requirements rationale to determine that for each threat
3520 it demonstrates that the SFRs are suitable to meet that threat.

3521 If no SFRs trace back to a threat, the evaluator action related to this work unit is assigned a fail
3522 verdict.

3523 The evaluator determines that the justification for a threat shows whether the threat is removed,
3524 diminished or mitigated.

3525 The evaluator determines that the justification for a threat demonstrates that the SFRs are
3526 sufficient: if all SFRs that trace back to the threat are achieved then, in the context of any applicable
3527 OSPs and assumptions, the threat is removed, sufficiently diminished, or the effects of the threat
3528 are sufficiently mitigated.

- 3529 Note that simply listing in the security requirements rationale the SFRs associated with each threat
 3530 may be part of a justification, but does not constitute a justification by itself. A descriptive
 3531 justification is required, although in simple cases this justification may be as minimal as “SFR X
 3532 directly counters Threat Y”.
- 3533 The evaluator also determines that each SFR that traces back to a threat is necessary: when the SFR
 3534 is implemented it actually contributes to the removal, diminishing or mitigation of that threat.
- 3535 ISO/IEC 15408-3 ASE_REQ.1.10C: The security requirements rationale shall demonstrate that the
 3536 SFRs (in conjunction with the security objectives for the environment) enforce all OSPs for the TOE.
- 3537 **10.8.1.3.14 Work unit ASE_REQ.1-14**
- 3538 The evaluator *shall examine* the security requirements rationale to determine that for each OSP it
 3539 justifies that the SFRs are suitable to enforce that OSP.
- 3540 If no SFRs or security objectives for the operational environment trace back to the OSP, the
 3541 evaluator action related to this work unit is assigned a fail verdict.
- 3542 The evaluator determines that the justification for an OSP demonstrates that the security
 3543 objectives are sufficient: if all SFRs that trace back to that OSP are achieved then, in the context of
 3544 any applicable assumptions, the OSP is enforced.
- 3545 The evaluator also determines that each SFR that traces back to an OSP is necessary: when the SFR
 3546 is implemented it actually contributes to the enforcement of the OSP.
- 3547 Note that simply listing in the security requirements rationale the SFRs associated with each OSP
 3548 may be part of a justification, but does not constitute a justification by itself. A descriptive
 3549 justification is required, although in simple cases this justification may be as minimal as “SFR X
 3550 directly enforces OSP Y”.
- 3551 ISO/IEC 15408-3 ASE_REQ.1.11C: The security requirements rationale shall demonstrate that the
 3552 SFRs (in conjunction with the security objectives for the environment) enforce all OSPs for the TOE.
- 3553 **10.8.1.3.15 Work unit ASE_REQ.1-15**
- 3554 The evaluator *shall examine* the security requirements rationale to determine that for each
 3555 assumption for the operational environment it contains an appropriate justification that the
 3556 security objectives for the operational environment are suitable to uphold that assumption.
- 3557 If no security objectives for the operational environment trace back to the assumption, the
 3558 evaluator action related to this work unit is assigned a fail verdict.
- 3559 The evaluator determines that the justification for an assumption about the operational
 3560 environment of the TOE demonstrates that the security objectives are sufficient: if all security
 3561 objectives for the operational environment that trace back to that assumption are achieved, the
 3562 operational environment upholds the assumption.
- 3563 The evaluator also determines that each security objective for the operational environment that
 3564 traces back to an assumption about the operational environment of the TOE is necessary: when the
 3565 security objective is achieved it actually contributes to the operational environment upholding the
 3566 assumption.
- 3567 Note that simply listing in the security requirements rationale the security objectives for the
 3568 operational environment associated with each assumption may be a part of a justification, but does
 3569 not constitute a justification by itself. A descriptive justification is required, although in simple
 3570 cases this justification may be as minimal as “Security Objective X directly upholds Assumption Y”.

3571 ISO/IEC 15408-3 ASE_REQ.1.12C: *The statement of security requirements shall be internally*
3572 *consistent.*

3573 **10.8.1.3.16 Work unit ASE_REQ.1-16**

3574 The evaluator **shall examine** the statement of security requirements to determine that it is
3575 internally consistent.

3576 The evaluator determines that the combined set of all SFRs and SARs is internally consistent. With
3577 respect to optional requirements, the evaluator determines that:

3578 a) All optional requirements either trace to an SPD element that is itself not optional, or trace
3579 to an SPD element that is clearly associated with that optional SFR;

3580 b) All optional requirements are clearly identified as being required if a conformance TOE
3581 implements the functionality covered by the requirement, or as being “purely optional”;
3582 and

3583 c) All optional requirements do not conflict with non-optional requirements (a capability
3584 cannot be both required and optional; however, a base capability can be required with
3585 enhancements to that capability being specified as optional).

3586

3587 The evaluator determines that on all occasions where different security requirements apply to the
3588 same types of developer evidence, events, operations, data, tests to be performed etc. or to “all
3589 objects”, “all subjects” etc., that these requirements do not conflict.

3590 Some possible conflicts are:

3591 a) an extended SAR specifying that the design of a certain cryptographic algorithm is to be
3592 kept secret, and another extended SAR specifying an open source review;

3593 b) **FAU_GEN.1 Audit data generation** specifying that subject identity is to be logged,
3594 **FDP_ACC.1 Subset access control** specifying who has access to these logs, and **FPR_UNO.1**
3595 **Unobservability** specifying that some actions of subjects should be unobservable to other
3596 subjects. If the subject that should not be able to see an activity may access logs of this
3597 activity, these SFRs conflict;

3598 c) **FDP_RIP.1 Subset residual information protection** specifying deletion of information no
3599 longer needed, and **FDP_ROL.1 Basic rollback** specifying that a TOE may return to a
3600 previous state. If the information that is needed for the rollback to the previous state has
3601 been deleted, these requirements conflict;

3602 d) Multiple iterations of **FDP_ACC.1 Subset access control** especially where some iterations
3603 cover the same subjects, objects, or operations. If one access control SFR allows a subject
3604 to perform an operation on an object, while another access control SFR does not allow
3605 this, these requirements conflict.

3606 **10.8.2 Evaluation of sub-activity (ASE_REQ.2)**

3607 **10.8.2.1 Objectives**

3608 The objective of this sub-activity is to determine whether the SFRs and SARs are clear,
3609 unambiguous and well-defined, whether they are internally consistent, and whether the SFRs meet
3610 the security objectives of the TOE.

3611 **10.8.2.2 Input**

3612 The evaluation evidence for this sub-activity is:

3613 a) the ST.

3614 **10.8.2.3 Action ASE_REQ.2.1E**3615 ISO/IEC 15408-3 ASE_REQ.2.1C: *The statement of security requirements shall describe the SFRs and*
3616 *the SARs.*3617 **10.8.2.3.1 Work unit ASE_REQ.2-1**3618 The evaluator **shall check** that the statement of security requirements describes the SFRs.

3619 The evaluator determines that each SFRs is identified by one of the following means:

3620 a) by reference to an individual component in ISO/IEC 15408-2;

3621 b) by reference to an extended component in the extended components definition of the ST;

3622 c) by reference to an individual component in a PP that the ST claims to be conformant with,
3623 including any optional requirements defined in the PP;3624 d) by reference to an individual component in a security requirements package that the ST
3625 claims to be conformant with;

3626 e) by reproduction in the ST.

3627 It is not required to use the same means of identification for all SFRs.

3628 **10.8.2.3.2 Work unit ASE_REQ.2-2**3629 The evaluator **shall check** that the statement of security requirements describes the SARs.

3630 The evaluator determines that all SARs are identified by one of the following means:

3631 a) by reference to an individual component in ISO/IEC 15408-3;

3632 b) by reference to an extended component in the extended components definition of the ST;

3633 c) by reference to an individual component in a PP that the ST claims to be conformant with;

3634 d) by reference to an individual component in a security requirements package that the ST
3635 claims to be conformant with;

3636 e) by reproduction in the ST.

3637 It is not required to use the same means of identification for all SARs.

3638 Note that if optional requirements are defined by the PP, there may be associated threats that are
3639 covered by this work unit.3640 ISO/IEC 15408-3 ASE_REQ.2.2C: *All subjects, objects, operations, security attributes, external entities*
3641 *and other terms that are used in the SFRs and the SARs shall be defined.*

3642 **10.8.2.3.3 Work unit ASE_REQ.2-3**

3643 The evaluator **shall examine** the ST to determine that all subjects, objects, operations, security
3644 attributes, external entities and other terms that are used in the SFRs and the SARs are defined.

3645 The evaluator determines that the ST defines all:

3646 — (types of) subjects and objects that are used in the SFRs;

3647 — (types of) security attributes of subjects, users, objects, information, sessions and/or resources,
3648 possible values that these attributes may take and any relations between these values (e.g.
3649 top_secret is “higher” than secret);

3650 — (types of) operations that are used in the SFRs, including the effects of these operations;

3651 — (types of) external entities in the SFRs;

3652 — other terms that are introduced in the SFRs and/or SARs by completing operations, if these
3653 terms are not immediately clear, or are used outside their dictionary definition.

3654 The goal of this work unit is to ensure that the SFRs and SARs are well-defined and that no
3655 misunderstanding may occur due to the introduction of vague terms. This work unit should not be
3656 taken into extremes, by forcing the ST author to define every single word. The general audience of
3657 a set of security requirements should be assumed to have a reasonable knowledge of IT, security
3658 and “Evaluation criteria for IT security”.

3659 All of the above may be presented in groups, classes, roles, types or other groupings or
3660 characterisations that allow easy understanding.

3661 The evaluator is reminded that these lists and definitions do not have to be part of the statement of
3662 security requirements, but may be placed (in part or in whole) in different subclauses. This may be
3663 especially applicable if the same terms are used in the rest of the ST.

3664 ISO/IEC 15408-3 ASE_REQ.2.3C: *The statement of security requirements shall identify all operations*
3665 *on the security requirements.*

3666 **10.8.2.3.4 Work unit ASE_REQ.2-4**

3667 The evaluator **shall check** that the statement of security requirements identifies all operations on
3668 the security requirements.

3669 The evaluator determines that all operations are identified in each SFR or SAR where such an
3670 operation is used. Identification may be achieved by typographical distinctions, or by explicit
3671 identification in the surrounding text, or by any other distinctive means.

3672 ISO/IEC 15408-3 ASE_REQ.2.4C: *All operations shall be performed correctly.*

3673 **10.8.2.3.5 Work unit ASE_REQ.2-5**

3674 The evaluator **shall examine** the statement of security requirements to determine that all
3675 assignment operations are performed correctly.

3676 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
3677 **Guidance for Operations.**

3678 **10.8.2.3.6 Work unit ASE_REQ.2-6**

3679 The evaluator *shall examine* the statement of security requirements to determine that all iteration
3680 operations are performed correctly.

3681 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
3682 **Guidance for Operations.**

3683 **10.8.2.3.7 Work unit ASE_REQ.2-7**

3684 The evaluator *shall examine* the statement of security requirements to determine that all selection
3685 operations are performed correctly.

3686 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
3687 **Guidance for Operations.**

3688 **10.8.2.3.8 Work unit ASE_REQ.2-8**

3689 The evaluator *shall examine* the statement of security requirements to determine that all
3690 refinement operations are performed correctly.

3691 Guidance on the correct performance of operations may be found in ISO/IEC 15408-1 Annex C,
3692 **Guidance for Operations.**

3693 ISO/IEC 15408-3 ASE_REQ.2.5C: *Each dependency of the security requirements shall either be*
3694 *satisfied, or the security requirements rationale shall justify the dependency not being satisfied.*

3695 **10.8.2.3.9 Work unit ASE_REQ.2-9**

3696 The evaluator *shall examine* the statement of security requirements to determine that each
3697 dependency of the security requirements is either satisfied, or that the security requirements
3698 rationale justifies the dependency not being satisfied.

3699 A dependency is satisfied by the inclusion of the relevant component (or one that is hierarchical to
3700 it) within the statement of security requirements. The component used to satisfy the dependency
3701 should, if necessary, be modified by operations to ensure that it actually satisfies that dependency.

3702 A justification that a dependency is not met should address either:

3703 a) why the dependency is not necessary or useful, in which case no further information is
3704 required; or

3705 b) that the dependency has been addressed by the operational environment of the TOE, in
3706 which case the justification should describe how the security objectives for the
3707 operational environment address this dependency.

3708 ISO/IEC 15408-3 ASE_REQ.2.6C: *The security requirements rationale shall trace each SFR back to the*
3709 *SPD elements for the TOE.*

3710 **10.8.2.3.10 Work unit ASE_REQ.2-10**

3711 The evaluator *shall check* that the security requirements rationale traces each SFR back to the
3712 security objectives for the TOE.

3713 Optional requirements may require Threats/OSPs to be specified, and security objectives
3714 associated with these SPD elements are also covered by this work unit.

3715 The evaluator determines that each SFR is traced back to at least one security objective for the TOE.

3716 Failure to trace implies that either the security requirements rationale is incomplete, the security
3717 objectives for the TOE are incomplete, or the SFR has no useful purpose.

3718 ISO/IEC 15408-3 ASE_REQ.2.7C: *The security requirements rationale shall demonstrate that the*
3719 *SFRs meet all security objectives for the TOE.*

3720 **10.8.2.3.11 Work unit ASE_REQ.2-11**

3721 The evaluator *shall examine* the security requirements rationale to determine that for each
3722 security objective for the TOE it demonstrates that the SFRs are suitable to meet that security
3723 objective for the TOE.

3724 If no SFRs trace back to the security objective for the TOE, the evaluator action related to this work
3725 unit is assigned a fail verdict.

3726 The evaluator determines that the justification for a security objective for the TOE demonstrates
3727 that the SFRs are sufficient: if all SFRs that trace back to the objective are satisfied, the security
3728 objective for the TOE is achieved.

3729 The evaluator also determines that each SFR that traces back to a security objective for the TOE is
3730 necessary: when the SFR is satisfied, it actually contributes to achieving the security objective.

3731 Note that the tracings from SFRs to security objectives for the TOE provided in the security
3732 requirements rationale may be a part of the justification, but do not constitute a justification by
3733 themselves.

3734 ISO/IEC 15408-3 ASE_REQ.2.8C: *The security requirements rationale shall explain why the SARs*
3735 *were chosen.*

3736 **10.8.2.3.12 Work unit ASE_REQ.2-12**

3737 The evaluator *shall check* that the security requirements rationale explains why the SARs were
3738 chosen.

3739 The evaluator is reminded that any explanation is correct, as long as it is coherent and neither the
3740 SARs nor the explanation have obvious inconsistencies with the remainder of the ST.

3741 An example of an obvious inconsistency between the SARs and the remainder of the ST would be to
3742 have threat agents that are very capable, but an AVA_VAN SAR that does not protect against these
3743 threat agents.

3744 ISO/IEC 15408-3 ASE_REQ.2.9C: *The statement of security requirements shall be internally*
3745 *consistent.*

3746 **10.8.2.3.13 Work unit ASE_REQ.2-13**

3747 The evaluator *shall examine* the statement of security requirements to determine that it is
3748 internally consistent.

3749 The evaluator determines that the combined set of all SFRs and SARs is internally consistent. With
3750 respect to optional requirements, the evaluator determines that:

3751 a) All optional requirements either trace to an SPD element that is itself not optional, or trace
3752 to an SPD element that is clearly associated with that optional SFR;

3753 b) All optional requirements are clearly identified as being required if a conformance TOE
3754 implements the functionality covered by the requirement, or as being “purely optional”;
3755 and

3756 c) All optional requirements do not conflict with non-optional requirements (a capability
3757 cannot be both required and optional; however, a base capability can be required with
3758 enhancements to that capability being specified as optional).

3759

3760 The evaluator determines that on all occasions where different security requirements apply to the
3761 same types of developer evidence, events, operations, data, tests to be performed etc. or to “all
3762 objects”, “all subjects” etc., that these requirements do not conflict.

3763 Some possible conflicts are:

3764 a) an extended SAR specifying that the design of a certain cryptographic algorithm is to be
3765 kept secret, and another extended assurance requirement specifying an open source
3766 review;

3767 b) **FAU_GEN.1 Audit data generation** specifying that subject identity is to be logged,
3768 **FDP_ACC.1 Subset access control** specifying who has access to these logs, and **FPR_UNO.1**
3769 **Unobservability** specifying that some actions of subjects should be unobservable to other
3770 subjects. If the subject that should not be able to see an activity may access logs of this
3771 activity, these SFRs conflict;

3772 c) **FDP_RIP.1 Subset residual information protection** specifying deletion of information no
3773 longer needed, and **FDP_ROL.1 Basic rollback** specifying that a TOE may return to a
3774 previous state. If the information that is needed for the rollback to the previous state has
3775 been deleted, these requirements conflict;

3776 d) Multiple iterations of **FDP_ACC.1 Subset access control** especially where some iterations
3777 cover the same subjects, objects, or operations. If one access control SFR allows a subject
3778 to perform an operation on an object, while another access control SFR does not allow
3779 this, these requirements conflict.

3780 **10.9 TOE summary specification (ASE_TSS)**

3781 **10.9.1 Evaluation of sub-activity (ASE_TSS.1)**

3782 **10.9.1.1 Objectives**

3783 The objective of this sub-activity is to determine whether the TOE summary specification
3784 addresses all SFRs, and whether the TOE summary specification is consistent with other narrative
3785 descriptions of the TOE.

3786 **10.9.1.2 Input**

3787 The evaluation evidence for this sub-activity is:

3788 a) the ST.

3789 **10.9.1.3 Action ASE_TSS.1.1E**

3790 ISO/IEC 15408-3 ASE_TSS.1.1C: *The TOE summary specification shall describe how the TOE meets*
3791 *each SFR.*

3792 **10.9.1.3.1 Work unit ASE_TSS.1-1**

3793 The evaluator **shall examine** the TOE summary specification to determine that it describes how
3794 the TOE meets each SFR.

3795 The evaluator determines that the TOE summary specification provides, for each SFR from the
3796 statement of security requirements, a description on how that SFR is met.

3797 The evaluator is reminded that the objective of each description is to provide potential consumers
3798 of the TOE with a high-level view of how the developer intends to satisfy each SFR and that the
3799 descriptions therefore should not be overly detailed. Often several SFRs will be implemented in
3800 one context; for instance a password authentication mechanism may implement FIA_UAU.1,
3801 FIA_SOS.1 and FIA_UID.1. Therefore usually the TSS will not consist of a long list with texts for each
3802 single SFR, but complete groups of SFRs may be covered by one text passage.

3803 For a composed TOE, the evaluator also determines that it is clear which component provides each
3804 SFR or how the components combine to meet each SFR.

3805 **10.9.1.4 Action ASE_TSS.1.2E**

3806 **10.9.1.4.1 Work unit ASE_TSS.1-2**

3807 The evaluator *shall examine* the TOE summary specification to determine that it is consistent with
3808 the TOE overview and the TOE description.

3809 The TOE overview, TOE description, and TOE summary specification describe the TOE in a
3810 narrative form at increasing levels of detail. These descriptions therefore need to be consistent.

3811 **10.9.2 Evaluation of sub-activity (ASE_TSS.2)**

3812 **10.9.2.1 Objectives**

3813 The objective of this sub-activity is to determine whether the TOE summary specification
3814 addresses all SFRs, whether the TOE summary specification addresses interference, logical
3815 tampering and bypass, and whether the TOE summary specification is consistent with other
3816 narrative descriptions of the TOE.

3817 **10.9.2.2 Input**

3818 The evaluation evidence for this sub-activity is:

3819 a) the ST.

3820 **10.9.2.3 Action ASE_TSS.2.1E**

3821 ISO/IEC 15408-3 ASE_TSS.2.1C: *The TOE summary specification shall describe how the TOE meets*
3822 *each SFR.*

3823 **10.9.2.3.1 Work unit ASE_TSS.2-1**

3824 The evaluator *shall examine* the TOE summary specification to determine that it describes how
3825 the TOE meets each SFR.

3826 The evaluator determines that the TOE summary specification provides, for each SFR from the
3827 statement of security requirements, a description on how that SFR is met.

3828 The evaluator is reminded that the objective of each description is to provide potential consumers
3829 of the TOE with a high-level view of how the developer intends to satisfy each SFR and that the
3830 descriptions therefore should not be overly detailed. Often several SFRs will be implemented in
3831 one context; for instance a password authentication mechanism may implement FIA_UAU.1,
3832 FIA_SOS.1 and FIA_UID.1. Therefore usually the TSS will not consist of a long list with texts for each
3833 single SFR, but complete groups of SFRs may be covered by one text passage.

3834 For a composed TOE, the evaluator also determines that it is clear which component provides each
3835 SFR or how the components combine to meet each SFR.

3836 ISO/IEC 15408-3 ASE_TSS.2.2C: *The TOE summary specification shall describe how the TOE protects*
3837 *itself against interference and logical tampering.*

3838 **10.9.2.3.2 Work unit ASE_TSS.2-2**

3839 The evaluator **shall examine** the TOE summary specification to determine that it describes how
3840 the TOE protects itself against interference and logical tampering.

3841 The evaluator is reminded that the objective of each description is to provide potential consumers
3842 of the TOE with a high-level view of how the developer intends to provide protection against
3843 interference and logical tampering and that the descriptions therefore should not be overly
3844 detailed.

3845 For a composed TOE, the evaluator also determines that it is clear which component provides the
3846 protection or how the components combine to provide protection.

3847 ISO/IEC 15408-3 ASE_TSS.2.3C: *The TOE summary specification shall describe how the TOE protects*
3848 *itself against bypass.*

3849 **10.9.2.3.3 Work unit ASE_TSS.2-3**

3850 The evaluator **shall examine** the TOE summary specification to determine that it describes how
3851 the TOE protects itself against bypass.

3852 The evaluator is reminded that the objective of each description is to provide potential consumers
3853 of the TOE with a high-level view of how the developer intends to provide protection against
3854 bypass and that the descriptions therefore should not be overly detailed.

3855 For a composed TOE, the evaluator also determines that it is clear which component provides the
3856 protection or how the components combine to provide protection.

3857 **10.9.2.4 Action ASE_TSS.2.2E**

3858 **10.9.2.4.1 Work unit ASE_TSS.2-4**

3859 The evaluator **shall examine** the TOE summary specification to determine that it is consistent with
3860 the TOE overview and the TOE description.

3861 The TOE overview, TOE description, and TOE summary specification describe the TOE in a
3862 narrative form at increasing levels of detail. These descriptions therefore need to be consistent.

3863 **10.10 Consistency of composite product Security Target (ASE_COMP)**

3864 The composite-specific work units defined in this chapter are intended to be integrated as
3865 refinements to the evaluation activities of the ASE class listed in the following table. The other
3866 activities of ASE class do not require composite-specific work units.

CC assurance family	Evaluation activity	Evaluation work unit	Composite-specific work
ASE_OBJ	ASE_OBJ.2.1C	ASE_OBJ.2-1	ASE_COMP.1-5
	ASE_OBJ.2.1C	ASE_OBJ.2-1	ASE_COMP.1-6

	ASE_OBJ.2.3C	ASE_OBJ.2-3	ASE_COMP.1-6
ASE_REQ	ASE_REQ.1.6C	ASE_REQ.1-10	ASE_COMP.1-1
	ASE_REQ.2.9C.	ASE_REQ.2-13	ASE_COMP.1-1
	ASE_REQ.1.6C	ASE_REQ.1-10	ASE_COMP.1-2
	ASE_REQ.2.9C	ASE_REQ.2-13	ASE_COMP.1-2
	ASE_REQ.2.8C	ASE_REQ.2-12	ASE_COMP.1-3
	ASE_REQ.2.3C	ASE_REQ.2-4	ASE_COMP.1-4

3867

3868 **10.10.1 Evaluation of sub-activity (ASE_COMP.1)**

3869 **10.10.1.1 Objectives**

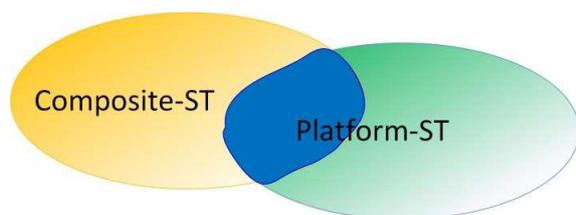
3870 The aim of this activity is to determine whether the Security Target of the composite product¹ does
 3871 not contradict the Security Target of the underlying platform².

3872 **10.10.1.2 Application notes**

3873 These application notes aid the developer to create as well as the evaluator to analyse a composite
 3874 Security Target and describe a general methodology for it. For detailed information/guidance
 3875 please refer to the single work units below. In order to create a composite Security Target the
 3876 developer should perform the following steps:

3877 Step 1: The developer formulates a preliminary Security Target for the composite product (the
 3878 Composite-ST) using the standard code of practice. The Composite-ST can be formulated
 3879 independently of the Security Target of the underlying platform (Platform-ST) – at least as long as
 3880 here are no formal PP conformance claims.

3881 Step 2: The developer determines the overlap between Platform-ST and Composite-ST through
 3882 analysing and comparing their TOE Security Functionality (TSF)^{3,4}:



3883

¹ denoted by Composite-ST in the following

² denoted by Platform-ST in the following. Generally, a Security Target expresses a security policy for the TOE defined.

³ because the TSF enforce the Security Target (together with organisational measures enforcing security objectives for the operational environment of the TOE).

⁴ The comparison shall be performed on the abstraction level of SFRs. If the developer defined security functionality groups (TSF-groups) in the TSS part of his Security Target, the evaluator should also consider them in order to get a better understanding for the context of the security services offered by the TOE.

- 3884 Step 3: The developer determines under which conditions they can trust in and rely on the
3885 Platform-TSF being used by the Composite-ST without a new examination.
- 3886 Having undertaken these steps the developer completes the preliminary Security Target for the
3887 composite product.
- 3888 It is not mandatory that the platform and the composite TOE are being certified according to same
3889 edition of the 15408. It is due to the fact that the application can rely on some security services of
3890 the platform, if (i) the assurance level of the platform covers the intended assurance level of the
3891 composite TOE and (ii) the platform's security certificate is valid and up-to-date. Equivalence of
3892 single assurance components (and, hence, of assurance levels) belonging to different ISO/IEC
3893 editions shall be established / acknowledged by the Composite Product Certification Body.
- 3894 If a PP conformance is claimed (e.g. composite ST claim conformance to a PP that claims
3895 conformance to a hardware PP), the consistency check can be reduced to the elements of the
3896 Security Target having not already been covered by these Protection Profiles.
- 3897 The fact of compliance to a PP is not sufficient to avoid inconsistencies. Assume the following
3898 situation, where → stands for "complies with"
- 3899 Composite-ST →SW PP →HW PP ←platform-ST
- 3900 The SW PP may require any kind of conformance, but this does not change the 'additional
3901 elements' that the platform-ST may introduce to the HW PP. In conclusion, these additions are not
3902 necessarily consistent with the composite-ST/SW PP additions: There is no scenario that ensures
3903 the consistency 'by construction'.
- 3904 Note that consistency may not be direct matching: e.g. objectives for the platform environment may
3905 become objectives for the composite TOE.
- 3906 **10.10.1.3 Action ASE_COMP.1.1E**
- 3907 The evaluator shall confirm that the information provided meets all requirements for content and
3908 presentation of evidence.
- 3909 *ISO/IEC 15408-3 ASE_COMP.1.1C: The statement of compatibility shall describe the separation of the*
3910 *Platform-TSF into relevant Platform-TSF being used by the Composite-ST and others.*
- 3911 **10.10.1.3.1 Work unit ASE_COMP.1-1**
- 3912 The evaluator shall check that the statement of compatibility describes the separation of the
3913 Platform-TSF into relevant Platform-TSF being used by the Composite-ST and others. 15
- 3914 Please note that TSF means 'TOE Security Functionality', whereby the TSF content is represented
3915 by SFRs¹². The respective TOE summary specification (TSS) shall provide, for each SFR, a
3916 description on how each SFR is met¹³. The evaluator shall use this description in order to
3917 understand the contextual frame of the SFRs.
- 3918 If the developer defined security functionality groups (TSF groups) in the TSS part of his Security
3919 Target as such contextual frame of the SFRs, the evaluator should also consider them in order to
3920 get a better understanding for the context of the security services offered by the TOE.
- 3921 This work unit relates to the Step 2 of the Application Notes above. In order to determine the
3922 intersection area the evaluator considers the list of the Platform-SFRs (given in the ST of the
3923 underlying platform) as single properties of the platform's security services.

3924 To give an example, let us assume that there are the following Platform-SFRs: Cryptographic
3925 operations FCS_COP.1/RSA, FCS_COP.1/AES, FCS_COP.1/EC as well as tamper-resistance
3926 FPT_PHP.3 and limited capabilities and availability FMT_LIM.1 and FMT_LIM.2

3927 These Platform-SFRs shall be separated in three groups:

3928 – **IP_SFR**: Irrelevant Platform-SFRs not being used by the Composite-ST.

3929 – **RP_SFR-SERV**: Relevant Platform-SFRs being used by the Composite-ST to implement a security
3930 service with associated TSFI. –

3931 **RP_SFR-MECH**: Relevant Platform-SFRs being used by the Composite-ST because of its security
3932 properties providing protection against attacks to the TOE as a whole and are addressed in
3933 ADV_ARC. These required security properties are a result of the security mechanisms and services
3934 that are implemented in the Platform TOE.

3935 The second and third group RP_SFR-SERV and RP_SFRMECH exactly represent the intersection
3936 area in question. For example, IP_SFR = {FCS_COP.1/AES}, RP_SFR-SERV= {FCS_COP.1/RSA,
3937 FCS_COP.1/EC} and RP_SFR-MECH = {FPT_PHP.3, FMT_LIM.1, FMT_LIM.2}, i.e. AES is not used by
3938 the composite TOE, but all other Platform-SFRs are used. However, the RP_SFR-MECH cannot be
3939 directly connected to SFRs in the Composite-ST.

3940 The size of the overlapping area (i.e. the content of the group RP_SFR-SERV and RP_SFR-MECH)
3941 results from the concrete properties of the Platform-ST and the Composite-ST. If the Composite-ST
3942 does not use any property of the Platform-ST and, hence, the intersection area is an empty set
3943 ($RP_SFRMECH \cap RP_SFR-SERV = \{\emptyset\}$), no further composite evaluation activities are necessary at
3944 all: In such a case there is a technical, but not a security composition.

3945 The result of this work unit shall be integrated to the result of ASE_REQ.1.6C/ ASE_REQ.1-10 (or
3946 the equivalent higher components if a higher assurance level is selected) and ASE_REQ.2.9C/
3947 ASE_REQ.2-13.

3948 **10.10.1.3.2 Work unit ASE_COMP.1-2**

3949 The evaluator shall examine the statement of compatibility to determine that the Platform-TSF
3950 being used by the Composite-ST is complete and consistent for the current composite TOE. 21 In
3951 order to determine the completeness of the list of the Platform-TSF being used by the Composite-
3952 ST, the evaluator shall verify that:

3953 • Platform-SFR = IP_SFR \cap RP_SFR-SERV \cap RP-SFR-MECH

3954 • Elements that belong to RP_SFR-SERV and RP-SFR-MECH are taken into account during
3955 the evaluation of the composite TOE. The IP-SFR are obviously part of the Platform-TOE
3956 but they are not considered during the evaluation of the composite TOE

3957 In order to determine the consistency of the list of the Platform TSF being used by the Composite-
3958 ST, the evaluator shall verify that there are no ambiguities and contradictory statements.

3959 The result of this work unit shall be integrated to the result of ASE_REQ.1.6C/ ASE_REQ.1-10 (or
3960 the equivalent higher components if a higher assurance level is selected) and ASE_REQ.2.9C/
3961 ASE_REQ.2-13.**ASE_COMP.1.2C**

3962 **10.10.1.3.3 Work unit ASE_COMP.1-3**

3963 The evaluator shall check that the security assurance requirements of the composite evaluation
3964 represent a subset of the security assurance requirements of the underlying platform.

3965 This work unit relates to the Step 2 of the Application Notes above. In order to ensure a sufficient
 3966 degree of trustworthiness of the Platform-TSF the evaluator compares the TOE security assurance
 3967 requirements¹⁵ of the composite evaluation with those of the underlying platform. The evaluator
 3968 decides that the degree of trustworthiness of the Platform-TSF is sufficient, if the Composite-SAR
 3969 represent a subset of the Platform-SAR:

3970 Platform-SAR \supseteq Composite-SAR,

3971 e.g. the EAL chosen for the composite evaluation does not exceed the EAL applied to the evaluation
 3972 of the platform.

3973 The result of this work unit shall be integrated to the result of ASE_REQ.2.8C/ ASE_REQ.2-12.

3974 **10.10.1.3.4 Work unit ASE_COMP.1-4**

3975 The evaluator shall examine the statement of compatibility to determine that all performed
 3976 operations on the relevant TOE security functional requirements of the platform are appropriate
 3977 for the Composite-ST.

3978 This work unit relates to Step 3 of the Application Notes above. The relevant TOE security
 3979 functional requirements of the platform comprise at least the elements of the group RP_SFRSERV
 3980 (cf. the work unit ASE_COMP.1-1) but also the RP-SFRMECH may be presented as relevant TOE
 3981 security functional requirements. The non-relevant TOE security functional requirements belong to
 3982 IP_SFR.

3983 In order to perform this work unit the evaluator compares single parameters of the relevant SFRs
 3984 of the platform with those of the composite evaluation. For example, the evaluator compares the
 3985 properties of the respective components FCS_COP.1/RSA and determines that the Composite-ST
 3986 requires a key length of 2048 bit and the Platform-ST enforces the RSA-function with a key length
 3987 of 1024 and 2048 bit, i.e. this parameter of the platform is appropriate for the Composite-ST. Note,
 3988 that the Composite-SFRs need not necessarily be the same as the Platform-SFRs, e.g. a trusted
 3989 channel (FTP_ITC.1) in the composite product can be built using an RSA implementation
 3990 (FCS_COP.1/RSA) of the platform.

3991 The result of this work unit shall be integrated to the result of ASE_REQ.2.3C/ ASE_REQ.2-4.

3992 **10.10.1.3.5 Work unit ASE_COMP.1-5**

3993 The evaluator shall examine the statement of compatibility to determine that the relevant TOE
 3994 security objectives of the Platform-ST are not contradictory to those of the Composite-ST.

3995 This work unit relates to Step 3 of the Application Notes above. The relevant TOE security
 3996 objectives of the Platform-ST are those that are mapped to the relevant SFRs of the Platform-ST (cf.
 3997 the work unit ASE_COMP.1-1).

3998 In order to perform this work unit the evaluator compares the relevant TOE security objectives of
 3999 the Platform-ST with those of the Composite-ST and determines whether they are not
 4000 contradictory.

4001 The result of this work unit shall be integrated to the result of ASE_OBJ.2.1C/ ASE_OBJ.2-1.

4002 **10.10.1.3.6 Work unit ASE_COMP.1-6**

4003 The evaluator shall examine the statement of compatibility to determine that the significant
 4004 security objectives for the operational environments of the Platform-ST are not contradictory to
 4005 those of the Composite-ST.

4006 This work unit relates to Step 3 of the Application Notes above. In order to determine which
4007 assumptions of the Platform-ST are significant for the Composite-ST the evaluator analyses the
4008 objectives for the environment of the Platform-ST and their separation in the following groups:

4009 • **IrOE:** The objectives for the environment being not relevant for the Composite-ST, e.g. the
4010 objectives for the environment about the developing and manufacturing phases of the
4011 platform.

4012 • **CfPOE:** The objectives for the environment being fulfilled by the Composite-ST
4013 automatically. Such objectives of the environment of the Platform-ST can always be
4014 assigned to the TOE security objectives of the Composite-ST. Due to this fact they will be
4015 fulfilled either by the Composite-SFR or by the Composite-SAR automatically. To give an
4016 example, let there be an Objective for the environment OE.Resp-Appl of the Platform-ST:
4017 'All User Data are owned by Smartcard Embedded Software. Therefore, it must be assumed
4018 that security relevant User Data (especially cryptographic keys) are treated by the
4019 Smartcard Embedded Software as defined for the specific application context' and a TOE
4020 security objective OT.Key_Secrecy of the Composite-ST: 'The secrecy of the signature
4021 private key used for signature generation is reasonably assured against attacks with a high
4022 attack potential.' If the private key is the only sensitive data element, then the Objective for
4023 the environment OE.Resp-Appl is covered by the TOE security objective OT.Key_Secrecy
4024 automatically.

4025 • **SgOE:** The remaining Objectives for the environment of the Platform-ST belonging neither
4026 to the group IrOE nor CfOE Exactly this group makes up the significant objectives for the
4027 environment for the Composite-ST, which shall be addressed in the Composite-ST.

4028 In order to accomplish this work unit the evaluator compares the significant security objectives for
4029 the operational environment of the Platform-ST with those of the Composite-ST and determines
4030 whether they are not contradictory. If necessary, the significant security objectives for the
4031 operational environment of the Platform-ST shall be included into the Composite-ST including the
4032 related assumptions from which the objectives for the environment are drawn. The inclusion is not
4033 necessary, if the Composite-ST already contains equivalent (or similar) security objectives
4034 (covering all relevant aspects) and assumptions.

4035 Since assurance of the development and manufacturing environment of the platform is confirmed
4036 by the platform certificate, the respective platform-objectives, if any, belong to the group IrOE

4037 Assurance of development and manufacturing environment is usually completely addressed by the
4038 assurance class ALC, and, hence, requires no explicit security objective.

4039 The result of this work unit shall be integrated to the result of ASE_OBJ.2.1C/ ASE_OBJ.2-1 and
4040 ASE_OBJ.2.3C/ ASE_OBJ.2-3.

4041

4042 **11 Class ADV: Development**

4043 **11.1 Introduction**

4044 The purpose of the development activity is to assess the design documentation in terms of its
4045 adequacy to understand how the TSF meets the SFRs and how the implementation of these SFRs
4046 cannot be tampered with or bypassed. This understanding is achieved through examination of
4047 increasingly refined descriptions of the TSF design documentation. Design documentation consists
4048 of a functional specification (which describes the interfaces of the TSF), a TOE design description
4049 (which describes the architecture of the TSF in terms of how it works in order to perform the
4050 functions related to the SFRs being claimed), and an implementation description (a source code
4051 level description). In addition, there is a security architecture description (which describes the
4052 architectural properties of the TSF to explain how its security enforcement cannot be compromised
4053 or bypassed), an internals description (which describes how the TSF was constructed in a manner
4054 that encourages understandability), and a security policy model (which formally describes the
4055 security policies enforced by the TSF).

4056 **11.2 Application notes**

4057 ISO/IEC 15408 requirements for design documentation are levelled by the amount, and detail of
4058 information provided, and the degree of formality of the presentation of the information. At lower
4059 levels, the most security-critical portions of the TSF are described with the most detail, while less
4060 security-critical portions of the TSF are merely summarised; added assurance is gained by
4061 increasing the amount of information about the most security-critical portions of the TSF, and
4062 increasing the details about the less security-critical portions. The most assurance is achieved
4063 when thorough details and information of all portions are provided.

4064 ISO/IEC 15408 considers a document's degree of formality (that is, whether it is informal or
4065 semiformal) to be hierarchical. An informal document is one that is expressed in a natural language.
4066 The methodology does not dictate the specific language that must be used; that issue is left for the
4067 scheme. The following paragraphs differentiate the contents of the different informal documents.

4068 A functional specification provides a description of the purpose and method-of-use of interfaces to
4069 the TSF. For example, if an operating system presents the user with a means of self-identification,
4070 of creating files, of modifying or deleting files, of setting permissions defining what other users may
4071 access files, and of communicating with remote machines, its functional specification would
4072 contain descriptions of each of these and how they are realised through interactions with the
4073 externally-visible interfaces to the TSF. If there is also audit functionality that detects and record
4074 the occurrences of such events, descriptions of this audit functionality would also be expected to be
4075 part of the functional specification; while this functionality is technically not directly invoked by
4076 the user at the external interface, it certainly is affected by what occurs at the user's external
4077 interface.

4078 A design description is expressed in terms of logical divisions (subsystems or modules) that each
4079 provide a comprehensible service or function. For example, a firewall might be composed of
4080 subsystems that deal with packet filtering, with remote administration, with auditing, and with
4081 connection-level filtering. The design description of the firewall would describe the actions that are
4082 taken, in terms of what actions each subsystem takes when an incoming packet arrives at the
4083 firewall.

4084 **11.3 Security Architecture (ADV_ARC)**

4085 **11.3.1 Evaluation of sub-activity (ADV_ARC.1)**

4086 **11.3.1.1 Objectives**

4087 The objective of this sub-activity is to determine whether the TSF is structured such that it cannot
4088 be tampered with or bypassed, and whether TSFs that provide security domains isolate those
4089 domains from each other.

4090 **11.3.1.2 Input**

4091 The evaluation evidence for this sub-activity is:

- 4092 a) the ST;
- 4093 b) the functional specification;
- 4094 c) the TOE design;
- 4095 d) the security architecture description;
- 4096 e) the implementation representation (if available);
- 4097 f) the operational user guidance;

4098 **11.3.1.3 Application notes**

4099 The notions of self-protection, domain separation, and non-bypassability are distinct from security
4100 functionality expressed in ISO/IEC 15408-2 SFRs because self-protection and non-bypassability
4101 largely have no directly observable interface at the TSF. Rather, they are properties of the TSF that
4102 are achieved through the design of the TOE, and enforced by the correct implementation of that
4103 design. Also, the evaluation of these properties is less straight-forward than the evaluation of
4104 mechanisms; it is more difficult to check for the absence of functionality than for its presence.
4105 However, the determination that these properties are being satisfied is just as critical as the
4106 determination that the mechanisms are properly implemented.

4107 The overall approach used is that the developer provides a TSF that meets the above-mentioned
4108 properties, and provides evidence (in the form of documentation) that can be analysed to show
4109 that the properties are indeed met. The evaluator has the responsibility for looking at the evidence
4110 and, coupled with other evidence delivered for the TOE, determining that the properties are
4111 achieved. The work units can be characterised as those detailing with what information has to be
4112 provided, and those dealing with the actual analysis the evaluator performs.

4113 The security architecture description describes how domains are defined and how the TSF keeps
4114 them separate. It describes what prevents untrusted processes from getting to the TSF and
4115 modifying it. It describes what ensures that all resources under the TSF's control are adequately
4116 protected and that all actions related to the SFRs are mediated by the TSF. It explains any role the
4117 environment plays in any of these (e.g. presuming it gets correctly invoked by its underlying
4118 environment, how is its security functionality invoked?). In short, it explains how the TOE is
4119 considered to be providing any kind of *security* service.

4120 The analyses the evaluator performs must be done in the context of all of the development
4121 evidence provided for the TOE, at the level of detail the evidence is provided. At lower assurance
4122 levels, there should not be the expectation that, for example, TSF self-protection is completely
4123 analysed, because only high-level design representations will be available. The evaluator also
4124 needs to be sure to use information gleaned from other portions of their analysis (e.g., analysis of

4125 the TOE design) in making their assessments for the properties being examined in the following
4126 work units.

4127 **11.3.1.4 Action ADV_ARC.1.1E**

4128 ISO/IEC 15408-3 ADV_ARC.1.1C: *The security architecture description shall be at a level of detail*
4129 *commensurate with the description of the SFR-enforcing abstractions described in the TOE design*
4130 *document.*

4131 **11.3.1.4.1 Work unit ADV_ARC.1-1**

4132 The evaluator **shall examine** the security architecture description to determine that the
4133 information provided in the evidence is presented at a level of detail commensurate with the
4134 descriptions of the SFR-enforcing abstractions contained in the functional specification and TOE
4135 design document.

4136 With respect to the functional specification, the evaluator should ensure that the self-protection
4137 functionality described cover those effects that are evident at the TSFI. Such a description might
4138 include protection placed upon the executable images of the TSF, and protection placed on objects
4139 (e.g., files used by the TSF). The evaluator ensures that the functionality that might be invoked
4140 through the TSFI is described.

4141 If Evaluation of sub-activity (ADV_TDS.1) or Evaluation of sub-activity (ADV_TDS.2) is included, the
4142 evaluator ensures the security architecture description contains information on how any
4143 subsystems that contribute to TSF domain separation work.

4144 If Evaluation of sub-activity (ADV_TDS.3) or higher is available, the evaluator ensures that the
4145 security architecture description also contains implementation-dependent information. For
4146 example, such a description might contain information pertaining to coding conventions for
4147 parameter checking that would prevent TSF compromises (e.g. buffer overflows), and information
4148 on stack management for call and return operations. The evaluator checks the descriptions of the
4149 mechanisms to ensure that the level of detail is such that there is little ambiguity between the
4150 description in the security architecture description and the implementation representation.

4151 The evaluator action related to this work unit is assigned a fail verdict if the security architecture
4152 description mentions any module, subsystem, or interface that is not described in the functional
4153 specification or TOE design document.

4154 ISO/IEC 15408-3 ADV_ARC.1.2C: *The security architecture description shall describe the security*
4155 *domains maintained by the TSF consistently with the SFRs.*

4156 **11.3.1.4.2 Work unit ADV_ARC.1-2**

4157 The evaluator **shall examine** the security architecture description to determine that it describes
4158 the security domains maintained by the TSF.

4159 Security domains refer to environments supplied by the TSF for use by potentially-harmful
4160 entities; for example, a typical secure operating system supplies a set of resources (address space,
4161 per-process environment variables) for use by processes with limited access rights and security
4162 properties. The evaluator determines that the developer's description of the security domains
4163 takes into account all of the SFRs claimed by the TOE.

4164 For some TOEs such domains do not exist because all of the interactions available to users are
4165 severely constrained by the TSF. A packet-filter firewall is an example of such a TOE. Users on the
4166 LAN or WAN do not interact with the TOE, so there need be no security domains; there are only
4167 data structures maintained by the TSF to keep the users' packets separated. The evaluator ensures
4168 that any claim that there are no domains is supported by the evidence and that no such domains
4169 are, in fact, available.

4170 ISO/IEC 15408-3 ADV_ARC.1.3C: *The security architecture description shall describe how the TSF*
4171 *initialisation process is secure.*

4172 **11.3.1.4.3 Work unit ADV_ARC.1-3**

4173 The evaluator *shall examine* the security architecture description to determine that the
4174 initialisation process preserves security.

4175 The information provided in the security architecture description relating to TSF initialisation is
4176 directed at the TOE components that are involved in bringing the TSF into an initial secure state
4177 (i.e. when all parts of the TSF are operational) when power-on or a reset is applied. This discussion
4178 in the security architecture description should list the system initialisation components and the
4179 processing that occurs in transitioning from the “down” state to the initial secure state.

4180 It is often the case that the components that perform this initialisation function are not accessible
4181 after the secure state is achieved; if this is the case then the security architecture description
4182 identifies the components and explains how they are not reachable by untrusted entities after the
4183 TSF has been established. In this respect, the property that needs to be preserved is that these
4184 components either 1) cannot be accessed by untrusted entities after the secure state is achieved, or
4185 2) if they provide interfaces to untrusted entities, these TSFI cannot be used to tamper with the
4186 TSF.

4187 The TOE components related to TSF initialisation, then, are treated themselves as part of the TSF,
4188 and analysed from that perspective. It should be noted that even though these are treated as part of
4189 the TSF, it is likely that a justification (as allowed by TSF internals (ADV_INT)) can be made that
4190 they do not have to meet the internal structuring requirements of ADV_INT.

4191 ISO/IEC 15408-3 ADV_ARC.1.4C: *The security architecture description shall demonstrate that the*
4192 *TSF protects itself from tampering.*

4193 **11.3.1.4.4 Work unit ADV_ARC.1-4**

4194 The evaluator *shall examine* the security architecture description to determine that it contains
4195 information sufficient to support a determination that the TSF is able to protect itself from
4196 tampering by untrusted active entities.

4197 “Self-protection” refers to the ability of the TSF to protect itself from manipulation from external
4198 entities that may result in changes to the TSF. For TOEs that have dependencies on other IT entities,
4199 it is often the case that the TOE uses services supplied by the other IT entities in order to perform
4200 its functions. In such cases, the TSF alone does not protect itself because it depends on the other IT
4201 entities to provide some of the protection. For the purposes of the security architecture description,
4202 the notion of *self-protection* applies only to the services provided by the TSF through its TSFI, and
4203 not to services provided by underlying IT entities that it uses.

4204 Self-protection is typically achieved by a variety of means, ranging from physical and logical
4205 restrictions on access to the TOE; to hardware-based means (e.g. “execution rings” and memory
4206 management functionality); to software-based means (e.g. boundary checking of inputs on a
4207 trusted server). The evaluator determines that all such mechanisms are described.

4208 The evaluator determines that the design description covers how user input is handled by the TSF
4209 in such a way that the TSF does not subject itself to being corrupted by that user input. For example,
4210 the TSF might implement the notion of privilege and protect itself by using privileged-mode
4211 routines to handle user input. The TSF might make use of processor-based separation mechanisms
4212 such as privilege levels or rings. The TSF might implement software protection constructs or
4213 coding conventions that contribute to implementing separation of software domains, perhaps by
4214 delineating user address space from system address space. And the TSF might have reliance its
4215 environment to provide some support to the protection of the TSF.

4216 All of the mechanisms contributing to the domain separation functions are described. The
 4217 evaluator should use knowledge gained from other evidence (functional specification, TOE design,
 4218 TSF internals description, other parts of the security architecture description, or implementation
 4219 representation, as included in the assurance package for the TOE) in determining if any
 4220 functionality contributing to self-protection was described that is not present in the security
 4221 architecture description.

4222 Accuracy of the description of the self-protection mechanisms is the property that the description
 4223 faithfully describes what is implemented. The evaluator should use other evidence (functional
 4224 specification, TOE design, TSF Internals documentation, other parts of the security architecture
 4225 description, implementation representation, as included in the ST for the TOE) in determining
 4226 whether there are discrepancies in any descriptions of the self-protection mechanisms. If
 4227 Implementation representation (ADV_IMP) is included in the assurance package for the TOE, the
 4228 evaluator will choose a sample of the implementation representation; the evaluator should also
 4229 ensure that the descriptions are accurate for the sample chosen. If an evaluator cannot understand
 4230 how a certain self-protection mechanism works or could work in the system architecture, it may be
 4231 the case that the description is not accurate.

4232 ISO/IEC 15408-3 ADV_ARC.1.5C: *The security architecture description shall demonstrate that the*
 4233 *TSF prevents bypass of the SFR-enforcing functionality.*

4234 **11.3.1.4.5 Work unit ADV_ARC.1-5**

4235 The evaluator ***shall examine*** the security architecture description to determine that it presents an
 4236 analysis that adequately describes how the SFR-enforcing mechanisms cannot be bypassed.

4237 Non-bypassability is a property that the security functionality of the TSF (as specified by the SFRs)
 4238 is always invoked. For example, if access control to files is specified as a capability of the TSF via an
 4239 SFR, there must be no interfaces through which files can be accessed without invoking the TSF's
 4240 access control mechanism (such as an interface through which a raw disk access takes place).

4241 Describing how the TSF mechanisms cannot be bypassed generally requires a systematic argument
 4242 based on the TSF and the TSFIs. The description of how the TSF works (contained in the design
 4243 decomposition evidence, such as the functional specification, TOE design documentation) - along
 4244 with the information in the TSS - provides the background necessary for the evaluator to
 4245 understand what resources are being protected and what security functions are being provided.
 4246 The functional specification provides descriptions of the TSFIs through which the
 4247 resources/functions are accessed.

4248 The evaluator assesses the description provided (and other information provided by the developer,
 4249 such as the functional specification) to ensure that no available interface can be used to bypass the
 4250 TSF. This means that every available interface must be either unrelated to the SFRs that are
 4251 claimed in the ST (and does not interact with anything that is used to satisfy SFRs) or else uses the
 4252 security functionality that is described in other development evidence in the manner described.
 4253 For example, a game would likely be unrelated to the SFRs, so there must be an explanation of how
 4254 it cannot affect security. Access to user data, however, is likely to be related to access control SFRs,
 4255 so the explanation would describe how the security functionality works when invoked through the
 4256 data-access interfaces. Such a description is needed for every available interface.

4257 An example of a description follows. Suppose the TSF provides file protection. Further suppose that
 4258 although the "traditional" system call TSFIs for open, read, and write invoke the file protection
 4259 mechanism described in the TOE design, there exists a TSFI that allows access to a batch job facility
 4260 (creating batch jobs, deleting jobs, modifying unprocessed jobs). The evaluator should be able to
 4261 determine from the vendor-provided description that this TSFI invokes the same protection
 4262 mechanisms as do the "traditional" interfaces. This could be done, for example, by referencing the
 4263 appropriate subclauses of the TOE design that discuss *how* the batch job facility TSFI achieves its
 4264 security objectives.

4265 Using this same example, suppose there is a TSFI whose sole purpose is to display the time of day.
4266 The evaluator should determine that the description adequately argues that this TSFI is not
4267 capable of manipulating any protected resources and should not invoke any security functionality.

4268 Another example of bypass is when the TSF is supposed to maintain confidentiality of a
4269 cryptographic key (one is allowed to use it for cryptographic operations, but is not allowed to
4270 read/write it). If an attacker has direct physical access to the device, they might be able to examine
4271 side-channels such as the power usage of the device, the exact timing of the device, or even any
4272 electromagnetic emanations of the device and, from this, infer the key.

4273 If such side-channels may be present, the demonstration should address the mechanisms that
4274 prevent these side-channels from occurring, such as random internal clocks, dual-line technology
4275 etc. Verification of these mechanisms would be verified by a combination of purely design-based
4276 arguments and testing.

4277 For a final example using security functionality rather than a protected resource, consider an ST
4278 that contains **FCO_NRO.2 Enforced proof of origin**, which requires that the TSF provides evidence
4279 of origination for information types specified in the ST. Suppose that the “information types”
4280 included all information that is sent by the TOE via e-mail. In this case, the evaluator should
4281 examine the description to ensure that all TSFI that can be invoked to send e-mail perform the
4282 “evidence of origination generation” function are detailed. The description might point to user
4283 guidance to show all places where e-mail can originate (e.g., e-mail program, notification from
4284 scripts/batch jobs) and then how each of these places invokes the evidence generation function.

4285 The evaluator should also ensure that the description is comprehensive, in that each interface is
4286 analysed with respect to the entire set of claimed SFRs. This may require the evaluator to examine
4287 supporting information (functional specification, TOE design, other parts of the security
4288 architecture description, operational user guidance, and perhaps even the implementation
4289 representation, as provided for the TOE) to determine that the description has correctly capture all
4290 aspects of an interface. The evaluator should consider what SFRs each TSFI might affect (from the
4291 description of the TSFI and its implementation in the supporting documentation), and then
4292 examine the description to determine whether it covers those aspects.

4293 **11.4 Functional specification (ADV_FSP)**

4294 **11.4.1 Evaluation of sub-activity (ADV_FSP.1)**

4295 **11.4.1.1 Objectives**

4296 The objective of this sub-activity is to determine whether the developer has provided a high-level
4297 description of at least the SFR-enforcing and SFR-supporting TSFIs, in terms of descriptions of their
4298 parameters. There is no other required evidence that can be expected to be available to measure
4299 the accuracy of these descriptions; the evaluator merely ensures the descriptions seem plausible.

4300 **11.4.1.2 Input**

4301 The evaluation evidence for this sub-activity is:

- 4302 a) the ST;
- 4303 b) the functional specification;
- 4304 c) the operational user guidance;

4305 **11.4.1.3 Action ADV_FSP.1.1E**

4306 ISO/IEC 15408-3 ADV_FSP.1.1C: *The functional specification shall describe the purpose and method*
4307 *of use for each SFR-enforcing and SFR-supporting TSFI.*

4308 **11.4.1.3.1 Work unit ADV_FSP.1-1**

4309 The evaluator *shall examine* the functional specification to determine that it states the purpose of
4310 each SFR-supporting and SFR-enforcing TSFI.

4311 The purpose of a TSFI is a general statement summarising the functionality provided by the
4312 interface. It is not intended to be a complete statement of the actions and results related to the
4313 interface, but rather a statement to help the reader understand in general what the interface is
4314 intended to be used for. The evaluator should not only determine that the purpose exists, but also
4315 that it accurately reflects the TSFI by taking into account other information about the interface,
4316 such as the description of the parameters; this can be done in association with other work units for
4317 this component.

4318 If an action available through an interface plays a role in enforcing any security policy on the TOE
4319 (that is, if one of the actions of the interface can be traced to one of the SFRs levied on the TSF),
4320 then that interface is *SFR-enforcing*. Such policies are not limited to the access control policies, but
4321 also refer to any functionality specified by one of the SFRs contained in the ST. Note that it is
4322 possible that an interface may have various actions and results, some of which may be SFR-
4323 enforcing and some of which may not.

4324 Interfaces to (or actions available through an interface relating to) actions that SFR-enforcing
4325 functionality depends on, but need only to function correctly in order for the security policies of
4326 the TOE to be preserved, are termed *SFR supporting*. Interfaces to actions on which SFR-enforcing
4327 functionality has no dependence are termed *SFR non-interfering*.

4328 It should be noted that in order for an interface to be SFR supporting or SFR non-interfering it must
4329 have *no* SFR-enforcing actions or results. In contrast, an SFR-enforcing interface may have SFR-
4330 supporting actions (for example, the ability to set the system clock may be an SFR-enforcing action
4331 of an interface, but if that same interface is used to display the system date that action may only be
4332 SFR supporting). An example of a purely SFR-supporting interface is a system call interface that is
4333 used both by untrusted users and by a portion of the TSF that is running in user mode.

4334 At this level, it is unlikely that a developer will have expended effort to label interfaces as SFR-
4335 enforcing and SFR-supporting. In the case that this has been done, the evaluator should verify to
4336 the extent that supporting documentation (e.g., operational user guidance) allows that this
4337 identification is correct. Note that this identification activity is necessary for several work units for
4338 this component.

4339 In the more likely case that the developer has not labelled the interfaces, the evaluator must
4340 perform their own identification of the interfaces first, and then determine whether the required
4341 information (for this work unit, the purpose) is present. Again, because of the lack of supporting
4342 evidence this identification will be difficult and have low assurance that all appropriate interfaces
4343 have been correctly identified, but nonetheless the evaluator examines other evidence available for
4344 the TOE to ensure as complete coverage as is possible.

4345 **11.4.1.3.2 Work unit ADV_FSP.1-2**

4346 The evaluator *shall examine* the functional specification to determine that the method of use for
4347 each SFR-supporting and SFR-enforcing TSFI is given.

4348 See work unit ADV_FSP.1-1 for a discussion on the identification of SFR-supporting and SFR-
4349 enforcing TSFI.

4350 The method of use for a TSFI summarises how the interface is manipulated in order to invoke the
4351 actions and obtain the results associated with the TSFI. The evaluator should be able to determine,
4352 from reading this material in the functional specification, how to use each interface. This does not
4353 necessarily mean that there needs to be a separate method of use for each TSFI, as it may be
4354 possible to describe in general how kernel calls are invoked, for instance, and then identify each

4355 interface using that general style. Different types of interfaces will require different method of use
4356 specifications. APIs, network protocol interfaces, system configuration parameters, and hardware
4357 bus interfaces all have very different methods of use, and this should be taken into account by the
4358 developer when developing the functional specification, as well as by the evaluator evaluating the
4359 functional specification.

4360 For administrative interfaces, whose functionality is documented as being inaccessible to
4361 untrusted users, the evaluator ensures that the method of making the functions inaccessible is
4362 described in the functional specification. It should be noted that this inaccessibility needs to be
4363 tested by the developer in their test suite.

4364 ISO/IEC 15408-3 ADV_FSP.1.2C: *The functional specification shall identify all parameters associated*
4365 *with each SFR-enforcing and SFR-supporting TSFI.*

4366 **11.4.1.3.3 Work unit ADV_FSP.1-3**

4367 The evaluator *shall examine* the presentation of the TSFI to determine that it identifies all
4368 parameters associated with each SFR-enforcing and SFR-supporting TSFI.

4369 See work unit ADV_FSP.1-1 for a discussion on the identification of SFR-supporting and SFR-
4370 enforcing TSFI.

4371 The evaluator examines the functional specification to ensure that all of the parameters are
4372 described for identified TSFI. Parameters are explicit inputs or outputs to an interface that control
4373 the behaviour of that interface. For examples, parameters are the arguments supplied to an API;
4374 the various fields in packet for a given network protocol; the individual key values in the Windows
4375 Registry; the signals across a set of pins on a chip; etc.

4376 While difficult to obtain much assurance that all parameters for the applicable TSFI have been
4377 identified, the evaluator should also check other evidence provided for the evaluation (e.g.,
4378 operational user guidance) to see if behaviour or additional parameters are described there but not
4379 in the functional specification.

4380 ISO/IEC 15408-3 ADV_FSP.1.3C: *The functional specification shall provide rationale for the implicit*
4381 *categorisation of interfaces as SFR-non-interfering.*

4382 **11.4.1.3.4 Work unit ADV_FSP.1-4**

4383 The evaluator *shall examine* the rationale provided by the developer for the implicit
4384 categorisation of interfaces as SFR-non-interfering to determine that it is accurate.

4385 In the case where the developer has provided adequate documentation to perform the analysis
4386 called for by the rest of the work units for this component without explicitly identifying SFR-
4387 enforcing and SFR-supporting interfaces, this work unit should be considered satisfied.

4388 This work unit is intended to apply to cases where the developer has not described a portion of the
4389 TSFI, claiming that it is SFR-non-interfering and therefore not subject to other requirements of this
4390 component. In such a case, the developer provides a rationale for this characterisation in sufficient
4391 detail such that the evaluator understands the rationale, the characteristics of the interfaces
4392 affected (e.g., their high-level function with respect to the TOE, such as “colour palette
4393 manipulation”), and that the claim that these are SFR-non-interfering is supported. Given the level
4394 of assurance the evaluator should not expect more detail than is provided for the SFR-enforcing or
4395 SFR-supporting interfaces, and in fact the detail should be much less. In most cases, individual
4396 interfaces should not need to be addressed in the developer-provided rationale subclause.

4397 ISO/IEC 15408-3 ADV_FSP.1.4C: *The tracing shall demonstrate that the SFRs trace to TSFIs in the*
4398 *functional specification.*

4399 **11.4.1.3.5 Work unit ADV_FSP.1-5**

4400 The evaluator *shall check* that the tracing links the SFRs to the corresponding TSFIs.

4401 The tracing is provided by the developer to serve as a guide to which SFRs are related to which
4402 TSFIs. This tracing can be as simple as a table; it is used as input to the evaluator for use in the
4403 following work units, in which the evaluator verifies its completeness and accuracy.

4404 **11.4.1.4 Action ADV_FSP.1.2E**4405 **11.4.1.4.1 Work unit ADV_FSP.1-6**

4406 The evaluator *shall examine* the functional specification to determine that it is a complete
4407 instantiation of the SFRs.

4408 To ensure that all SFRs are covered by the functional specification, as well as the test coverage
4409 analysis, the evaluator may build upon the developer's tracing (see ADV_FSP.1-5 a map between
4410 the TOE security functional requirements and the TSFI). Note that this map may have to be at a
4411 level of detail below the component or even element level of the requirements, because of
4412 operations (assignments, refinements, selections) performed on the functional requirement by the
4413 ST author.

4414 For example, the FDP_ACC.1 component contains an element with assignments. If the ST contained,
4415 for instance, ten rules in the FDP_ACC.1 assignment, and these ten rules were covered by three
4416 different TSFI, it would be inadequate for the evaluator to map FDP_ACC.1 to TSFI A, B, and C and
4417 claim they had completed the work unit. Instead, the evaluator would map FDP_ACC.1 (rule 1) to
4418 TSFI A; FDP_ACC.1 (rule 2) to TSFI B; etc. It might also be the case that the interface is a wrapper
4419 interface (e.g., IOCTL), in which case the mapping would need to be specific to certain set of
4420 parameters for a given interface.

4421 The evaluator must recognise that for requirements that have little or no manifestation at the TSF
4422 boundary (e.g., FDP_RIP) it is not expected that they completely map those requirements to the
4423 TSFI. The analysis for those requirements will be performed in the analysis for the TOE design
4424 (ADV_TDS) when included in the ST. It is also important to note that since the parameters
4425 associated with TSFIs must be fully specified, the evaluator should be able to determine if all
4426 aspects of an SFR appear to be implemented at the interface level.

4427 **11.4.1.4.2 Work unit ADV_FSP.1-7**

4428 The evaluator *shall examine* the functional specification to determine that it is an accurate
4429 instantiation of the SFRs.

4430 For each functional requirement in the ST that results in effects visible at the TSF boundary, the
4431 information in the associated TSFI for that requirement specifies the required functionality
4432 described by the requirement. For example, if the ST contains a requirement for access control lists,
4433 and the only TSFI that map to that requirement specify functionality for Unix-style protection bits,
4434 then the functional specification is not accurate with respect to the requirements.

4435 The evaluator must recognise that for requirements that have little or no manifestation at the TSF
4436 boundary (e.g., FDP_RIP) it is not expected that the evaluator completely map those requirements
4437 to the TSFI. The analysis for those requirements will be performed in the analysis for the TOE
4438 design (ADV_TDS) when included in the ST.

4439 **11.4.2 Evaluation of sub-activity (ADV_FSP.2)**

4440 **11.4.2.1 Objectives**

4441 The objective of this sub-activity is to determine whether the developer has provided a description
4442 of the TSFIs in terms of their purpose, method of use, and parameters. In addition, the SFR-
4443 enforcing actions, results and error messages of each TSFI that is SFR-enforcing are also described.

4444 **11.4.2.2 Input**

4445 The evaluation evidence for this sub-activity that is required by the work-units is:

- 4446 a) the ST;
- 4447 b) the functional specification;
- 4448 c) the TOE design.

4449 The evaluation evidence for this sub-activity that is used if included in the ST for the TOE is:

- 4450 a) the security architecture description;
- 4451 b) the operational user guidance;

4452 **11.4.2.3 Action ADV_FSP.2.1E**

4453 ISO/IEC 15408-3 ADV_FSP.2.1C: *The functional specification shall completely represent the TSF.*

4454 **11.4.2.3.1 Work unit ADV_FSP.2-1**

4455 The evaluator ***shall examine*** the functional specification to determine that the TSF is fully
4456 represented.

4457 The identification of the TSFI is a necessary prerequisite to all other activities in this sub-activity.
4458 The TSF must be identified (done as part of the TOE design (ADV_TDS) work units) in order to
4459 identify the TSFI. This activity can be done at a high level to ensure that no large groups of
4460 interfaces have been missed (network protocols, hardware interfaces, configuration files), or at a
4461 low level as the evaluation of the functional specification proceeds.

4462 In making an assessment for this work unit, the evaluator determines that all portions of the TSF
4463 are addressed in terms of the interfaces listed in the functional specification. All portions of the TSF
4464 should have a corresponding interface description, or if there are no corresponding interfaces for a
4465 portion of the TSF, the evaluator determines that that is acceptable.

4466 ISO/IEC 15408-3 ADV_FSP.2.2C: *The functional specification shall describe the purpose and method*
4467 *of use for all TSFI.*

4468 **11.4.2.3.2 Work unit ADV_FSP.2-2**

4469 The evaluator ***shall examine*** the functional specification to determine that it states the purpose of
4470 each TSFI.

4471 The purpose of a TSFI is a general statement summarising the functionality provided by the
4472 interface. It is not intended to be a complete statement of the actions and results related to the
4473 interface, but rather a statement to help the reader understand in general what the interface is
4474 intended to be used for. The evaluator should not only determine that the purpose exists, but also
4475 that it accurately reflects the TSFI by taking into account other information about the interface,
4476 such as the description of actions and error messages.

4477 **11.4.2.3.3 Work unit ADV_FSP.2-3**

4478 The evaluator *shall examine* the functional specification to determine that the method of use for
4479 each TSFI is given.

4480 The method of use for a TSFI summarises how the interface is manipulated in order to invoke the
4481 actions and obtain the results associated with the TSFI. The evaluator should be able to determine,
4482 from reading this material in the functional specification, how to use each interface. This does not
4483 necessarily mean that there needs to be a separate method of use for each TSFI, as it may be
4484 possible to describe in general how kernel calls are invoked, for instance, and then identify each
4485 interface using that general style. Different types of interfaces will require different method of use
4486 specifications. APIs, network protocol interfaces, system configuration parameters, and hardware
4487 bus interfaces all have very different methods of use, and this should be taken into account by the
4488 developer when developing the functional specification, as well as by the evaluator evaluating the
4489 functional specification.

4490 For administrative interfaces, whose functionality is documented as being inaccessible to
4491 untrusted users, the evaluator ensures that the method of making the functions inaccessible is
4492 described in the functional specification. It should be noted that this inaccessibility needs to be
4493 tested by the developer in their test suite.

4494 The evaluator should not only determine that the set of method of use descriptions exist, but also
4495 that they accurately cover each TSFI.

4496 ISO/IEC 15408-3 ADV_FSP.2.3C: *The functional specification shall identify and describe all*
4497 *parameters associated with each TSFI.*

4498 **11.4.2.3.4 Work unit ADV_FSP.2-4**

4499 The evaluator *shall examine* the presentation of the TSFI to determine that it completely identifies
4500 all parameters associated with every TSFI.

4501 The evaluator examines the functional specification to ensure that all of the parameters are
4502 described for each TSFI. Parameters are explicit inputs or outputs to an interface that control the
4503 behaviour of that interface. For examples, parameters are the arguments supplied to an API; the
4504 various fields in packet for a given network protocol; the individual key values in the Windows
4505 Registry; the signals across a set of pins on a chip; etc.

4506 In order to determine that all of the parameters are present in the TSFI, the evaluator should
4507 examine the rest of the interface description (actions, error messages, etc.) to determine if the
4508 effects of the parameter are accounted for in the description. The evaluator should also check other
4509 evidence provided for the evaluation (e.g., TOE design, security architecture description,
4510 operational user guidance, implementation representation) to see if behaviour or additional
4511 parameters are described there but not in the functional specification.

4512 **11.4.2.3.5 Work unit ADV_FSP.2-5**

4513 The evaluator *shall examine* the presentation of the TSFI to determine that it completely and
4514 accurately describes all parameters associated with every TSFI.

4515 Once all of the parameters have been identified, the evaluator needs to ensure that they are
4516 accurately described, and that the description of the parameters is complete. A parameter
4517 description tells what the parameter is in some meaningful way. For instance, the interface *foo(i)*
4518 could be described as having "parameter i which is an integer"; this is not an acceptable parameter
4519 description. A description such as "parameter i is an integer that indicates the number of users
4520 currently logged in to the system" is much more acceptable.

4521 In order to determine that the description of the parameters is complete, the evaluator should
4522 examine the rest of the interface description (purpose, method of use, actions, error messages, etc.)
4523 to determine if the descriptions of the parameter(s) are accounted for in the description. The
4524 evaluator should also check other evidence provided (e.g., TOE design, architectural design,
4525 operational user guidance, implementation representation) to see if behaviour or additional
4526 parameters are described there but not in the functional specification.

4527 ISO/IEC 15408-3 ADV_FSP.2.4C: *For each SFR-enforcing TSFI, the functional specification shall*
4528 *describe the SFR-enforcing actions associated with the TSFI.*

4529 **11.4.2.3.6 Work unit ADV_FSP.2-6**

4530 The evaluator **shall examine** the presentation of the TSFI to determine that it completely and
4531 accurately describes the SFR-enforcing actions associated with the SFR-enforcing TSFIs.

4532 If an action available through an interface can be traced to one of the SFRs levied on the TSF, then
4533 that interface is *SFR-enforcing*. Such policies are not limited to the access control policies, but also
4534 refer to any functionality specified by one of the SFRs contained in the ST. Note that it is possible
4535 that an interface may have various actions and results, some of which may be SFR-enforcing and
4536 some of which may not.

4537 The developer is not required to “label” interfaces as SFR-enforcing, and likewise is not required to
4538 identify actions available through an interface as SFR-enforcing. It is the evaluator's responsibility
4539 to examine the evidence provided by the developer and determine that the required information is
4540 present. In the case where the developer has identified the SFR-enforcing TSFI and SFR-enforcing
4541 actions available through those TSFI, the evaluator must judge completeness and accuracy based
4542 on other information supplied for the evaluation (e.g., TOE design, security architecture description,
4543 operational user guidance), and on the other information presented for the interfaces (parameters
4544 and parameter descriptions, error messages, etc.).

4545 In this case (where the developer has provided only the SFR-enforcing information for SFR-
4546 enforcing TSFI) the evaluator also ensures that no interfaces have been mis-categorised. This is
4547 done by examining other information supplied for the evaluation (e.g., TOE design, security
4548 architecture description, operational user guidance), and the other information presented for the
4549 interfaces (parameters and parameter descriptions, for example) not labelled as SFR-enforcing.

4550 In the case where the developer has provided the same level of information on all interfaces, the
4551 evaluator performs the same type of analysis mentioned in the previous paragraphs. The evaluator
4552 should determine which interfaces are SFR-enforcing and which are not, and subsequently ensure
4553 that the SFR-enforcing aspects of the SFR-enforcing actions are appropriately described.

4554 The SFR-enforcing actions are those that are visible at any external interface and that provide for
4555 the enforcement of the SFRs being claimed. For example, if audit requirements are included in the
4556 ST, then audit-related actions would be SFR-enforcing and therefore must be described, even if the
4557 result of that action is generally not visible through the invoked interface (as is often the case with
4558 audit, where a user action at one interface would produce an audit record visible at another
4559 interface).

4560 The level of description that is required is that sufficient for the reader to understand what role the
4561 TSFI actions play with respect to the SFR. The evaluator should keep in mind that the description
4562 should be detailed enough to support the generation (and assessment) of test cases against that
4563 interface. If the description is unclear or lacking detail such that meaningful testing cannot be
4564 conducted against the TSFI, it is likely that the description is inadequate.

4565 ISO/IEC 15408-3 ADV_FSP.2.5C: *For each SFR-enforcing TSFI, the functional specification shall*
4566 *describe direct error messages resulting from processing associated with the SFR-enforcing actions.*

4567 **11.4.2.3.7 Work unit ADV_FSP.2-7**

4568 The evaluator *shall examine* the presentation of the TSFI to determine that it completely and
 4569 accurately describes error messages that may result from SFR-enforcing actions associated with
 4570 each SFR-enforcing TSFI.

4571 This work unit should be performed in conjunction with, or after, work unit ADV_FSP.2-6 in order
 4572 to ensure the set of SFR-enforcing TSFI and SFR-enforcing actions is correctly identified. The
 4573 developer may provide more information than is required (for example, all error messages
 4574 associated with each interface), in which the case the evaluator should restrict their assessment of
 4575 completeness and accuracy to only those that they determine to be associated with SFR-enforcing
 4576 actions of SFR-enforcing TSFI.

4577 Errors can take many forms, depending on the interface being described. For an API, the interface
 4578 itself may return an error code, set a global error condition, or set a certain parameter with an
 4579 error code. For a configuration file, an incorrectly configured parameter may cause an error
 4580 message to be written to a log file. For a hardware PCI card, an error condition may raise a signal
 4581 on the bus, or trigger an exception condition to the CPU.

4582 Errors (and the associated error messages) come about through the invocation of an interface. The
 4583 processing that occurs in response to the interface invocation may encounter error conditions,
 4584 which trigger (through an implementation-specific mechanism) an error message to be generated.
 4585 In some instances, this may be a return value from the interface itself; in other instances a global
 4586 value may be set and checked after the invocation of an interface. It is likely that a TOE will have a
 4587 number of low-level error messages that may result from fundamental resource conditions, such as
 4588 "disk full" or "resource locked". While these error messages may map to a large number of TSFI,
 4589 they could be used to detect instances where detail from an interface description has been omitted.
 4590 For instance, a TSFI that produces a "disk full" message, but has no obvious description of why that
 4591 TSFI should cause an access to the disk in its description of actions, might cause the evaluator to
 4592 examine other evidence (Security Architecture (ADV_ARC), TOE design (ADV_TDS)) related that
 4593 TSFI to determine if the description is accurate.

4594 In order to determine that the description of the error messages of a TSFI is accurate and complete,
 4595 the evaluator measures the interface description against the other evidence provided for the
 4596 evaluation (e.g., TOE design, security architecture description, operational user guidance), as well
 4597 as other evidence available for that TSFI (parameters, analysis from work unit ADV_FSP.2-6).

4598 ISO/IEC 15408-3 ADV_FSP.2.6C: *The tracing shall demonstrate that the SFRs trace to TSFIs in the*
 4599 *functional specification.*

4600 **11.4.2.3.8 Work unit ADV_FSP.2-8**

4601 The evaluator *shall check* that the tracing links the SFRs to the corresponding TSFIs.

4602 The tracing is provided by the developer to serve as a guide to which SFRs are related to which
 4603 TSFIs. This tracing can be as simple as a table; it is used as input to the evaluator for use in the
 4604 following work units, in which the evaluator verifies its completeness and accuracy.

4605 **11.4.2.4 Action ADV_FSP.2.2E**4606 **11.4.2.4.1 Work unit ADV_FSP.2-9**

4607 The evaluator *shall examine* the functional specification to determine that it is a complete
 4608 instantiation of the SFRs.

4609 To ensure that all SFRs are covered by the functional specification, as well as the test coverage
 4610 analysis, the evaluator may build upon the developer's tracing (see ADV_FSP.2-8 a map between
 4611 the TOE security functional requirements and the TSFI. Note that this map may have to be at a level

4612 of detail below the component or even element level of the requirements, because of operations
4613 (assignments, refinements, selections) performed on the functional requirement by the ST author.

4614 For example, the FDP_ACC.1 component contains an element with assignments. If the ST contained,
4615 for instance, ten rules in the FDP_ACC.1 assignment, and these ten rules were covered by three
4616 different TSFI, it would be inadequate for the evaluator to map FDP_ACC.1 to TSFI A, B, and C and
4617 claim they had completed the work unit. Instead, the evaluator would map FDP_ACC.1 (rule 1) to
4618 TSFI A; FDP_ACC.1 (rule 2) to TSFI B; etc. It might also be the case that the interface is a wrapper
4619 interface (e.g., IOCTL), in which case the mapping would need to be specific to certain set of
4620 parameters for a given interface.

4621 The evaluator must recognise that for requirements that have little or no manifestation at the TSF
4622 boundary (e.g., FDP_RIP) it is not expected that they completely map those requirements to the
4623 TSFI. The analysis for those requirements will be performed in the analysis for the TOE design
4624 (ADV_TDS) when included in the ST. It is also important to note that since the parameters, actions,
4625 and error messages associated with TSFIs must be fully specified, the evaluator should be able to
4626 determine if all aspects of an SFR appear to be implemented at the interface level.

4627 **11.4.2.4.2 Work unit ADV_FSP.2-10**

4628 The evaluator *shall examine* the functional specification to determine that it is an accurate
4629 instantiation of the SFRs.

4630 For each functional requirement in the ST that results in effects visible at the TSF boundary, the
4631 information in the associated TSFI for that requirement specifies the required functionality
4632 described by the requirement. For example, if the ST contains a requirement for access control lists,
4633 and the only TSFI that map to that requirement specify functionality for Unix-style protection bits,
4634 then the functional specification is not accurate with respect to the requirements.

4635 The evaluator must recognise that for requirements that have little or no manifestation at the TSF
4636 boundary (e.g., FDP_RIP) it is not expected that the evaluator completely map those requirements
4637 to the TSFI. The analysis for those requirements will be performed in the analysis for the TOE
4638 design (ADV_TDS) when included in the ST.

4639 **11.4.3 Evaluation of sub-activity (ADV_FSP.3)**

4640 **11.4.3.1 Objectives**

4641 The objective of this sub-activity is to determine whether the developer has provided a description
4642 of the TSFIs in terms of their purpose, method of use, and parameters. In addition, the actions,
4643 results and error messages of each TSFI are also described sufficiently that it can be determined
4644 whether they are SFR-enforcing, with the SFR-enforcing TSFI being described in more detail than
4645 other TSFIs.

4646 **11.4.3.2 Input**

4647 The evaluation evidence for this sub-activity that is required by the work-units is:

- 4648 a) the ST;
- 4649 b) the functional specification;
- 4650 c) the TOE design.

4651 The evaluation evidence for this sub-activity that is used if included in the ST for the TOE is:

- 4652 a) the security architecture description;

4653 b) the implementation representation;

4654 c) the TSF internals description;

4655 d) the operational user guidance;

4656 **11.4.3.3 Action ADV_FSP.3.1E**

4657 ISO/IEC 15408-3 ADV_FSP.3.1C: *The functional specification shall completely represent the TSF.*

4658 **11.4.3.3.1 Work unit ADV_FSP.3-1**

4659 The evaluator **shall examine** the functional specification to determine that the TSF is fully
4660 represented.

4661 The identification of the TSFI is a necessary prerequisite to all other activities in this sub-activity.
4662 The TSF must be identified (done as part of the TOE design (ADV_TDS) work units) in order to
4663 identify the TSFI. This activity can be done at a high level to ensure that no large groups of
4664 interfaces have been missed (network protocols, hardware interfaces, configuration files), or at a
4665 low level as the evaluation of the functional specification proceeds.

4666 In making an assessment for this work unit, the evaluator determines that all portions of the TSF
4667 are addressed in terms of the interfaces listed in the functional specification. All portions of the TSF
4668 should have a corresponding interface description, or if there are no corresponding interfaces for a
4669 portion of the TSF, the evaluator determines that that is acceptable.

4670 ISO/IEC 15408-3 ADV_FSP.3.2C: *The functional specification shall describe the purpose and method
4671 of use for all TSFI.*

4672 **11.4.3.3.2 Work unit ADV_FSP.3-2**

4673 The evaluator **shall examine** the functional specification to determine that it states the purpose of
4674 each TSFI.

4675 The purpose of a TSFI is a general statement summarising the functionality provided by the
4676 interface. It is not intended to be a complete statement of the actions and results related to the
4677 interface, but rather a statement to help the reader understand in general what the interface is
4678 intended to be used for. The evaluator should not only determine that the purpose exists, but also
4679 that it accurately reflects the TSFI by taking into account other information about the interface,
4680 such as the description of actions and error messages.

4681 **11.4.3.3.3 Work unit ADV_FSP.3-3**

4682 The evaluator **shall examine** the functional specification to determine that the method of use for
4683 each TSFI is given.

4684 The method of use for a TSFI summarises how the interface is manipulated in order to invoke the
4685 actions and obtain the results associated with the TSFI. The evaluator should be able to determine,
4686 from reading this material in the functional specification, how to use each interface. This does not
4687 necessarily mean that there needs to be a separate method of use for each TSFI, as it may be
4688 possible to describe in general how kernel calls are invoked, for instance, and then identify each
4689 interface using that general style. Different types of interfaces will require different method of use
4690 specifications. APIs, network protocol interfaces, system configuration parameters, and hardware
4691 bus interfaces all have very different methods of use, and this should be taken into account by the
4692 developer when developing the functional specification, as well as by the evaluator evaluating the
4693 functional specification.

4694 For administrative interfaces whose functionality is documented as being inaccessible to untrusted
4695 users, the evaluator ensures that the method of making the functions inaccessible is described in
4696 the functional specification. It should be noted that this inaccessibility needs to be tested by the
4697 developer in their test suite.

4698 The evaluator should not only determine that the set of method of use descriptions exist, but also
4699 that they accurately cover each TSFI.

4700 ISO/IEC 15408-3 ADV_FSP.3.3C: *The functional specification shall identify and describe all*
4701 *parameters associated with each TSFI.*

4702 **11.4.3.3.4 Work unit ADV_FSP.3-4**

4703 The evaluator **shall examine** the presentation of the TSFI to determine that it completely identifies
4704 all parameters associated with every TSFI.

4705 The evaluator examines the functional specification to ensure that all of the parameters are
4706 described for each TSFI. Parameters are explicit inputs or outputs to an interface that control the
4707 behaviour of that interface. For examples, parameters are the arguments supplied to an API; the
4708 various fields in packet for a given network protocol; the individual key values in the Windows
4709 Registry; the signals across a set of pins on a chip; etc.

4710 In order to determine that all of the parameters are present in the TSFI, the evaluator should
4711 examine the rest of the interface description (actions, error messages, etc.) to determine if the
4712 effects of the parameter are accounted for in the description. The evaluator should also check other
4713 evidence provided for the evaluation (e.g., TOE design, security architecture description,
4714 operational user guidance, implementation representation) to see if behaviour or additional
4715 parameters are described there but not in the functional specification.

4716 **11.4.3.3.5 Work unit ADV_FSP.3-5**

4717 The evaluator **shall examine** the presentation of the TSFI to determine that it completely and
4718 accurately describes all parameters associated with every TSFI.

4719 Once all of the parameters have been identified, the evaluator needs to ensure that they are
4720 accurately described, and that the description of the parameters is complete. A parameter
4721 description tells what the parameter is in some meaningful way. For instance, the interface *foo(i)*
4722 could be described as having “parameter *i* which is an integer”; this is not an acceptable parameter
4723 description. A description such as “parameter *i* is an integer that indicates the number of users
4724 currently logged in to the system” is much more acceptable.

4725 In order to determine that the description of the parameters is complete, the evaluator should
4726 examine the rest of the interface description (purpose, method of use, actions, error messages, etc.)
4727 to determine if the descriptions of the parameter(s) are accounted for in the description. The
4728 evaluator should also check other evidence provided (e.g., TOE design, architectural design,
4729 operational user guidance, implementation representation) to see if behaviour or additional
4730 parameters are described there but not in the functional specification.

4731 ISO/IEC 15408-3 ADV_FSP.3.4C: *For each SFR-enforcing TSFI, the functional specification shall*
4732 *describe the SFR-enforcing actions associated with the TSFI.*

4733 **11.4.3.3.6 Work unit ADV_FSP.3-6**

4734 The evaluator **shall examine** the presentation of the TSFI to determine that it completely and
4735 accurately describes the SFR-enforcing actions associated with the SFR-enforcing TSFIs.

4736 If an action available through an interface plays a role in enforcing any security policy on the TOE
4737 (that is, if one of the actions of the interface can be traced to one of the SFRs levied on the TSF),

4738 then that interface is *SFR-enforcing*. Such policies are not limited to the access control policies, but
 4739 also refer to any functionality specified by one of the SFRs contained in the ST. Note that it is
 4740 possible that an interface may have various actions and results, some of which may be SFR-
 4741 enforcing and some of which may not.

4742 The developer is not required to “label” interfaces as SFR-enforcing, and likewise is not required to
 4743 identify actions available through an interface as SFR-enforcing. It is the evaluator's responsibility
 4744 to examine the evidence provided by the developer and determine that the required information is
 4745 present. In the case where the developer has identified the SFR-enforcing TSFI and SFR-enforcing
 4746 actions available through those TSFI, the evaluator must judge completeness and accuracy based
 4747 on other information supplied for the evaluation (e.g., TOE design, security architecture description,
 4748 operational user guidance), and on the other information presented for the interfaces (parameters
 4749 and parameter descriptions, error messages, etc.).

4750 In this case (developer has provided only the SFR-enforcing information for SFR-enforcing TSFI)
 4751 the evaluator also ensures that no interfaces have been mis-categorised. This is done by examining
 4752 other information supplied for the evaluation (e.g., TOE design, security architecture description,
 4753 operational user guidance), and the other information presented for the interfaces (parameters
 4754 and parameter descriptions, for example) not labelled as SFR-enforcing. The analysis done for
 4755 work units ADV_FSP.3-7 and ADV_FSP.3-8 are also used in making this determination.

4756 In the case where the developer has provided the same level of information on all interfaces, the
 4757 evaluator performs the same type of analysis mentioned in the previous paragraphs. The evaluator
 4758 should determine which interfaces are SFR-enforcing and which are not, and subsequently ensure
 4759 that the SFR-enforcing aspects of the SFR-enforcing actions are appropriately described. Note that
 4760 in this case, the evaluator should be able to perform the bulk of the work associated with work unit
 4761 ADV_FSP.3-8 in the course of performing this SFR-enforcing analysis.

4762 The SFR-enforcing actions are those that are visible at any external interface and that provide for
 4763 the enforcement of the SFRs being claimed. For example, if audit requirements are included in the
 4764 ST, then audit-related actions would be SFR-enforcing and therefore must be described, even if the
 4765 result of that action is generally not visible through the invoked interface (as is often the case with
 4766 audit, where a user action at one interface would produce an audit record visible at another
 4767 interface).

4768 The level of description that is required is that sufficient for the reader to understand what role the
 4769 TSFI actions play with respect to the SFR. The evaluator should keep in mind that the description
 4770 should be detailed enough to support the generation (and assessment) of test cases against that
 4771 interface. If the description is unclear or lacking detail such that meaningful testing cannot be
 4772 conducted against the TSFI, it is likely that the description is inadequate.

4773 ISO/IEC 15408-3 ADV_FSP.3.5C: *For each SFR-enforcing TSFI, the functional specification shall*
 4774 *describe direct error messages resulting from SFR-enforcing actions and exceptions associated with*
 4775 *invocation of the TSFI.*

4776 **11.4.3.3.7 Work unit ADV_FSP.3-7**

4777 The evaluator ***shall examine*** the presentation of the TSFI to determine that it completely and
 4778 accurately describes error messages that may result from an invocation of each SFR-enforcing TSFI.

4779 This work unit should be performed in conjunction with, or after, work unit ADV_FSP.3-6 in order
 4780 to ensure the set of SFR-enforcing TSFI is correctly identified. The evaluator should note that the
 4781 requirement and associated work unit is that all direct error messages associated with an SFR-
 4782 enforcing TSFI must be described, that are associated with SFR-enforcing actions. This is because
 4783 at this level of assurance, the “extra” information provided by the error message descriptions
 4784 should be used in determining whether all of the SFR-enforcing aspects of an interface have been
 4785 appropriately described. For instance, if an error message associated with a TSFI (e.g., “access
 4786 denied”) indicated that an SFR-enforcing decision or action had taken place, but in the description

4787 of the SFR-enforcing actions there was no mention of that particular SFR-enforcing mechanism,
4788 then the description may not be complete.

4789 Errors can take many forms, depending on the interface being described. For an API, the interface
4790 itself may return an error code, set a global error condition, or set a certain parameter with an
4791 error code. For a configuration file, an incorrectly configured parameter may cause an error
4792 message to be written to a log file. For a hardware PCI card, an error condition may raise a signal
4793 on the bus, or trigger an exception condition to the CPU.

4794 **11.4.3.4 Errors (and the associated error messages) come about through the invocation of**
4795 **an interface. The processing that occurs in response to the interface invocation may**
4796 **encounter error conditions, which trigger (through an implementation-specific**
4797 **mechanism) an error message to be generated. In some instances this may be a return**
4798 **value from the interface itself; in other instances a global value may be set and checked**
4799 **after the invocation of an interface. It is likely that a TOE will have a number of low-level**
4800 **error messages that may result from fundamental resource conditions, such as “disk full”**
4801 **or “resource locked”. While these error messages may map to a large number of TSFI, they**
4802 **could be used to detect instances where detail from an interface description has been**
4803 **omitted. For instance, a TSFI that produces a “disk full” message, but has no obvious**
4804 **description of why that TSFI should cause an access to the disk in its description of actions,**
4805 **might cause the evaluator to examine other evidence (Security Architecture (ADV_ARC),**
4806 **TOE design (ADV_TDS)) related that TSFI to determine if the description is accurate.**

4807 In order to determine that the description of the error messages of a TSFI is accurate and complete,
4808 the evaluator measures the interface description against the other evidence provided for the
4809 evaluation (e.g., TOE design, security architecture description, operational user guidance), as well
4810 as for other evidence supplied for that TSFI (description of SFR-enforcing actions, summary of SFR-
4811 supporting and SFR-non-interfering actions and results).

4812 ISO/IEC 15408-3 ADV_FSP.3.6C: *The functional specification shall summarise the SFR-supporting*
4813 *and SFR-non-interfering actions associated with each TSFI.*

4814 **11.4.3.4.1 Work unit ADV_FSP.3-8**

4815 The evaluator **shall examine** the presentation of the TSFI to determine that it summarises the SFR-
4816 supporting and SFR-non-interfering actions associated with each TSFI.

4817 The purpose of this work unit is to supplement the details about the SFR-enforcing actions
4818 (provided in work unit ADV_FSP.3-6) with a summary of the remaining actions (i.e., those that are
4819 not SFR-enforcing). This covers *all* SFR-supporting and SFR-non-interfering actions, whether
4820 invokable through SFR-enforcing TSFI or through SFR-supporting or SFR-non-interfering TSFI.
4821 Such a summary about all SFR-supporting and SFR-non-interfering actions helps to provide a more
4822 complete picture of the functions provided by the TSF, and is to be used by the evaluator in
4823 determining whether an action or TSFI may have been mis-categorised.

4824 The information to be provided is more abstract than that required for SFR-enforcing actions.
4825 While it should still be detailed enough so that the reader can understand what the action does, the
4826 description does not have to be detailed enough to support writing tests against it, for instance. For
4827 the evaluator, the key is that the information must be sufficient to make a positive determination
4828 that the action is SFR-supporting or SFR-non-interfering. If that level of information is missing, the
4829 summary is insufficient and more information must be obtained.

4830 ISO/IEC 15408-3 ADV_FSP.3.7C: *The tracing shall demonstrate that the SFRs trace to TSFIs in the*
4831 *functional specification.*

4832 **11.4.3.4.2 Work unit ADV_FSP.3-9**

4833 The evaluator **shall check** that the tracing links the SFRs to the corresponding TSFIs.

4834 The tracing is provided by the developer to serve as a guide to which SFRs are related to which
 4835 TSFIs. This tracing can be as simple as a table; it is used as input to the evaluator for use in the
 4836 following work units, in which the evaluator verifies its completeness and accuracy.

4837 **11.4.3.5 Action ADV_FSP.3.2E**

4838 **11.4.3.5.1 Work unit ADV_FSP.3-10**

4839 The evaluator *shall examine* the functional specification to determine that it is a complete
 4840 instantiation of the SFRs.

4841 To ensure that all SFRs are covered by the functional specification, as well as the test coverage
 4842 analysis, the evaluator may build upon the developer's tracing (see ADV_FSP.3-9 a map between
 4843 the TOE security functional requirements and the TSFI. Note that this map may have to be at a level
 4844 of detail below the component or even element level of the requirements, because of operations
 4845 (assignments, refinements, selections) performed on the functional requirement by the ST author.

4846 For example, the FDP_ACC.1 component contains an element with assignments. If the ST contained,
 4847 for instance, ten rules in the FDP_ACC.1 assignment, and these ten rules were covered by three
 4848 different TSFI, it would be inadequate for the evaluator to map FDP_ACC.1 to TSFI A, B, and C and
 4849 claim they had completed the work unit. Instead, the evaluator would map FDP_ACC.1 (rule 1) to
 4850 TSFI A; FDP_ACC.1 (rule 2) to TSFI B; etc. It might also be the case that the interface is a wrapper
 4851 interface (e.g., IOCTL), in which case the mapping would need to be specific to certain set of
 4852 parameters for a given interface.

4853 The evaluator must recognise that for requirements that have little or no manifestation at the TSF
 4854 boundary (e.g., FDP_RIP) it is not expected that they completely map those requirements to the
 4855 TSFI. The analysis for those requirements will be performed in the analysis for the TOE design
 4856 (ADV_TDS) when included in the ST. It is also important to note that since the parameters, actions,
 4857 and error messages associated with TSFIs must be fully specified, the evaluator should be able to
 4858 determine if all aspects of an SFR appear to be implemented at the interface level.

4859 **11.4.3.5.2 Work unit ADV_FSP.3-11**

4860 The evaluator *shall examine* the functional specification to determine that it is an accurate
 4861 instantiation of the SFRs.

4862 For each functional requirement in the ST that results in effects visible at the TSF boundary, the
 4863 information in the associated TSFI for that requirement specifies the required functionality
 4864 described by the requirement. For example, if the ST contains a requirement for access control lists,
 4865 and the only TSFI that map to that requirement specify functionality for Unix-style protection bits,
 4866 then the functional specification is not accurate with respect to the requirements.

4867 The evaluator must recognise that for requirements that have little or no manifestation at the TSF
 4868 boundary (e.g., FDP_RIP) it is not expected that the evaluator completely map those requirements
 4869 to the TSFI. The analysis for those requirements will be performed in the analysis for the TOE
 4870 design (ADV_TDS) when included in the ST.

4871 **11.4.4 Evaluation of sub-activity (ADV_FSP.4)**

4872 **11.4.4.1 Objectives**

4873 The objective of this sub-activity is to determine whether the developer has completely described
 4874 all of the TSFI in a manner such that the evaluator is able to determine whether the TSFI are
 4875 completely and accurately described, and appears to implement the security functional
 4876 requirements of the ST.

4877 **11.4.4.2 Input**

4878 The evaluation evidence for this sub-activity that is required by the work-units is:

- 4879 a) the ST;
- 4880 b) the functional specification;
- 4881 c) the TOE design.

4882 The evaluation evidence for this sub-activity that is used if included in the ST for the TOE is:

- 4883 a) the security architecture description;
- 4884 b) the implementation representation;
- 4885 c) the TSF internals description;
- 4886 d) the operational user guidance;

4887 **11.4.4.3 Application notes**

4888 The functional specification describes the interfaces to the TSF (the TSFI) in a structured manner.
4889 Because of the dependency on Evaluation of sub-activity (ADV_TDS.1), the evaluator is expected to
4890 have identified the TSF prior to beginning work on this sub-activity. Without firm knowledge of
4891 what comprises the TSF, it is not possible to assess the completeness of the TSFI.

4892 In performing the various work units included in this family, the evaluator is asked to make
4893 assessments of accuracy and completeness of several factors (the TSFI itself, as well as the
4894 individual components (parameters, actions, error messages, etc.) of the TSFI). In doing this
4895 analysis, the evaluator is expected to use the documentation provided for the evaluation. This
4896 includes the ST, the TOE design, and may include other documentation such as the operational user
4897 guidance, security architecture description, and implementation representation. The
4898 documentation should be examined in an iterative fashion. The evaluator may read, for example, in
4899 the TOE design how a certain function is implemented, but see no way to invoke that function from
4900 the interface. This might cause the evaluator to question the completeness of a particular TSFI
4901 description, or whether an interface has been left out of the functional specification altogether.
4902 Describing analysis activities of this sort in the ETR is a key method in providing rationale that the
4903 work units have been performed appropriately.

4904 It should be recognised that there exist functional requirements whose functionality is manifested
4905 wholly or in part architecturally, rather than through a specific mechanism. An example of this is
4906 the implementation of mechanisms implementing the **Residual information protection (FDP_RIP)**
4907 requirements. Such mechanisms typically are implemented to ensure a behaviour isn't present,
4908 which is difficult to test and typically is verified through analysis. In the cases where such
4909 functional requirements are included in the ST, it is expected that the evaluator recognise that
4910 there may be SFRs of this type that have no interfaces, and that this should not be considered a
4911 deficiency in the functional specification.

4912 **11.4.4.4 Action ADV_FSP.4.1E**

4913 ISO/IEC 15408-3 ADV_FSP.4.1C: *The functional specification shall completely represent the TSF.*

4914 **11.4.4.4.1 Work unit ADV_FSP.4-1**

4915 The evaluator **shall examine** the functional specification to determine that the TSF is fully
4916 represented.

4917 The identification of the TSFI is a necessary prerequisite to all other activities in this sub-activity.
 4918 The TSF must be identified (done as part of the TOE design (ADV_TDS) work units) in order to
 4919 identify the TSFI. This activity can be done at a high level to ensure that no large groups of
 4920 interfaces have been missed (network protocols, hardware interfaces, configuration files), or at a
 4921 low level as the evaluation of the functional specification proceeds.

4922 In making an assessment for this work unit, the evaluator determines that all portions of the TSF
 4923 are addressed in terms of the interfaces listed in the functional specification. All portions of the TSF
 4924 should have a corresponding interface description, or if there are no corresponding interfaces for a
 4925 portion of the TSF, the evaluator determines that that is acceptable.

4926 ISO/IEC 15408-3 ADV_FSP.4.2C: *The functional specification shall describe the purpose and method*
 4927 *of use for all TSFI.*

4928 **11.4.4.4.2 Work unit ADV_FSP.4-2**

4929 The evaluator *shall examine* the functional specification to determine that it states the purpose of
 4930 each TSFI.

4931 The purpose of a TSFI is a general statement summarising the functionality provided by the
 4932 interface. It is not intended to be a complete statement of the actions and results related to the
 4933 interface, but rather a statement to help the reader understand in general what the interface is
 4934 intended to be used for. The evaluator should not only determine that the purpose exists, but also
 4935 that it accurately reflects the TSFI by taking into account other information about the interface,
 4936 such as the description of actions and error messages.

4937 **11.4.4.4.3 Work unit ADV_FSP.4-3**

4938 The evaluator *shall examine* the functional specification to determine that the method of use for
 4939 each TSFI is given.

4940 The method of use for a TSFI summarises how the interface is manipulated in order to invoke the
 4941 actions and obtain the results associated with the TSFI. The evaluator should be able to determine,
 4942 from reading this material in the functional specification, how to use each interface. This does not
 4943 necessarily mean that there needs to be a separate method of use for each TSFI, as it may be
 4944 possible to describe in general how kernel calls are invoked, for instance, and then identify each
 4945 interface using that general style. Different types of interfaces will require different method of use
 4946 specifications. APIs, network protocol interfaces, system configuration parameters, and hardware
 4947 bus interfaces all have very different methods of use, and this should be taken into account by the
 4948 developer when developing the functional specification, as well as by the evaluator evaluating the
 4949 functional specification.

4950 For administrative interfaces whose functionality is documented as being inaccessible to untrusted
 4951 users, the evaluator ensures that the method of making the functions inaccessible is described in
 4952 the functional specification. It should be noted that this inaccessibility needs to be tested by the
 4953 developer in their test suite.

4954 The evaluator should not only determine that the set of method of use descriptions exist, but also
 4955 that they accurately cover each TSFI.

4956 **11.4.4.4.4 Work unit ADV_FSP.4-4**

4957 The evaluator *shall examine* the functional specification to determine the completeness of the
 4958 TSFI

4959 The evaluator shall use the design documentation to identify the possible types of interfaces. The
 4960 evaluator shall search the design documentation and the guidance documentation for potential
 4961 TSFI not contained in the developer's documentation, thus indicating that the set of TSFI defined

4962 by the developer is incomplete. The evaluator **shall examine** the arguments presented by the
4963 developer that the TSFI is complete and check down to the lowest level of design or with the
4964 implementation representation that no additional TSFI exist.

4965 ISO/IEC 15408-3 ADV_FSP.4.3C: *The functional specification shall identify and describe all*
4966 *parameters associated with each TSFI.*

4967 **11.4.4.4.5 Work unit ADV_FSP.4-5**

4968 The evaluator **shall examine** the presentation of the TSFI to determine that it completely identifies
4969 all parameters associated with every TSFI.

4970 The evaluator examines the functional specification to ensure that all of the parameters are
4971 described for each TSFI. Parameters are explicit inputs or outputs to an interface that control the
4972 behaviour of that interface. For examples, parameters are the arguments supplied to an API; the
4973 various fields in packet for a given network protocol; the individual key values in the Windows
4974 Registry; the signals across a set of pins on a chip; etc.

4975 In order to determine that all of the parameters are present in the TSFI, the evaluator should
4976 examine the rest of the interface description (actions, error messages, etc.) to determine if the
4977 effects of the parameter are accounted for in the description. The evaluator should also check other
4978 evidence provided for the evaluation (e.g., TOE design, security architecture description,
4979 operational user guidance, implementation representation) to see if behaviour or additional
4980 parameters are described there but not in the functional specification.

4981 **11.4.4.4.6 Work unit ADV_FSP.4-6**

4982 The evaluator **shall examine** the presentation of the TSFI to determine that it completely and
4983 accurately describes all parameters associated with every TSFI.

4984 Once all of the parameters have been identified, the evaluator needs to ensure that they are
4985 accurately described, and that the description of the parameters is complete. A parameter
4986 description tells what the parameter is in some meaningful way. For instance, the interface *foo(i)*
4987 could be described as having “parameter i which is an integer”; this is not an acceptable parameter
4988 description. A description such as “parameter i is an integer that indicates the number of users
4989 currently logged in to the system” is much more acceptable.

4990 In order to determine that the description of the parameters is complete, the evaluator should
4991 examine the rest of the interface description (purpose, method of use, actions, error messages, etc.)
4992 to determine if the descriptions of the parameter(s) are accounted for in the description. The
4993 evaluator should also check other evidence provided (e.g., TOE design, architectural design,
4994 operational user guidance, implementation representation) to see if behaviour or additional
4995 parameters are described there but not in the functional specification.

4996 ISO/IEC 15408-3 ADV_FSP.4.4C: *The functional specification shall describe all actions associated*
4997 *with each TSFI.*

4998 **11.4.4.4.7 Work unit ADV_FSP.4-7**

4999 The evaluator **shall examine** the presentation of the TSFI to determine that it completely and
5000 accurately describes all actions associated with every TSFI.

5001 The evaluator checks to ensure that all of the actions are described. actions available through an
5002 interface describe what the interface does (as opposed to the TOE design, which describes how the
5003 actions are provided by the TSF).

5004 Actions of an interface describe functionality that can be invoked through the interface, and can be
5005 categorised as *regular* actions, and *SFR-related* actions. Regular actions are descriptions of what the

5006 interface does. The amount of information provided for this description is dependant on the
 5007 complexity of the interface. The SFR-related actions are those that are visible at any external
 5008 interface (for instance, audit activity caused by the invocation of an interface (assuming audit
 5009 requirements are included in the ST) should be described, even though the result of that action is
 5010 generally not visible through the invoked interface). Depending on the parameters of an interface,
 5011 there may be many different actions able to be invoked through the interface (for instance, an API
 5012 might have the first parameter be a “subcommand”, and the following parameters be specific to
 5013 that subcommand. The IOCTL API in some Unix systems is an example of such an interface).

5014 In order to determine that the description of the actions of a TSFI is complete, the evaluator should
 5015 review the rest of the interface description (parameter descriptions, error messages, etc.) to
 5016 determine if the actions described are accounted for. The evaluator should also analyse other
 5017 evidence provided for the evaluation (e.g., TOE design, security architecture description,
 5018 operational user guidance, implementation representation) to see if there is evidence of actions
 5019 that are described there but not in the functional specification.

5020 ISO/IEC 15408-3 ADV_FSP.4.5C: *The functional specification shall describe all direct error messages*
 5021 *that may result from an invocation of each TSFI.*

5022 **11.4.4.4.8 Work unit ADV_FSP.4-8**

5023 The evaluator ***shall examine*** the presentation of the TSFI to determine that it completely and
 5024 accurately describes all error messages resulting from an invocation of each TSFI.

5025 Errors can take many forms, depending on the interface being described. For an API, the interface
 5026 itself may return an error code; set a global error condition, or set a certain parameter with an
 5027 error code. For a configuration file, an incorrectly configured parameter may cause an error
 5028 message to be written to a log file. For a hardware PCI card, an error condition may raise a signal
 5029 on the bus, or trigger an exception condition to the CPU.

5030 Errors (and the associated error messages) come about through the invocation of an interface. The
 5031 processing that occurs in response to the interface invocation may encounter error conditions,
 5032 which trigger (through an implementation-specific mechanism) an error message to be generated.
 5033 In some instances this may be a return value from the interface itself; in other instances a global
 5034 value may be set and checked after the invocation of an interface. It is likely that a TOE will have a
 5035 number of low-level error messages that may result from fundamental resource conditions, such as
 5036 “disk full” or “resource locked”. While these error messages may map to a large number of TSFI,
 5037 they could be used to detect instances where detail from an interface description has been omitted.
 5038 For instance, a TSFI that produces a “disk full” message, but has no obvious description of why that
 5039 TSFI should cause an access to the disk in its description of actions, might cause the evaluator to
 5040 examine other evidence (Security Architecture (ADV_ARC), TOE design (TOE_TDS)) related that
 5041 TSFI to determine if the description is complete and accurate.

5042 The evaluator determines that, for each TSFI, the exact set of error messages that can be returned
 5043 on invoking that interface can be determined. The evaluator reviews the evidence provided for the
 5044 interface to determine if the set of errors seems complete. They cross-check this information with
 5045 other evidence provided for the evaluation (e.g., TOE design, security architecture description,
 5046 operational user guidance, implementation representation) to ensure that there are no errors
 5047 steaming from processing mentioned that are not included in the functional specification.

5048 **11.4.4.4.9 Work unit ADV_FSP.4-9**

5049 The evaluator ***shall examine*** the presentation of the TSFI to determine that it completely and
 5050 accurately describes the meaning of all error messages resulting from an invocation of each TSFI.

5051 In order to determine accuracy, the evaluator must be able to understand meaning of the error. For
 5052 example, if an interface returns a numeric code of 0, 1, or 2, the evaluator would not be able to
 5053 understand the error if the functional specification only listed: “possible errors resulting from

5054 invocation of the *foo()* interface are 0, 1, or 2". Instead the evaluator checks to ensure that the
5055 errors are described such as: "possible errors resulting from invocation of the *foo()* interface are 0
5056 (processing successful), 1 (file not found), or 2 (incorrect filename specification)".

5057 In order to determine that the description of the errors due to invoking a TSFI is complete, the
5058 evaluator examines the rest of the interface description (parameter descriptions, actions, etc.) to
5059 determine if potential error conditions that might be caused by using such an interface are
5060 accounted for. The evaluator also checks other evidence provided for the evaluation (e.g. TOE
5061 design, security architecture description, operational user guidance, implementation
5062 representation) to see if error processing related to the TSFI is described there but is not described
5063 in the functional specification.

5064 ISO/IEC 15408-3 ADV_FSP.4.6C: *The tracing shall demonstrate that the SFRs trace to TSFIs in the*
5065 *functional specification.*

5066 **11.4.4.4.10 Work unit ADV_FSP.4-10**

5067 The evaluator **shall check** that the tracing links the SFRs to the corresponding TSFIs.

5068 The tracing is provided by the developer to serve as a guide to which SFRs are related to which
5069 TSFIs. This tracing can be as simple as a table; it is used as input to the evaluator for use in the
5070 following work units, in which the evaluator verifies its completeness and accuracy.

5071 **11.4.4.5 Action ADV_FSP.4.2E**

5072 **11.4.4.5.1 Work unit ADV_FSP.4-11**

5073 The evaluator **shall examine** the functional specification to determine that it is a complete
5074 instantiation of the SFRs.

5075 To ensure that all SFRs are covered by the functional specification, as well as the test coverage
5076 analysis, the evaluator may build upon the developer's tracing (see ADV_FSP.4-10 a map between
5077 the TOE security functional requirements and the TSFI. Note that this map may have to be at a level
5078 of detail below the component or even element level of the requirements, because of operations
5079 (assignments, refinements, selections) performed on the functional requirement by the ST author.

5080 For example, the FDP_ACC.1 component contains an element with assignments. If the ST contained,
5081 for instance, ten rules in the FDP_ACC.1 assignment, and these ten rules were covered by three
5082 different TSFI, it would be inadequate for the evaluator to map FDP_ACC.1 to TSFI A, B, and C and
5083 claim they had completed the work unit. Instead, the evaluator would map FDP_ACC.1 (rule 1) to
5084 TSFI A; FDP_ACC.1 (rule 2) to TSFI B; etc. It might also be the case that the interface is a wrapper
5085 interface (e.g., IOCTL), in which case the mapping would need to be specific to certain set of
5086 parameters for a given interface.

5087 The evaluator must recognise that for requirements that have little or no manifestation at the TSF
5088 boundary (e.g., FDP_RIP) it is not expected that they completely map those requirements to the
5089 TSFI. The analysis for those requirements will be performed in the analysis for the TOE design
5090 (ADV_TDS) when included in the ST. It is also important to note that since the parameters, actions,
5091 and error messages associated with TSFIs must be fully specified, the evaluator should be able to
5092 determine if all aspects of an SFR appear to be implemented at the interface level.

5093 **11.4.4.5.2 Work unit ADV_FSP.4-12**

5094 The evaluator **shall examine** the functional specification to determine that it is an accurate
5095 instantiation of the SFRs.

5096 For each functional requirement in the ST that results in effects visible at the TSF boundary, the
5097 information in the associated TSFI for that requirement specifies the required functionality

5098 described by the requirement. For example, if the ST contains a requirement for access control lists,
5099 and the only TSFI that map to that requirement specify functionality for Unix-style protection bits,
5100 then the functional specification is not accurate with respect to the requirements.

5101 The evaluator must recognise that for requirements that have little or no manifestation at the TSF
5102 boundary (e.g., FDP_RIP) it is not expected that the evaluator completely map those requirements
5103 to the TSFI. The analysis for those requirements will be performed in the analysis for the TOE
5104 design (ADV_TDS) when included in the ST.

5105 **11.4.5 Evaluation of sub-activity (ADV_FSP.5)**

5106 **11.4.5.1 Objectives**

5107 The objective of this sub-activity is to determine whether the developer has completely described
5108 all of the TSFI in a manner such that the evaluator is able to determine whether the TSFI are
5109 completely and accurately described, and appears to implement the security functional
5110 requirements of the ST. The completeness of the interfaces is judged based upon the
5111 implementation representation.

5112 **11.4.5.2 Input**

5113 The evaluation evidence for this sub-activity that is required by the work-units is:

- 5114 a) the ST;
- 5115 b) the functional specification;
- 5116 c) the TOE design;
- 5117 d) the implementation representation.

5118 The evaluation evidence for this sub-activity that is used if included in the ST for the TOE is:

- 5119 a) the security architecture description;
- 5120 b) the TSF internals description;
- 5121 c) the formal security policy model;
- 5122 d) the operational user guidance;

5123 **11.4.5.3 Action ADV_FSP.5.1E**

5124 ISO/IEC 15408-3 ADV_FSP.5.1C: *The functional specification shall completely represent the TSF.*

5125 **11.4.5.3.1 Work unit ADV_FSP.5-1**

5126 The evaluator ***shall examine*** the functional specification to determine that the TSF is fully
5127 represented.

5128 The identification of the TSFI is a necessary prerequisite to all other activities in this sub-activity.
5129 The TSF must be identified (done as part of the TOE design (TOE_TDS) work units) in order to
5130 identify the TSFI. This activity can be done at a high level to ensure that no large groups of
5131 interfaces have been missed (network protocols, hardware interfaces, configuration files), or at a
5132 low level as the evaluation of the functional specification proceeds.

5133 In making an assessment for this work unit, the evaluator determines that all portions of the TSF
5134 are addressed in terms of the interfaces listed in the functional specification. All portions of the TSF

5135 should have a corresponding interface description, or if there are no corresponding interfaces for a
5136 portion of the TSF, the evaluator determines that that is acceptable.

5137 ISO/IEC 15408-3 ADV_FSP.5.2C: *The functional specification shall describe the TSFI using a semi-*
5138 *formal style.*

5139 **11.4.5.3.2 Work unit ADV_FSP.5-2**

5140 The evaluator **shall examine** the functional specification to determine that it is presented using a
5141 semiformal style.

5142 A semi-formal presentation is characterised by a standardised format with a well-defined syntax
5143 that reduces ambiguity that may occur in informal presentations. Since the intent of the semi-
5144 formal format is to enhance the reader's ability to understand the presentation, use of certain
5145 structured presentation methods (pseudo-code, flow charts, block diagrams) are appropriate,
5146 though not required.

5147 For the purposes of this activity, the evaluator should ensure that the interface descriptions are
5148 formatted in a structured, consistent manner and use common terminology. A semiformal
5149 presentation of the interfaces also implies that the level of detail of the presentation for the
5150 interfaces is largely consistent across all TSFI. For the functional specification, it is acceptable to
5151 refer to external specifications for portions of the interface as long as those external specifications
5152 are themselves semiformal.

5153 ISO/IEC 15408-3 ADV_FSP.5.3C: *The functional specification shall describe the purpose and method*
5154 *of use for all TSFI.*

5155 **11.4.5.3.3 Work unit ADV_FSP.5-3**

5156 The evaluator **shall examine** the functional specification to determine that it states the purpose of
5157 each TSFI.

5158 The purpose of a TSFI is a general statement summarising the functionality provided by the
5159 interface. It is not intended to be a complete statement of the actions and results related to the
5160 interface, but rather a statement to help the reader understand in general what the interface is
5161 intended to be used for. The evaluator should not only determine that the purpose exists, but also
5162 that it accurately reflects the TSFI by taking into account other information about the interface,
5163 such as the description of actions and error messages.

5164 Work unit ADV_FSP.5-4

5165 The evaluator **shall examine** the functional specification to determine that the method of use for
5166 each TSFI is given.

5167 The method of use for a TSFI summarises how the interface is manipulated in order to invoke the
5168 actions and obtain the results associated with the TSFI. The evaluator should be able to determine,
5169 from reading this material in the functional specification, how to use each interface. This does not
5170 necessarily mean that there needs to be a separate method of use for each TSFI, as it may be
5171 possible to describe in general how kernel calls are invoked, for instance, and then identify each
5172 interface using that general style. Different types of interfaces will require different method of use
5173 specifications. APIs, network protocol interfaces, system configuration parameters, and hardware
5174 bus interfaces all have very different methods of use, and this should be taken into account by the
5175 developer when developing the functional specification, as well as by the evaluator evaluating the
5176 functional specification.

5177 For administrative interfaces whose functionality is documented as being inaccessible to untrusted
5178 users, the evaluator ensures that the method of making the functions inaccessible is described in

5179 the functional specification. It should be noted that this inaccessibility needs to be tested by the
5180 developer in their test suite.

5181 The evaluator should not only determine that the set of method of use descriptions exist, but also
5182 that they accurately cover each TSFI.

5183 **11.4.5.3.4 Work unit ADV_FSP.5-5**

5184 The evaluator *shall examine* the functional specification to determine the completeness of the
5185 TSFI

5186 The evaluator shall use the design documentation to identify the possible types of interfaces. The
5187 evaluator shall search the design documentation and the guidance documentation for potential
5188 TSFI not contained in the developer's documentation, thus indicating that the set of TSFI defined
5189 by the developer is incomplete. The evaluator *shall examine* the arguments presented by the
5190 developer that the TSFI is complete and check down to the lowest level of design or with the
5191 implementation representation that no additional TSFI exist.

5192 ISO/IEC 15408-3 ADV_FSP.5.4C: *The functional specification shall identify and describe all*
5193 *parameters associated with each TSFI.*

5194 **11.4.5.3.5 Work unit ADV_FSP.5-6**

5195 The evaluator *shall examine* the presentation of the TSFI to determine that it completely identifies
5196 all parameters associated with every TSFI.

5197 The evaluator examines the functional specification to ensure that all of the parameters are
5198 described for each TSFI. Parameters are explicit inputs or outputs to an interface that control the
5199 behaviour of that interface. For examples, parameters are the arguments supplied to an API; the
5200 various fields in packet for a given network protocol; the individual key values in the Windows
5201 Registry; the signals across a set of pins on a chip; etc.

5202 In order to determine that all of the parameters are present in the TSFI, the evaluator should
5203 examine the rest of the interface description (actions, error messages, etc.) to determine if the
5204 effects of the parameter are accounted for in the description. The evaluator should also check other
5205 evidence provided for the evaluation (e.g., TOE design, security architecture description,
5206 operational user guidance, implementation representation) to see if behaviour or additional
5207 parameters are described there but not in the functional specification.

5208 **11.4.5.3.6 Work unit ADV_FSP.5-7**

5209 The evaluator *shall examine* the presentation of the TSFI to determine that it completely and
5210 accurately describes all parameters associated with every TSFI.

5211 Once all of the parameters have been identified, the evaluator needs to ensure that they are
5212 accurately described, and that the description of the parameters is complete. A parameter
5213 description tells what the parameter is in some meaningful way. For instance, the interface *foo(i)*
5214 could be described as having "parameter i which is an integer"; this is not an acceptable parameter
5215 description. A description such as "parameter i is an integer that indicates the number of users
5216 currently logged in to the system". is much more acceptable.

5217 In order to determine that the description of the parameters is complete, the evaluator should
5218 examine the rest of the interface description (purpose, method of use, actions, error messages, etc.)
5219 to determine if the descriptions of the parameter(s) are accounted for in the description. The
5220 evaluator should also check other evidence provided (e.g., TOE design, architectural design,
5221 operational user guidance, implementation representation) to see if behaviour or additional
5222 parameters are described there but not in the functional specification.

5223 ISO/IEC 15408-3 ADV_FSP.5.5C: *The functional specification shall describe all actions associated*
5224 *with each TSFI.*

5225 **11.4.5.3.7 Work unit ADV_FSP.5-8**

5226 The evaluator **shall examine** the presentation of the TSFI to determine that it completely and
5227 accurately describes all actions associated with every TSFI.

5228 The evaluator checks to ensure that all of the actions are described. actions available through an
5229 interface describe what the interface does (as opposed to the TOE design, which describes how the
5230 actions are provided by the TSF).

5231 actions of an interface describe functionality that can be invoked through the interface, and can be
5232 categorised as *regular* actions, and *SFR-related* actions. Regular actions are descriptions of what the
5233 interface does. The amount of information provided for this description is dependant on the
5234 complexity of the interface. The SFR-related actions are those that are visible at any external
5235 interface (for instance, audit activity caused by the invocation of an interface (assuming audit
5236 requirements are included in the ST) should be described, even though the result of that action is
5237 generally not visible through the invoked interface). Depending on the parameters of an interface,
5238 there may be many different actions able to be invoked through the interface (for instance, an API
5239 might have the first parameter be a “subcommand”, and the following parameters be specific to
5240 that subcommand. The IOCTL API in some Unix systems is an example of such an interface).

5241 In order to determine that the description of the actions of a TSFI is complete, the evaluator should
5242 review the rest of the interface description (parameter descriptions, error messages, etc.) to
5243 determine if the actions described are accounted for. The evaluator should also analyse other
5244 evidence provided for the evaluation (e.g., TOE design, security architecture description,
5245 operational user guidance, implementation representation) to see if there is evidence of actions
5246 that are described there but not in the functional specification.

5247 ISO/IEC 15408-3 ADV_FSP.5.6C: *The functional specification shall describe all direct error messages*
5248 *that may result from an invocation of each TSFI.*

5249 **11.4.5.3.8 Work unit ADV_FSP.5-9**

5250 The evaluator **shall examine** the presentation of the TSFI to determine that it completely and
5251 accurately describes all error messages resulting from an invocation of each TSFI.

5252 Errors can take many forms, depending on the interface being described. For an API, the interface
5253 itself may return an error code; set a global error condition, or set a certain parameter with an
5254 error code. For a configuration file, an incorrectly configured parameter may cause an error
5255 message to be written to a log file. For a hardware PCI card, an error condition may raise a signal
5256 on the bus, or trigger an exception condition to the CPU.

5257 Errors (and the associated error messages) come about through the invocation of an interface. The
5258 processing that occurs in response to the interface invocation may encounter error conditions,
5259 which trigger (through an implementation-specific mechanism) an error message to be generated.
5260 In some instances this may be a return value from the interface itself; in other instances a global
5261 value may be set and checked after the invocation of an interface. It is likely that a TOE will have a
5262 number of low-level error messages that may result from fundamental resource conditions, such as
5263 “disk full” or “resource locked”. While these error messages may map to a large number of TSFI,
5264 they could be used to detect instances where detail from an interface description has been omitted.
5265 For instance, a TSFI that produces a “disk full” message, but has no obvious description of why that
5266 TSFI should cause an access to the disk in its description of actions, might cause the evaluator to
5267 examine other evidence (ADV_ARC, ADV_TDS) related that TSFI to determine if the description is
5268 complete and accurate.

5269 The evaluator determines that, for each TSFI, the exact set of error messages that can be returned
 5270 on invoking that interface can be determined. The evaluator reviews the evidence provided for the
 5271 interface to determine if the set of errors seems complete. They cross-check this information with
 5272 other evidence provided for the evaluation (e.g., TOE design, security architecture description,
 5273 operational user guidance, implementation representation) to ensure that there are no errors
 5274 steaming from processing mentioned that are not included in the functional specification.

5275 **11.4.5.3.9 Work unit ADV_FSP.5-10**

5276 The evaluator *shall examine* the presentation of the TSFI to determine that it completely and
 5277 accurately describes the meaning of all error messages resulting from an invocation of each TSFI.

5278 In order to determine accuracy, the evaluator must be able to understand meaning of the error. For
 5279 example, if an interface returns a numeric code of 0, 1, or 2, the evaluator would not be able to
 5280 understand the error if the functional specification only listed: “possible errors resulting from
 5281 invocation of the *foo()* interface are 0, 1, or 2”. Instead the evaluator checks to ensure that the
 5282 errors are described such as: “possible errors resulting from invocation of the *foo()* interface are 0
 5283 (processing successful), 1 (file not found), or 2 (incorrect filename specification)”.

5284 In order to determine that the description of the errors due to invoking a TSFI is complete, the
 5285 evaluator examines the rest of the interface description (parameter descriptions, actions, etc.) to
 5286 determine if potential error conditions that might be caused by using such an interface are
 5287 accounted for. The evaluator also checks other evidence provided for the evaluation (e.g., TOE
 5288 design, security architecture description, operational user guidance, implementation
 5289 representation) to see if error processing related to the TSFI is described there but is not described
 5290 in the functional specification.

5291 ISO/IEC 15408-3 ADV_FSP.5.7C: *The functional specification shall describe all error messages that*
 5292 *do not result from an invocation of a TSFI.*

5293 **11.4.5.3.10 Work unit ADV_FSP.5-11**

5294 The evaluator *shall examine* the functional specification to determine that it completely and
 5295 accurately describes all error messages that do not result from an invocation of any TSFI.

5296 This work unit complements work unit ADV_FSP.5-9, which describes those error messages that
 5297 result from an invocation of the TSFI. Taken together, these work units cover all error messages
 5298 that might be generated by the TSF.

5299 The evaluator assesses the completeness and accuracy of the functional specification by comparing
 5300 its contents to instances of error message generation within the implementation representation.
 5301 Most of these error messages will have already been covered by work unit ADV_FSP.5-9.

5302 The error messages related to this work unit are typically those that are not expected to be
 5303 generated, but are constructed as a matter of good programming practises. For example, a case
 5304 statement that defines actions resulting from each of a list of cases may end with a final *else*
 5305 statement to apply to anything that might not be expected; this practise ensures the TSF does not
 5306 get into an undefined state. However, it is not expected that the path of execution would ever get to
 5307 this *else* statement; therefore, any error message generation within this *else* statement would never
 5308 be generated. Although it would not get generated, it must still be included in the functional
 5309 specification.

5310 ISO/IEC 15408-3 ADV_FSP.5.8C: *The functional specification shall provide a rationale for each error*
 5311 *message contained in the TSF implementation yet does not result from an invocation of a TSFI.*

5312 **11.4.5.3.11 Work unit ADV_FSP.5-12**

5313 The evaluator *shall examine* the functional specification to determine that it provides a rationale
5314 for each error message contained in the TSF implementation yet does not result from an invocation
5315 of a TSFI.

5316 The evaluator ensures that every error message found under work unit ADV_FSP.5-11 contains a
5317 rationale describing why it cannot be invoked from the TSFI.

5318 As was described in the previous work unit, this rationale might be as straightforward as the fact
5319 that the error message in question is provided for completeness of execution logic and that it is
5320 never expected to be generated. The evaluator ensures that the rationale for each such error
5321 message is logical.

5322 ISO/IEC 15408-3 ADV_FSP.5.9C: *The tracing shall demonstrate that the SFRs trace to TSFIs in the*
5323 *functional specification.*

5324 **11.4.5.3.12 Work unit ADV_FSP.5-13**

5325 The evaluator *shall check* that the tracing links the SFRs to the corresponding TSFIs.

5326 The tracing is provided by the developer to serve as a guide to which SFRs are related to which
5327 TSFIs. This tracing can be as simple as a table; it is used as input to the evaluator for use in the
5328 following work units, in which the evaluator verifies its completeness and accuracy.

5329 **11.4.5.4 Action ADV_FSP.5.2E**

5330 **11.4.5.4.1 Work unit ADV_FSP.5-14**

5331 The evaluator *shall examine* the functional specification to determine that it is a complete
5332 instantiation of the SFRs.

5333 To ensure that all SFRs are covered by the functional specification, as well as the test coverage
5334 analysis, the evaluator may build upon the developer's tracing (see ADV_FSP.5-13 a map between
5335 the TOE security functional requirements and the TSFI. Note that this map may have to be at a level
5336 of detail below the component or even element level of the requirements, because of operations
5337 (assignments, refinements, selections) performed on the functional requirement by the ST author.

5338 For example, the FDP_ACC.1 component contains an element with assignments. If the ST contained,
5339 for instance, ten rules in the FDP_ACC.1 assignment, and these ten rules were covered by three
5340 different TSFI, it would be inadequate for the evaluator to map FDP_ACC.1 to TSFI A, B, and C and
5341 claim they had completed the work unit. Instead, the evaluator would map FDP_ACC.1 (rule 1) to
5342 TSFI A; FDP_ACC.1 (rule 2) to TSFI B; etc. It might also be the case that the interface is a wrapper
5343 interface (e.g., IOCTL), in which case the mapping would need to be specific to certain set of
5344 parameters for a given interface.

5345 The evaluator must recognise that for requirements that have little or no manifestation at the TSF
5346 boundary (e.g., FDP_RIP) it is not expected that they completely map those requirements to the
5347 TSFI. The analysis for those requirements will be performed in the analysis for the TOE design
5348 (ADV_TDS) when included in the ST. It is also important to note that since the parameters, actions,
5349 and error messages associated with TSFIs must be fully specified, the evaluator should be able to
5350 determine if all aspects of an SFR appear to be implemented at the interface level.

5351 **11.4.5.4.2 Work unit ADV_FSP.5-15**

5352 The evaluator *shall examine* the functional specification to determine that it is an accurate
5353 instantiation of the SFRs.

5354 For each functional requirement in the ST that results in effects visible at the TSF boundary, the
 5355 information in the associated TSFI for that requirement specifies the required functionality
 5356 described by the requirement. For example, if the ST contains a requirement for access control lists,
 5357 and the only TSFI that map to that requirement specify functionality for Unix-style protection bits,
 5358 then the functional specification is not accurate with respect to the requirements.

5359 The evaluator must recognise that for requirements that have little or no manifestation at the TSF
 5360 boundary (e.g., FDP_RIP) it is not expected that the evaluator completely map those requirements
 5361 to the TSFI. The analysis for those requirements will be performed in the analysis for the TOE
 5362 design (ADV_TDS) when included in the ST.

5363 **11.4.6 Evaluation of sub-activity (ADV_FSP.6)**

5364 There is no general guidance; the scheme should be consulted for guidance on this sub-activity.

5365 **11.5 Implementation representation (ADV_IMP)**

5366 **11.5.1 Evaluation of sub-activity (ADV_IMP.1)**

5367 **11.5.1.1 Objectives**

5368 The objective of this sub-activity is to determine that the implementation representation made
 5369 available by the developer is suitable for use in other analysis activities; *suitability* is judged by its
 5370 conformance to the requirements for this component.

5371 **11.5.1.2 Input**

5372 The evaluation evidence for this sub-activity is:

- 5373 a) the implementation representation;
- 5374 b) the documentation of the development tools, as resulting from ALC_TAT ;
- 5375 c) TOE design description.

5376 **11.5.1.3 Application notes**

5377 The entire implementation representation is made available to ensure that analysis activities are
 5378 not curtailed due to lack of information. This does not, however, imply that all of the representation
 5379 is examined when the analysis activities are being performed. This is likely impractical in almost all
 5380 cases, in addition to the fact that it most likely will not result in a higher-assurance TOE vs. targeted
 5381 sampling of the implementation representation. For this sub-activity, this is even truer. It would
 5382 not be productive for the evaluator to spend large amounts of time verifying the requirements for
 5383 one portion of the implementation representation, and then use a different portion of the
 5384 implementation representation in performing analysis for other work units. Therefore, the
 5385 evaluator is encouraged to select the sample of the implementation representation from the areas
 5386 of the TOE that will be of most interest during the analysis performed during work units from other
 5387 families (e.g. ATE_IND, AVA_VAN and ADV_INT).

5388 **11.5.1.4 Action ADV_IMP.1.1E**

5389 ISO/IEC 15408-3 ADV_IMP.1.1C: *The implementation representation shall define the TSF to a level of*
 5390 *detail such that the TSF can be generated without further design decisions.*

5391 **11.5.1.4.1 Work unit ADV_IMP.1-1**

5392 The evaluator **shall check** that the implementation representation defines the TSF to a level of
 5393 detail such that the TSF can be generated without further design decisions.

5394 Source code or hardware diagrams and/or IC hardware design language code or layout data that
5395 are used to build the actual hardware are examples of parts of an implementation representation.
5396 The evaluator samples the implementation representation to gain confidence that it is at the
5397 appropriate level and not, for instance, a pseudo-code level which requires additional design
5398 decisions to be made. The evaluator is encouraged to perform a quick check when first looking at
5399 the implementation representation to assure themselves that the developer is on the right track.
5400 However, the evaluator is also encourage to perform the bulk of this check while working on other
5401 work units that call for examining the implementation; this will ensure the sample examined for
5402 this work unit is relevant.

5403 ISO/IEC 15408-3 ADV_IMP.1.2C: *The implementation representation shall be in the form used by the*
5404 *development personnel.*

5405 **11.5.1.4.2 Work unit ADV_IMP.1-2**

5406 The evaluator **shall check** that the implementation representation is in the form used by
5407 development personnel.

5408 The implementation representation is manipulated by the developer in form that it suitable for
5409 transformation to the actual implementation. For instance, the developer may work with files
5410 containing source code, which is eventually compiled to become part of the TSF. The developer
5411 makes available the implementation representation in the form they use, so that the evaluator may
5412 use automated techniques in the analysis. This also increases the confidence that the
5413 implementation representation examined is actually the one used in the production of the TSF (as
5414 opposed to the case where it is supplied in an alternate presentation format, such as a word
5415 processor document). It should be noted that other forms of the implementation representation
5416 may also be used by the developer; these forms are supplied as well. The overall goal is to supply
5417 the evaluator with the information that will maximise the evaluator's analysis efforts.

5418 The evaluator samples the implementation representation to gain confidence that it is the version
5419 that is usable by the developer. The sample is such that the evaluator has assurance that all areas of
5420 the implementation representation are in conformance with the requirement; however, a complete
5421 examination of the entire implementation representation is unnecessary.

5422 Conventions in some forms of the implementation representation may make it difficult or
5423 impossible to determine from just the implementation representation itself what the actual result
5424 of the compilation or run-time interpretation will be. For example, compiler directives for C
5425 language compilers will cause the compiler to exclude or include entire portions of the code.

5426 Some forms of the implementation representation may require additional information because
5427 they introduce significant barriers to understanding and analysis. Examples include shrouded
5428 source code or source code that has been obfuscated in other ways such that it prevents
5429 understanding and/or analysis. These forms of implementation representation typically result
5430 from by taking a version of the implementation representation that is used by the TOE developer
5431 and running a shrouding or obfuscation program on it. While the shrouded representation is what
5432 is compiled and may be closer to the implementation (in terms of structure) than the original, un-
5433 shrouded representation, supplying such obfuscated code may cause significantly more time to be
5434 spent in analysis tasks involving the representation. When such forms of representation are
5435 created, the components require details on the shrouding tools/algorithms used so that the un-
5436 shrouded representation can be supplied, and the additional information can be used to gain
5437 confidence that the shrouding process does not compromise any security mechanisms.

5438 The evaluator samples the implementation representation to gain confidence that all of the
5439 information needed to interpret the implementation representation has been supplied. Note that
5440 the tools are among those referenced by Tools and techniques (ALC_TAT) components. The
5441 evaluator is encouraged to perform a quick check when first looking at the implementation
5442 representation to assure themselves that the developer is on the right track. However, the
5443 evaluator is also encouraged to perform the bulk of this check while working on other work units

5444 that call for examining the implementation; this will ensure the sample examined for this work unit
5445 is relevant.

5446 ISO/IEC 15408-3 ADV_IMP.1.3C: *The mapping between the TOE design description and the sample of*
5447 *the implementation representation shall demonstrate their correspondence.*

5448 **11.5.1.4.3 Work unit ADV_IMP.1-3**

5449 The evaluator **shall examine** the mapping between the TOE design description and the sample of
5450 the implementation representation to determine that it is accurate.

5451 The evaluator augments the determination of existence (specified in work unit ADV_IMP.1-1) by
5452 verifying the accuracy of a portion of the implementation representation and the TOE design
5453 description. For parts of the TOE design description that are interesting, the evaluator would verify
5454 the implementation representation accurately reflects the description provided in the TOE design
5455 description.

5456 For example, the TOE design description might identify a login module that is used to identify and
5457 authenticate users. If user authentication is sufficiently significant, the evaluator would verify that
5458 the corresponding code in fact implements that service as described in the TOE design description.
5459 It might also be worthwhile to verify that the code accepts the parameters as described in the
5460 functional specification.

5461 It is worth pointing out the developer must choose whether to perform the mapping for the entire
5462 implementation representation, thereby guaranteeing that the chosen sample will be covered, or
5463 waiting for the sample to be chosen before performing the mapping. The first option is likely more
5464 work, but may be completed before the evaluation begins. The second option is less work, but will
5465 produce a suspension of evaluation activity while the necessary evidence is being produced.

5466 **11.5.2 Evaluation of sub-activity (ADV_IMP.2)**

5467

5468 **11.5.2.1 Objectives**

5469 The objective of this sub-activity is to determine that the implementation representation made
5470 available by the developer is suitable for use in other analysis activities; suitability is judged by its
5471 conformance to the requirements for this component.

5472 **11.5.2.2 Input**

5473 The evaluation evidence for this sub-activity is:

5474

5475 a) the implementation representation;

5476 b) the documentation of the development tools, as resulting from ALC_TAT;

5477 c) the TOE design description.

5478 **11.5.2.3 Application notes**

5479 The entire implementation representation is made available to ensure that analysis activities are
5480 not curtailed due to lack of information. This does not, however, imply that all of the representation
5481 is examined in detail when the analysis activities are being performed. This is likely impractical in
5482 almost all cases, in addition to the fact that it most likely will not result in a higher-assurance TOE.

5483 The new aspect for ADV_IMP.2 in comparison to ADV_IMP.1 is that the developer needs to
5484 demonstrate and the evaluator will confirm that the complete implementation representation is
5485 mapped to the TOE design description. This does, however, not imply that all other work units
5486 need an examination of the complete implementation representation. Aspects like appropriate
5487 level of detail and form of the implementation representation can be covered by sampling as for
5488 ADV_IMP.1.

5489 **11.5.2.4 Action ADV_IMP.2.1E**

5490 ISO/IEC 15408-3 ADV_IMP.2.1C *The implementation representation shall define the TSF to a level of*
5491 *detail such that the TSF can be generated without further design decisions.*

5492 **11.5.2.4.1 Work unit ADV_IMP.2-1**

5493 The evaluator **shall check** that the implementation representation defines the TSF to a level of
5494 detail such that the TSF can be generated without further design decisions.

5495 Source code or hardware diagrams and/or IC hardware design language code or layout data that
5496 are used to build the actual hardware are examples of parts of an implementation representation.
5497 The evaluator samples the implementation representation to gain confidence that it is at the
5498 appropriate level and not, for instance, a pseudo-code level which requires additional design
5499 decisions to be made. The evaluator is encouraged to perform a quick check when first looking at
5500 the implementation representation to assure themselves that the developer is on the right track.
5501 However, the evaluator is also encourage to perform the bulk of this check while working on other
5502 work units that call for examining the implementation; this will ensure the sample examined for
5503 this work unit is relevant.

5504 If the evaluator has the possibility to actually execute or witness the "built" procedure used to
5505 transfer the implementation representation into the actual implementation, and to compare the
5506 result to the TOE as delivered, this may provide an easier and at the same time more reliable check
5507 for this work unit (and possibly also for the following one).

5508 ISO/IEC 15408-3 ADV_IMP.2.2C *The implementation representation shall be in the form used by the*
5509 *development personnel.*

5510 **11.5.2.4.2 Work unit ADV_IMP.2-2**

5511 The evaluator **shall check** that the implementation representation is in the form used by
5512 development personnel.

5513 The implementation representation is manipulated by the developer in form that it suitable for
5514 transformation to the actual implementation. For instance, the developer may work with files
5515 containing source code, which is eventually compiled to become part of the TSF. The developer
5516 makes available the implementation representation in the form they use, so that the evaluator may
5517 use automated techniques in the analysis. This also increases the confidence that the
5518 implementation representation examined is actually the one used in the production of the TSF (as
5519 opposed to the case where it is supplied in an alternate presentation format, such as a word
5520 processor document). It should be noted that other forms of the implementation representation
5521 may also be used by the developer; these forms are supplied as well. The overall goal is to supply
5522 the evaluator with the information that will maximise the evaluator's analysis efforts.

5523 The evaluator samples the implementation representation to gain confidence that it is the version
5524 that is usable by the developer. The sample is such that the evaluator has assurance that all areas of
5525 the implementation representation are in conformance with the requirement; however, a complete
5526 examination of the entire implementation representation is unnecessary.

5527 Conventions in some forms of the implementation representation may make it difficult or
5528 impossible to determine from just the implementation representation itself what the actual result

5529 of the compilation or run-time interpretation will be. For example, compiler directives for C
5530 language compilers will cause the compiler to exclude or include entire portions of the code.

5531 Some forms of the implementation representation may require additional information because
5532 they introduce significant barriers to understanding and analysis. Examples include shrouded
5533 source code or source code that has been obfuscated in other ways such that it prevents
5534 understanding and/or analysis. These forms of implementation representation typically result
5535 from by taking a version of the implementation representation that is used by the TOE developer
5536 and running a shrouding or obfuscation program on it. While the shrouded representation is what
5537 is compiled and may be closer to the implementation (in terms of structure) than the original, un-
5538 shrouded representation, supplying such obfuscated code may cause significantly more time to be
5539 spent in analysis tasks involving the representation. When such forms of representation are
5540 created, the components require details on the shrouding tools/algorithms used so that the un-
5541 shrouded representation can be supplied, and the additional information can be used to gain
5542 confidence that the shrouding process does not compromise any security mechanisms.

5543 The evaluator samples the implementation representation to gain confidence that all of the
5544 information needed to interpret the implementation representation has been supplied. Note that
5545 the tools are among those referenced by Tools and techniques (ALC_TAT) components. The
5546 evaluator is encouraged to perform a quick check when first looking at the implementation
5547 representation to assure themselves that the developer is on the right track. However, the
5548 evaluator is also encouraged to perform the bulk of this check while working on other work units
5549 that call for examining the implementation; this will ensure the sample examined for this work unit
5550 is relevant.

5551 ISO/IEC 15408-3 ADV_IMP.2.3C *The mapping between the TOE design description and the entire*
5552 *implementation representation shall demonstrate their correspondence.*

5553 **11.5.2.4.3 Work unit ADV_IMP.2-3**

5554 The evaluator ***shall examine*** the mapping between the TOE design description and the entire
5555 implementation representation to determine that it is accurate.

5556 The evaluator augments the determination of existence (specified in work unit ADV_IMP.2-1) by
5557 verifying the accuracy of the implementation representation and the TOE design description. For
5558 those parts of TOE design description that are interesting, the evaluator would verify the
5559 implementation representation accurately reflects the description provided in the TOE design
5560 description.

5561 For example, the TOE design description might identify a login module that is used to identify and
5562 authenticate users. If user authentication is sufficiently significant, the evaluator would verify that
5563 the corresponding code in fact implements that service as described in the TOE design description.
5564 It might also be worthwhile to verify that the code accepts the parameters as described in the
5565 functional specification.

5566 Usually it will be expected that the evaluator considers at least the functionality required by the
5567 SFRs chosen in the ST and aspects described in the security architecture description as
5568 "interesting" in the sense discussed above. Note however that not all aspects of the security
5569 architecture are necessarily traceable to specific parts of the implementation representation.

5570 It is worth pointing out the developer must perform the mapping for the entire implementation
5571 representation, thereby guaranteeing that the chosen sample will be covered.

5572 **11.5.2.4.4 Work unit ADV_IMP.2-4**

5573 The evaluator ***shall examine*** the mapping between the TOE design description and the entire
5574 implementation representation to determine that it is complete.

5575 Note that the completeness here is relevant in both directions: The complete TOE design needs to
5576 be covered by the implementation representation and all parts of the implementation
5577 representation needs to be mapped to a corresponding part of the TOE design.

5578 In order to confirm that the entire implementation representation is covered by the mapping the
5579 evaluator will not need to examine the content of every part of the implementation representation.
5580 If (in the case of a software TOE) the mapping is for example described by mapping each source
5581 code file to a module in the TOE design description, it will be sufficient if this mapping is plausible
5582 from the role of the source code file the evaluator can conclude from information like the naming of
5583 the source code files, their grouping in subdirectories or their grouping in "built" procedures. Note,
5584 that aspects of accuracy are covered by the preceding work unit.

5585 In order to confirm that the entire design description is covered by the implementation, the
5586 evaluator may either use a similar argument as in the other direction, i. e. that all modules
5587 contained in the TOE design description are mapped to parts of the implementation representation
5588 in a plausible way. In addition, if the evaluator has established in the preceding work unit that all
5589 SFRs and all applicable parts of the security architecture description are traceable to the
5590 implementation representation this may be seen as sufficient evidence that the mapping is
5591 complete.

5592 **11.6 TSF internals (ADV_INT)**

5593 **11.6.1 Evaluation of sub-activity (ADV_INT.1)**

5594 **11.6.1.1 Objectives**

5595 The objective of this sub-activity is to determine whether the defined subset of the TSF is designed
5596 and structured such that the likelihood of flaws is reduced and that maintenance can be more
5597 readily performed without the introduction of flaws.

5598 **11.6.1.2 Input**

5599 The evaluation evidence for this sub-activity is:

- 5600 a) the ST;
- 5601 b) the TOE design description;
- 5602 c) the implementation representation (if ADV_IMP is part of the claimed assurance);
- 5603 d) the TSF internals description and justification;
- 5604 e) the documentation of the coding standards, as resulting from ALC_TAT.

5605 **11.6.1.3 Application notes**

5606 The role of the internals description is to provide evidence of the structure of the design and
5607 implementation of the TSF.

5608 The structure of the design has two aspects: the constituent parts of the TSF and the procedures
5609 used to design the TSF. In cases where the TSF is designed in a manner consistent with the design
5610 represented by the TOE design (see ADV_TDS), the assessment of the TSF design is obvious. In
5611 cases where the design procedures (see ALC_TAT) are being followed, the assessment of the TSF
5612 design procedures is similarly obvious.

5613 In cases where the TSF is implemented using procedure-based software, this structure is assessed
5614 on the basis of its modularity; the modules identified in the internals description are the same as

5615 the modules identified in the TOE design (TOE design (TOE_TDS)). A module consists of one or
5616 more source code files that cannot be decomposed into smaller compilable units.

5617 The use of the assignment in this component levies stricter constraints on the subset of the TSF
5618 that is explicitly identified in the assignment ADV_INT.1.1D than on the remainder of the TSF.
5619 While the entire TSF is to be designed using good engineering principles and result in a well-
5620 structured TSF, only the specified subset is specifically analysed for this characteristic. The
5621 evaluator determines that the developer's application of coding standards result in a TSF that is
5622 understandable.

5623 The primary goal of this component is to ensure the TSF subset's implementation representation is
5624 understandable to facilitate maintenance and analysis (of both the developer and evaluator).

5625 **11.6.1.4 Action ADV_INT.1.1E**

5626 ISO/IEC 15408-3 ADV_INT.1.1C: *The justification shall explain the characteristics used to judge the*
5627 *meaning of "well-structured".*

5628 **11.6.1.4.1 Work unit ADV_INT.1-1**

5629 The evaluator **shall examine** the justification to determine that it identifies the basis for
5630 determining whether the TSF is well-structured.

5631 The evaluator verifies that the criteria for determining the characteristic of being well-structured
5632 are clearly defined in the justification. Acceptable criteria typically originate from industry
5633 standards for the technology discipline. For example, procedural software that executes linearly is
5634 traditionally viewed as well-structured if it adheres to software engineering programming
5635 practises, such as those defined in the IEEE Standard (*IEEE Std 610.12-1990*). For example, it would
5636 identify the criteria for the procedural software portions of the TSF subset:

5637 a) the process used for modular decomposition

5638 b) coding standards used in the development of the implementation

5639 c) a description of the maximum acceptable level of intermodule coupling exhibited by the
5640 TSF subset

5641 d) a description of the minimum acceptable level of cohesion exhibited the modules of the
5642 TSF subset

5643 For other types of technologies used in the TOE - such as non-procedural software (e.g. object-
5644 oriented programming), widespread commodity hardware (e.g. PC microprocessors), and special-
5645 purpose hardware (e.g. smart-card processors) - the evaluator should seek guidance from the
5646 evaluation authority for determining the adequacy of criteria for being "well-structured".

5647 ISO/IEC 15408-3 ADV_INT.1.2C: *The TSF internals description shall demonstrate that the assigned*
5648 *subset of the TSF is well-structured.*

5649 **11.6.1.4.2 Work unit ADV_INT.1-2**

5650 The evaluator **shall check** the TSF internals description to determine that it identifies the Assigned
5651 subset of the TSF.

5652 This subset may be identified in terms of the internals of the TSF at any layer of abstraction. For
5653 example, it may be in terms of the structural elements of the TSF as identified in the TOE design
5654 (e.g. the audit subsystem), or in terms of the implementation (e.g. *encrypt.c* and *decrypt.c* files, or
5655 the 6227 IC chip).

5656 It is insufficient to identify this subset in terms of the claimed SFRs (e.g. the portion of the TSF that
5657 provide anonymity as defined in FPR_ANO.2) because this does not indicate where to focus the
5658 analysis.

5659 **11.6.1.4.3 Work unit ADV_INT.1-3**

5660 The evaluator *shall examine* the TSF internals description to determine that it demonstrates that
5661 the assigned TSF subset is well-structured.

5662 The evaluator examines the internals description to ensure that it provides a sound explanation of
5663 how the TSF subset meets the criteria from ADV_INT.1-1

5664 For example, it would explain how the procedural software portions of the TSF subset meets the
5665 following:

5666 a) that there is a one-to-one correspondence between the modules identified in the TSF
5667 subset and the modules described in the TOE design (ADV_TDS)

5668 b) how the TSF design is a reflection of the modular decomposition process

5669 c) a justification for all instances where the coding standards were not used or met

5670 d) a justification for any coupling or cohesion outside the acceptable bounds

5671 **11.6.1.5 Action ADV_INT.1.2E**

5672 **11.6.1.5.1 Work unit ADV_INT.1-4**

5673 The evaluator *shall determine* that the TOE design for the assigned TSF subset is well-structured.

5674 The evaluator examines a sample of the TOE design to verify the accuracy of the justification. For
5675 example, a sample of the TOE design is analysed to determine its adherence to the design
5676 standards, etc. As with all areas where the evaluator performs activities on a subset the evaluator
5677 provides a justification of the sample size and scope

5678 The description of the TOE's decomposition into subsystems and modules will make the argument
5679 that the TSF subset is well-structured self-evident. Verification that the procedures for structuring
5680 the TSF (as examined in ALC_TAT) are being followed will make it self-evident that the TSF subset
5681 is well-structured.

5682 **11.6.1.5.2 Work unit ADV_INT.1-5**

5683 The evaluator *shall determine* that the assigned TSF subset is well-structured.

5684 If ADV_IMP is not part of the claimed assurance, then this work unit is not applicable and is
5685 therefore considered to be satisfied.

5686 The evaluator examines a sample of the TSF subset to verify the accuracy of the internals
5687 description. For example, a sample of the procedural software portions of the TSF subset is
5688 analysed to determine its cohesion and coupling, its adherence to the coding standards, etc. As with
5689 all areas where the evaluator performs activities on a subset the evaluator provides a justification
5690 of the sample size and scope.

5691 **11.6.2 Evaluation of sub-activity (ADV_INT.2)**5692 **11.6.2.1 Objectives**

5693 The objective of this sub-activity is to determine whether the TSF is designed and structured such
5694 that the likelihood of flaws is reduced and that maintenance can be more readily performed
5695 without the introduction of flaws.

5696 **11.6.2.2 Input**

5697 The evaluation evidence for this sub-activity is:

- 5698 a) the modular design description;
- 5699 b) the implementation representation (if ADV_IMP is part of the claimed assurance));
- 5700 c) the TSF internals description;
- 5701 d) the documentation of the coding standards, as resulting from ALC_TAT.

5702 **11.6.2.3 Application notes**

5703 The role of the internals description is to provide evidence of the structure of the design and
5704 implementation of the TSF.

5705 The structure of the design has two aspects: the constituent parts of the TSF and the procedures
5706 used to design the TSF. In cases where the TSF is designed in a manner consistent with the design
5707 represented by the TOE design (see ADV_TDS), the assessment of the TSF design is obvious. In
5708 cases where the design procedures (see ALC_TAT) are being followed, the assessment of the TSF
5709 design procedures is similarly obvious.

5710 In cases where the TSF is implemented using procedure-based software, this structure is assessed
5711 on the basis of its modularity; the modules identified in the internals description are the same as
5712 the modules identified in the TOE design (TOE design (ADV_TDS)). A module consists of one or
5713 more source code files that cannot be decomposed into smaller compilable units.

5714 The primary goal of this component is to ensure the TSF's implementation representation is
5715 understandable to facilitate maintenance and analysis (of both the developer and evaluator).

5716 **11.6.2.4 Action ADV_INT.2.1E**

5717 ISO/IEC 15408-3 ADV_INT.2.1C: *The justification shall describe the characteristics used to judge the*
5718 *meaning of "well-structured".*

5719 **11.6.2.4.1 Work unit ADV_INT.2-1**

5720 The evaluator **shall examine** the justification to determine that it identifies the basis for
5721 determining whether the TSF is well-structured.

5722 The evaluator verifies that the criteria for determining the characteristic of being well-structured
5723 are clearly defined in the justification. Acceptable criteria typically originate from industry
5724 standards for the technology discipline. For example, procedural software that executes linearly is
5725 traditionally viewed as well-structured if it adheres to software engineering programming
5726 practises, such as those defined in the IEEE Standard (*IEEE Std 610.12-1990*). For example, it would
5727 identify the criteria for the procedural software portions of the TSF:

- 5728 a) the process used for modular decomposition

- 5729 b) coding standards used in the development of the implementation
- 5730 c) a description of the maximum acceptable level of intermodule coupling exhibited by the
5731 TSF
- 5732 d) a description of the minimum acceptable level of cohesion exhibited the modules of the
5733 TSF
- 5734 For other types of technologies used in the TOE - such as non-procedural software (e.g. object-
5735 oriented programming), widespread commodity hardware (e.g. PC microprocessors), and special-
5736 purpose hardware (e.g. smart-card processors) - the evaluation authority should be consulted for
5737 determining the adequacy of criteria for being “well-structured”.
- 5738 ISO/IEC 15408-3 ADV_INT.2.2C: *The TSF internals description shall demonstrate that the entire TSF*
5739 *is well-structured.*
- 5740 **11.6.2.4.2 Work unit ADV_INT.2-2**
- 5741 The evaluator ***shall examine*** the TSF internals description to determine that it demonstrates that
5742 the TSF is well-structured.
- 5743 The evaluator examines the internals description to ensure that it provides a sound explanation of
5744 how the TSF meets the criteria from ADV_INT.2-1
- 5745 For example, it would explain how the procedural software portions of the TSF meet the following:
- 5746 a) that there is a one-to-one correspondence between the modules identified in the TSF and
5747 the modules described in the TOE design (ADV_TDS)
- 5748 b) how the TSF design is a reflection of the modular decomposition process
- 5749 c) a justification for all instances where the coding standards were not used or met
- 5750 d) a justification for any coupling or cohesion outside the acceptable bounds
- 5751 **11.6.2.5 Action ADV_INT.2.2E**
- 5752 **11.6.2.5.1 Work unit ADV_INT.2-3**
- 5753 The evaluator ***shall determine*** that the TOE design is well-structured.
- 5754 The evaluator examines the TOE design of a sample of the TSF to verify the accuracy of the
5755 justification. For example, a sample of the TOE design is analysed to determine its adherence to the
5756 design standards, etc. As with all areas where the evaluator performs activities on a subset the
5757 evaluator provides a justification of the sample size and scope
- 5758 The description of the TOE's decomposition into subsystems and modules will make the argument
5759 that the TSF subset is well-structured self-evident. Verification that the procedures for structuring
5760 the TSF (as examined in ALC_TAT) are being followed will make it self-evident that the TSF subset
5761 is well-structured.
- 5762 **11.6.2.5.2 Work unit ADV_INT.2-4**
- 5763 The evaluator ***shall determine*** that the TSF is well-structured.
- 5764 If ADV_IMP is not part of the claimed assurance, then this work unit is not applicable and is
5765 therefore considered to be satisfied.

5766 The evaluator examines a sample of the TSF to verify the accuracy of the internals description. For
 5767 example, a sample of the procedural software portions of the TSF is analysed to determine its
 5768 cohesion and coupling, its adherence to the coding standards, etc. As with all areas where the
 5769 evaluator performs activities on a subset the evaluator provides a justification of the sample size
 5770 and scope.

5771 **11.6.3 Evaluation of sub-activity (ADV_INT.3)**

5772 **11.6.3.1 Objectives**

5773 The objective of this sub-activity is to determine whether the TSF is designed and structured such
 5774 that the likelihood of flaws is reduced and that maintenance can be more readily performed
 5775 without the introduction of flaws.

5776 **11.6.3.2 Input**

5777 The evaluation evidence for this sub-activity is:

- 5778 a) the modular design description;
- 5779 b) the implementation representation (if ADV_IMP is part of the claimed
 5780 assurance);
- 5781 c) the TSF internals description;
- 5782 d) the documentation of the coding standards, as resulting from ALC_TAT.

5783 **11.6.3.3 Application notes**

5784 The role of the internals description is to provide evidence of the structure of the design and
 5785 implementation of the TSF.

5786 The structure of the design has two aspects: the constituent parts of the TSF and the procedures
 5787 used to design the TSF. In cases where the TSF is designed in a manner consistent with the design
 5788 represented by the TOE design (see ADV_TDS), the assessment of the TSF design is obvious. In
 5789 cases where the design procedures (see ALC_TAT) are being followed, the assessment of the TSF
 5790 design procedures is similarly obvious.

5791 In cases where the TSF is implemented using procedure-based software, this structure is assessed
 5792 on the basis of its modularity; the modules identified in the internals description are the same as
 5793 the modules identified in the TOE design (TOE design (ADV_TDS)). A module consists of one or
 5794 more source code files that cannot be decomposed into smaller compilable units.

5795 The primary goal of this component is to ensure the TSF's implementation representation is
 5796 understandable to facilitate maintenance and analysis (of both the developer and evaluator).

5797 **11.6.3.4 Action ADV_INT.3.1E**

5798 ADV_INT.3.1C *The justification shall describe the characteristics used to judge the meaning of*
 5799 *"well-structured" and "complex".*

5800 **11.6.3.4.1 Work unit ADV_INT.3-1**

5801 The evaluator **shall examine** the justification to determine that it identifies the basis for
 5802 determining whether the TSF is "well-structured" and "not overly complex".

5803 The evaluator verifies that the criteria for determining the characteristic of being "well-structured"
 5804 and "complex" are clearly defined in the justification. Acceptable criteria typically originate from

5805 industry standards for the technology discipline. For example, procedural software that executes
5806 linearly is traditionally viewed as well-structured if it adheres to software engineering
5807 programming practises, such as those defined in the IEEE Standard (IEEE Std 610.12-1990). For
5808 example, it would identify the criteria for the procedural software portions of the TSF:

- 5809 a) the process used for modular decomposition
- 5810 b) coding standards used in the development of the implementation
- 5811 c) a description of the maximum acceptable level of intermodule coupling
5812 exhibited by the TSF
- 5813 d) a description of the minimum acceptable level of cohesion exhibited the
5814 modules of the TSF

5815 Complexity can for example be measured in the number of decision points and logical paths of
5816 execution that code takes. Software engineering literature cites complexity as a negative
5817 characteristic of software because it impedes understanding of the logic and flow of the code.
5818 Another impediment to the understanding of code is the presence of code that is unnecessary, in
5819 that it is unused or redundant.

5820 1 Design complexity minimisation is a key characteristic of a reference validation
5821 mechanism, the purpose of which is to arrive at a TSF that is easily understood so
5822 that it can be completely analysed.

5823 2 See also CC 3.1, Part 3, Annex A.3 for additional information on TSF internals.

5824 3 The consideration in that annex and those made in the preceding paragraphs of this work
5825 unit are mainly derived from common knowledge about procedural software. For
5826 other types of technologies used in the TOE - such as non-procedural software
5827 (e.g. object-oriented programming), widespread commodity hardware (e.g. PC
5828 microprocessors), and special-purpose hardware (e.g. smart-card processors) - the
5829 evaluation authority should be consulted for determining the adequacy of criteria
5830 for being "well-structured" and "not overly complex".

5831 4 The evaluator is reminded to be open for plausible definitions given by the developer. If,
5832 for example, a smart card developer can justify that the metrics used by him to
5833 measure complexity are an industry standard in their field, this should usually be
5834 sufficient for acceptance of such metrics.

5835 5

5836 ISO/IEC 15408-3 ADV_INT.3.2C *The TSF internals description shall demonstrate that the entire TSF*
5837 *is well-structured and is not overly complex.*

5838 11.6.3.4.2 Work unit ADV_INT.3-2

5839 The evaluator **shall examine** the TSF internals description to determine that it demonstrates that
5840 the TSF is well-structured and not overly complex.

5841 The evaluator examines the internals description to ensure that it provides a sound explanation of
5842 how the TSF meets the criteria from ADV_INT.3-1

5843 For example, it would explain how the procedural software portions of the TSF meet the following:

- 5844 a) that there is a one-to-one correspondence between the modules identified in the
5845 TSF and the modules described in the TOE design (ADV_TDS)
- 5846 b) how the TSF design is a reflection of the modular decomposition process

- 5847 c) a justification for all instances where the coding standards were not used or met
- 5848 d) a justification for any coupling or cohesion outside the acceptable bounds
- 5849 e) how the modular decomposition process reduces complexity

5850 **11.6.3.5 Action ADV_INT.3.2E**

5851 **11.6.3.5.1 Work unit ADV_INT.3-3**

5852 The evaluator *shall determine* that the entire TOE design is well-structured and not overly
5853 complex.

5854 The evaluator examines the TOE design description of the TSF to verify the accuracy of the
5855 justification. For example, a sample of the TOE design is analysed to determine its adherence to the
5856 design standards, etc. As with all areas where the evaluator performs activities on a subset the
5857 evaluator provides a justification of the sample size and scope

5858 The description of the TOE's decomposition into subsystems and modules will make the argument
5859 that the TSF is well-structured self-evident. Verification that the procedures for structuring the TSF
5860 (as examined in ALC_TAT) are being followed will make it self-evident that the TSF is well-
5861 structured.

5862 Using the metrics defined by the developer for measuring the complexity of the design will show if
5863 the metrics is met. If the metrics is only defined for the implementation representation and not for
5864 the TOE design (note that adequateness of the metrics was considered already in work unit
5865 ADV_INT.3-1), there may be no need for using the metrics in this work unit, the complexity-issue is
5866 then covered by the next work unit.

5867 **11.6.3.5.2 Work unit ADV_INT.3-4**

5868 The evaluator *shall determine* that the entire TSF is well-structured and not overly complex.

5869 If ADV_IMP is not part of the claimed assurance, then this work unit is not applicable and is
5870 therefore considered to be satisfied.

5871 The evaluator examines a sample of the TSF to verify the accuracy of the internal description. For
5872 example, a sample of the procedural software portions of the TSF is analysed to determine its
5873 cohesion and coupling, its adherence to the coding standards, etc. As with all areas where the
5874 evaluator performs activities on a subset the evaluator provides a justification of the sample size
5875 and scope.

5876 Similarly the evaluator applies the metric for complexity as defined by the developer and examined
5877 in work unit ADV_INT.3-1 to either a sample of the implementation representation or the complete
5878 implementation representation (this may depend on the metric) and verifies that the metric is in
5879 fact met. The evaluator may only restrict their application of the metrics to a sample if the
5880 developer has provided the results of the application of the metrics for the entire TSF and the
5881 sampling serves as means to convince the evaluator that the application as done by the developer
5882 was correct (similar to the evaluator's sampling of functional testing already done by the
5883 developer).

5884 **11.7 Security policy modelling (ADV_SPM)**

5885 **11.7.1 Evaluation of sub-activity (ADV_SPM.1)**

5886 **11.7.1.1 Objectives**

5887 The objectives of this sub-activity are to determine whether the formal security policy model of the
5888 TSF clearly and consistently describes the rules and characteristics of the security policies and
5889 whether this description corresponds with the description of security functions in the functional
5890 specification.

5891 **11.7.1.2 Input**

5892 The evaluation evidence for this sub-activity is:

5893 a) the ST;

5894 the functional specification;

5895 formal security policy model (ADV_SPM.1.1D);

5896 formal proof of correspondence between the model and any formal functional specification
5897 (ADV_SPM.1.3D);

5898 demonstration of correspondence between the model and the functional specification
5899 (ADV_SPM.1.4D).

5900 **11.7.1.3 Application notes**

5901 This activity applies to cases where the developer has provided a formal security policy model of
5902 the TOE.

5903 A formal TOE security policy model is a representation of the rules (synonymously termed
5904 "principles") of security policies and characteristics of the TSF behaviour in mathematical terms.
5905 Their formal counterparts are called security properties and security features, respectively. The
5906 representation includes but is not limited to algebraic specifications, finite state machines and logic
5907 formalisms strong enough to formally infer the properties from the features. The formal TSP model
5908 is accompanied by an informal interpretation explaining how the rules and characteristics are
5909 mapped to the respective properties and features.

5910 The creation of a formal security policy model helps to identify and eliminate ambiguous,
5911 inconsistent, contradictory, or unenforceable security policy elements. Once the TOE has been built,
5912 the formal model serves the evaluation effort by contributing to the evaluator's judgement of how
5913 well the developer has understood the security functionality being implemented and whether
5914 there are inconsistencies between the security requirements and the TOE design. The confidence in
5915 the model is accompanied by a proof that it contains no inconsistencies.

5916 A formal security model is a precise formal presentation of the important aspects of security and
5917 their relationship to the behaviour of the TOE; it identifies the set of rules (principles) that defines
5918 the TOE security policy and the set of practises (characteristics) that regulates how the TSF
5919 manages, protects, and otherwise controls the system resources. The model includes the set of
5920 restrictions and properties that specify how information and computing resources are prevented
5921 from being used to violate the SFRs, accompanied by a persuasive set of engineering arguments
5922 showing that these restrictions and properties play a key role in the enforcement of the SFRs. It
5923 consists both of the formalisms that express the security functionality, as well as ancillary text to
5924 explain the model and to provide it with context. The security behaviour of the TSF is modelled
5925 both in terms of external behaviour (i.e. how the TSF interacts with the rest of the TOE and with its
5926 operational environment), as well as its internal behaviour.

5927 The Security Policy Model of the TOE is informally abstracted from its realisation by considering
 5928 the proposed security requirements of the ST. The informal abstraction is taken to be successful if
 5929 the TOE's principles turn out to be enforced by its characteristics. The purpose of formal methods
 5930 lies within the enhancement of the rigour of enforcement. Informal arguments are always prone to
 5931 fallacies; especially if relationships among subjects, objects and operations get more and more
 5932 involved. In order to minimise the risk of insecure state arrivals the rules and characteristics of the
 5933 security policy model are mapped to respective properties and features within some formal system,
 5934 whose rigour and strength can afterwards be used to obtain the security properties by means of
 5935 theorems and formal proof.

5936 While the term "formal security policy model" is used in academic circles, the CC's approach has no
 5937 fixed definition of "security"; it would equate to whatever SFRs are being claimed. Therefore, the
 5938 formal security policy model is merely a formal representation of the set of SFRs being claimed.

5939 The term security policy has traditionally been associated with only access control policies,
 5940 whether label-based (mandatory access control) or user-based (discretionary access control).
 5941 However, a security policy is not limited to access control; there are also audit policies,
 5942 identification policies, authentication policies, encryption policies, management policies, and any
 5943 other security policies that are enforced by the TOE, as described in the PP/ST. ADV_SPM.1.1D
 5944 contains an assignment for identifying these policies that are formally modelled.

5945 It is recognized that not all policies can be formally modelled for all TOEs. This is because either a
 5946 given policy can not be formally modelled in the otherwise well suited framework, or because the
 5947 nature of the TOE renders impossible the modelling of policies that would otherwise be possible to
 5948 model.

5949 **11.7.1.4 Action ADV_SPM.1.1E**

5950 ADV_SPM.1.1C *The model shall be in a formal style, supported by explanatory text as required,*
 5951 *and identify the security policies of the TSF that are modelled.*

5952 **11.7.1.4.1 Work unit ADV_SPM.1-1**

5953 The evaluator **shall examine** the TOE security policy model to determine that it is written in a
 5954 formal style.

5955 The evaluator identifies the formal framework upon which the TOE security policy model is based
 5956 and ensures that it is founded on well established mathematical concepts. They also identify the
 5957 security properties and features addressed in the application notes and ensure the formalization of
 5958 at least one security policy.

5959 For guidance on formal methods refer to ISO/IEC 15408-3

5960 **11.7.1.4.2 Work unit ADV_SPM.1-2**

5961 The evaluator **shall examine** the TOE security policy model to determine that it contains all
 5962 necessary informal explanatory text.

5963 Supporting narrative descriptions are necessary for all parts of the model (for example, to make
 5964 clear the meaning of any formal notation and how they are used) including the security properties
 5965 and features.

5966 **11.7.1.4.3 Work unit ADV_SPM.1-3**

5967 The evaluator **shall examine** the TOE security policy model to determine that all security policies
 5968 of the TSF are identified that are modelled.

5969 The evaluator determines whether the SPM identifies the security policies for which a model is
5970 provided, identifying the relevant portions of the statement of SFRs that comprise each of the
5971 modelled policies.

5972 The evaluator determines whether the list of security policies identified by the SPM is consistent
5973 with the assignment of ADV_SPM.1.1D in the ST.

5974 The evaluator determines whether for each security policy identified by the SPM a model is in fact
5975 provided.

5976 ADV_SPM.1.2C *For all policies that are modelled, the model shall define security for the TOE and*
5977 *provide a formal proof that the TOE cannot reach a state that is not secure.*

5978 **11.7.1.4.4 Work unit ADV_SPM.1-4**

5979 The evaluator *shall examine* the principles and characteristics of the security policies to determine
5980 that the modelled security behaviour of the TOE is clearly articulated.

5981 The security policies are expressed in terms of security principles (rules) which are modelled by
5982 security properties and define the secure state of the TOE. For example, a model based on state
5983 transitions could describe the security policies in terms of principles of its states, identify its initial
5984 state, and define what it means to be a secure state.

5985 The evaluator determines that the security policies are reflected within their formal counterparts
5986 of the TSP model.

5987 The TOE security behaviour is expressed in terms of security characteristics (i.e. portions of TOE
5988 security functionality managing, protecting, and otherwise controlling the system resources
5989 including attributes and conditions of the TOE) which are modelled by security features. For
5990 example, a model based on state transitions could describe the characteristics as possible actions
5991 in each secure state in a level of detail sufficient to decide into which state the TOE will be
5992 transformed by that action.

5993 Together the security principles and characteristics describe the entire security posture of the TOE.

5994 In the context of a formal TOE security policy model the security behaviour is considered to be
5995 clearly articulated only if an adequate mapping from principles and characteristics to their
5996 respective formal counterparts properties and features has been given. The mapping is considered
5997 to be adequate if the level of abstraction from the TOE's realization is detailed enough to allow for
5998 correct identification of all security objectives and the relation to the security environment.

5999 The above condition for clear articulation is necessary but not sufficient. An informal
6000 interpretation of all formal concepts (including attributes, predicates and variables, if available)
6001 must be provided in order to make clear their intended meaning.

6002 **11.7.1.4.5 Work unit ADV_SPM.1-5**

6003 The evaluator *shall examine* the TOE security policy model rationale to determine that it formally
6004 proves that the security features enforce the security properties.

6005 To determine the enforcement, the evaluator considers the security properties and the security
6006 features and verifies that the arguments used in the proof are valid. The proof of correspondence
6007 between the security properties and the security features shall be formal.

6008 The validity of the security properties shall mean that the TOE is in a secure state. By this, the
6009 evaluator confirms by means of the rationale that the TOE never reaches an insecure state.

6010 **11.7.1.4.6 Work unit ADV_SPM.1-6**

6011 The evaluator *shall examine* the TOE security policy model rationale to determine that it proves
6012 the internal consistency of the TOE security policy model.

6013 The proof shall show the absence of contradictions within the TOE security policy model. In
6014 determining the absence of contradictions, the evaluator verifies that the arguments used in the
6015 proof are valid.

6016 Since the TOE security policy model is formal, the proof of its internal consistency shall be formal.
6017 It is recognized that a complete formal proof of the internal consistency of the TOE security policy
6018 model usually is not possible due to the fundamental nature of formal frameworks. Generally, it is
6019 sufficient to generate evidence using formal proofs based on the specific TOE security policy model
6020 that prove the internal consistency by means of a combination with generic arguments of the
6021 formal framework.

6022 ADV_SPM.1.3C *The correspondence between the model and the functional specification shall be at*
6023 *the correct level of formality.*

6024 **11.7.1.4.7 Work unit ADV_SPM.1-7**

6025 The evaluator *shall examine* the correspondence between the model and the functional
6026 specification to determine that a semiformal demonstration of correspondence between the model
6027 and any semiformal functional specification is provided.

6028 This work unit is only applicable to a semiformal presentation of the functional specification, which
6029 is required by ADV_FSP.5.2C.

6030 A semiformal correspondence is one that results from a structured approach with a substantial
6031 degree of rigor (in terms of completeness and correctness), but is not as rigorous as a
6032 mathematical proof. Such a semiformal correspondence limits the subjective interpretations of its
6033 terms, and so it provides less ambiguity than would exist in an informal correspondence.

6034 For guidance on semiformal methods refer to Annex 3.1.1 'Semiformal and formal methods'.

6035 **11.7.1.4.8 Work unit ADV_SPM.1-8**

6036 The evaluator *shall examine* the correspondence between the model and the functional
6037 specification to determine that a formal proof of correspondence between the model and any
6038 formal functional specification is provided.

6039 This work unit is only applicable to a formal presentation of the functional specification, which is
6040 required by ADV_FSP.6.2D.

6041 There should be a formal proof of correspondence between the model and any formal functional
6042 specification.

6043 The formal proof of correspondence removes all subjective interpretations of its terms by enlisting
6044 well-established mathematical concepts to define the syntax and semantics of the formal notation
6045 and uses rules that support logical reasoning. The security features within the TOE (which are
6046 identified in the formal TSP model) are expressed in a formal specification language and shown to
6047 be satisfied by the formal specification.

6048 For guidance on formal methods refer to ISO/IEC 15408-3.

6049 ADV_SPM.1.4C *The correspondence shall show that the functional specification is consistent and*
6050 *complete with respect to the model.*

6051 **11.7.1.4.9 Work unit ADV_SPM.1-9**

6052 The evaluator *shall examine* the correspondence to determine that the behaviour at the TSF
6053 interfaces (as articulated in the functional specification) is complete with respect to the behaviour
6054 modelled by the security features.

6055 The term “correspondence” here means both the formal proof of correspondence between the
6056 formal SPM and any formal FSP required by ADV_SPM.1.2D and the demonstration of
6057 correspondence between the formal SPM and the FSP required by ADV_SPM.1.3D.

6058 In determining completeness of the correspondence, the evaluator considers the description of
6059 TSFI behaviour and maps adequate portions (characteristics) to corresponding features of the TSP
6060 model. The demonstration should show that all characteristics belonging to policies that are
6061 required to be modelled have an associated feature description in the TOE security policy model,
6062 and that each feature of the TSP model does occur in the mapping.

6063 Abstention from formally modelling TSFI behaviour always calls for justification on the developer’s
6064 side (also confer the application notes above).

6065 **11.7.1.4.10 Work unit ADV_SPM.1-10**

6066 The evaluator *shall examine* the correspondence to determine that the behaviour at the TSF
6067 interfaces (as articulated in the functional specification) is consistent with respect to the behaviour
6068 modelled by the security features.

6069 The term “correspondence” here means both the formal proof of correspondence between the
6070 formal SPM and any formal FSP required by ADV_SPM.1.3D and the demonstration of
6071 correspondence between the SPM and the FSP required by ADV_SPM.1.4D.

6072 The meaning of consistency reflects the conventional understanding in contrast to the internal
6073 consistency concept of work unit ADV_SPM.1-6.

6074 In determining consistency, the evaluator resumes the mapping of TSFI behaviour to security
6075 features established in the preceding work unit and verifies that the correspondence shows that
6076 each security feature of the TSP model accurately reflects the corresponding TSFI behaviour.

6077 For example, if TSFI behaviour dealt with access management on the granularity of single
6078 individuals, then a TSP model describing the security behaviour of the TOE in terms of groups of
6079 users would not be consistent. Likewise, if TSFI behaviour dealt with access management for
6080 groups of users, then a TSP model describing the security behaviour of the TOE in terms of
6081 individual users would also not be consistent.

6082 As another example, if remote untrusted users had to pass more stringent authentication
6083 procedures than administrators whose only point of access were within a physically-protected
6084 area, then this difference in authentication procedures had to be reflected in the security features.

6085 **11.8 TOE design (ADV_TDS)**

6086 **11.8.1 Evaluation of sub-activity (ADV_TDS.1)**

6087 **11.8.1.1 Input**

6088 The evaluation evidence for this sub-activity is:

6089 a) the ST;

6090 b) the functional specification;

6091 c) security architecture description;

6092 d) the TOE design.

6093 **11.8.1.2 Action ADV_TDS.1.1E**

6094 ISO/IEC 15408-3 ADV_TDS.1.1C: *The design shall describe the structure of the TOE in terms of*
6095 *subsystems.*

6096 **11.8.1.2.1 Work unit ADV_TDS.1-1**

6097 The evaluator **shall examine** the TOE design to determine that the structure of the entire TOE is
6098 described in terms of subsystems.

6099 The evaluator ensures that all of the subsystems of the TOE are identified. This description of the
6100 TOE will be used as input to work unit ADV_TDS.1-2, where the parts of the TOE that make up the
6101 TSF are identified. That is, this requirement is on the entire TOE rather than on only the TSF.

6102 The TOE (and TSF) may be described in multiple layers of abstraction (i.e. subsystems and
6103 modules). Depending upon the complexity of the TOE, its design may be described in terms of
6104 subsystems and modules, as described in ISO/IEC 15408-3 **Annex A.4, ADV_TDS: Subsystems and**
6105 **Modules**. At this level of assurance, the decomposition only need be at the “subsystem” level.

6106 In performing this activity, the evaluator examines other evidence presented for the TOE (e.g., ST,
6107 operator user guidance) to determine that the description of the TOE in such evidence is consistent
6108 with the description contained in the TOE design.

6109 ISO/IEC 15408-3 ADV_TDS.1.2C: *The design shall identify all subsystems of the TSF.*

6110 **11.8.1.2.2 Work unit ADV_TDS.1-2**

6111 The evaluator **shall examine** the TOE design to determine that all subsystems of the TSF are
6112 identified.

6113 In work unit ADV_TDS.1-1 all of the subsystems of the TOE were identified, and a determination
6114 made that the non-TSF subsystems were correctly characterised. Building on that work, the
6115 subsystems that were not characterised as non-TSF subsystems should be precisely identified. The
6116 evaluator determines that, of the hardware and software installed and configured according to the
6117 Preparative procedures (AGD_PRE) guidance, each subsystem has been accounted for as either one
6118 that is part of the TSF, or one that is not.

6119 ISO/IEC 15408-3 ADV_TDS.1.3C: *The design shall describe the behaviour of each SFR-supporting or*
6120 *SFR-non-interfering TSF subsystem in sufficient detail to determine that it is not SFR-enforcing.*

6121 **11.8.1.2.3 Work unit ADV_TDS.1-3**

6122 The evaluator **shall examine** the TOE design to determine that each SFR-supporting or SFR-non-
6123 interfering subsystem of the TSF is described such that the evaluator can determine that the
6124 subsystem is SFR-supporting or SFR-non-interfering.

6125 SFR-supporting and SFR-non-interfering subsystems do not need to be described in detail as to
6126 how they function in the system. However, the evaluator makes a determination, based on the
6127 evidence provided by the developer, that the subsystems that do not have high-level descriptions
6128 are SFR-supporting or SFR-non-interfering. Note that if the developer provides a uniform level of
6129 detailed documentation then this work unit will be largely satisfied, since the point of categorising
6130 the subsystems is to allow the developer to provide less information for SFR-supporting and SFR-
6131 non-interfering subsystems than for SFR-enforcing subsystems.

6132 An SFR-supporting subsystem is one that is depended on by an SFR-enforcing subsystem in order
6133 to implement an SFR, but does not play as direct a role as an SFR-enforcing subsystem. An SFR-
6134 non-interfering subsystem is one that is not depended upon, in either a supporting or enforcing
6135 role, to implement an SFR.

6136 ISO/IEC 15408-3 ADV_TDS.1.4C: *The design shall summarise the SFR-enforcing behaviour of the*
6137 *SFR-enforcing subsystems.*

6138 **11.8.1.2.4 Work unit ADV_TDS.1-4**

6139 The evaluator **shall examine** the TOE design to determine that it provides a complete, accurate,
6140 and high-level summary of the SFR-enforcing behaviour of the SFR-enforcing subsystems.

6141 The developer may designate subsystems as SFR-enforcing, SFR-supporting, and SFR non-
6142 interfering, but these “tags” are used only to describe the amount and type of information the
6143 developer must provide, and can be used to limit the amount of information the developer has to
6144 develop if their engineering process does not produce the documentation required. Whether the
6145 subsystems have been categorised by the developer or not, it is the evaluator's responsibility to
6146 determine that the subsystems have the appropriate information for their role (SFR-enforcing, etc.)
6147 in the TOE, and to obtain the appropriate information from the developer should the developer fail
6148 to provide the required information for a particular subsystem.

6149 SFR-enforcing behaviour refers to how a subsystem provides the functionality that implements an
6150 SFR. The goal of evaluator's assessment is to give the evaluator with an understanding of the way
6151 each SFR-enforcing subsystem works. The information provided for the behaviour summary does
6152 not have to be as detailed as that provided by the behaviour description. For example, data
6153 structures or data items will likely not need to be described in detail. It is the evaluator's
6154 determination, however, with respect to what “high-level” means for a particular TOE, and the
6155 evaluator obtains enough information from the developer (even if it turns out to be equivalent to
6156 information provided for subsystem behaviour) to make a sound verdict for this work unit.

6157 The evaluator is cautioned, however, that “perfect” assurance is not a goal nor required by this
6158 work unit, so judgement will have to be exercised in determine the amount and composition of the
6159 evidence required to make a verdict on this work unit.

6160 To determine completeness and accuracy, the evaluator examines other information available (e.g.,
6161 functional specification, security architecture description). Summaries of functionality in these
6162 documents should be consistent with what is provided for evidence for this work unit.

6163 ISO/IEC 15408-3 ADV_TDS.1.5C: *The design shall provide a description of the interactions among*
6164 *SFR-enforcing subsystems of the TSF, and between the SFR-enforcing subsystems of the TSF and other*
6165 *subsystems of the TSF.*

6166 **11.8.1.2.5 Work unit ADV_TDS.1-5**

6167 The evaluator **shall examine** the TOE design to determine that interactions between the
6168 subsystems of the TSF are described.

6169 The goal of describing the interactions between the SFR-enforcing subsystems and other
6170 subsystems is to help provide the reader a better understanding of how the TSF performs its
6171 functions. These interactions do not need to be characterised at the implementation level (e.g.,
6172 parameters passed from one routine in a subsystem to a routine in a different subsystem; global
6173 variables; hardware signals (e.g., interrupts) from a hardware subsystem to an interrupt-handling
6174 subsystem), but the data elements identified for a particular subsystem that are going to be used by
6175 another subsystem need to be covered in this discussion. Any control relationships between
6176 subsystems (e.g., a subsystem responsible for configuring a rule base for a firewall system and the
6177 subsystem that actually implements these rules) should also be described.

6178 The evaluators need to use their own judgement in assessing the completeness of the description.
 6179 If the reason for an interaction is unclear, or if there are SFR-related interactions (discovered, for
 6180 instance, in examining the descriptions of subsystem behaviour) that do not appear to be described,
 6181 the evaluator ensures that this information is provided by the developer. However, if the evaluator
 6182 can determine that interactions among a particular set of subsystems, while incompletely
 6183 described by the developer, will not aid in understanding the overall functionality nor security
 6184 functionality provided by the TSF, then the evaluator may choose to consider the description
 6185 sufficient, and not pursue completeness for its own sake.

6186 ISO/IEC 15408-3 ADV_TDS.1.6C: *The mapping shall demonstrate that all TSFIs trace to the*
 6187 *behaviour described in the TOE design that they invoke.*

6188 **11.8.1.2.6 Work unit ADV_TDS.1-6**

6189 The evaluator **shall examine** the TOE design to determine that it contains a complete and accurate
 6190 mapping from the TSFI described in the functional specification to the subsystems of the TSF
 6191 described in the TOE design.

6192 The subsystems described in the TOE design provide a description of how the TSF works at a
 6193 detailed level for SFR-enforcing portions of the TSF, and at a higher level for other portions of the
 6194 TSF. The TSFI provide a description of how the implementation is exercised. The evidence from the
 6195 developer identifies the subsystem that is initially involved when an operation is requested at the
 6196 TSFI, and identify the various subsystems that are primarily responsible for implementing the
 6197 functionality. Note that a complete “call tree” for each TSFI is not required for this work unit.

6198 The evaluator assesses the completeness of the mapping by ensuring that all of the TSFI map to at
 6199 least one subsystem. The verification of accuracy is more complex.

6200 The first aspect of accuracy is that each TSFI is mapped to a subsystem at the TSF boundary. This
 6201 determination can be made by reviewing the subsystem description and interactions, and from this
 6202 information determining its place in the architecture. The next aspect of accuracy is that the
 6203 mapping makes sense. For instance, mapping a TSFI dealing with access control to a subsystem
 6204 that checks passwords is not accurate. The evaluator should again use judgement in making this
 6205 determination. The goal is that this information aids the evaluator in understanding the system and
 6206 implementation of the SFRs, and ways in which entities at the TSF boundary can interact with the
 6207 TSF. The bulk of the assessment of whether the SFRs are described accurately by the subsystems is
 6208 performed in other work units.

6209 **11.8.1.3 Action ADV_TDS.1.2E**

6210 **11.8.1.3.1 Work unit ADV_TDS.1-7**

6211 The evaluator **shall examine** the TOE security functional requirements and the TOE design, to
 6212 determine that all ST security functional requirements are covered by the TOE design.

6213 The evaluator may construct a map between the TOE security functional requirements and the TOE
 6214 design. This map will likely be from a functional requirement to a set of subsystems. Note that this
 6215 map may have to be at a level of detail below the component or even element level of the
 6216 requirements, because of operations (assignments, refinements, selections) performed on the
 6217 functional requirement by the ST author.

6218 For example, the **FDP_ACC.1 Subset access control** component contains an element with
 6219 assignments. If the ST contained, for instance, ten rules in the **FDP_ACC.1 Subset access control**
 6220 assignment, and these ten rules were implemented in specific places within fifteen modules, it
 6221 would be inadequate for the evaluator to map **FDP_ACC.1 Subset access control** to one subsystem
 6222 and claim the work unit had been completed. Instead, the evaluator would map **FDP_ACC.1 Subset**
 6223 **access control** (rule 1) to subsystem A, behaviours x, y, and z; **FDP_ACC.1 Subset access control**
 6224 (rule 2) to subsystem A, behaviours x, p, and q; etc.

6225 **11.8.1.3.2 Work unit ADV_TDS.1-8**

6226 The evaluator *shall examine* the TOE design to determine that it is an accurate instantiation of all
6227 security functional requirements.

6228 The evaluator ensures that each security requirement listed in the TOE security functional
6229 requirements subclause of the ST has a corresponding design description in the TOE design that
6230 accurately details how the TSF meets that requirement. This requires that the evaluator identify a
6231 collection of subsystems that are responsible for implementing a given functional requirement, and
6232 then examine those subsystems to understand how the requirement is implemented. Finally, the
6233 evaluator would assess whether the requirement was accurately implemented.

6234 As an example, if the ST requirements specified a role-based access control mechanism, the
6235 evaluator would first identify the subsystems that contribute to this mechanism's implementation.
6236 This could be done by in-depth knowledge or understanding of the TOE design or by work done in
6237 the previous work unit. Note that this trace is only to identify the subsystems, and is not the
6238 complete analysis.

6239 The next step would be to understand what mechanism the subsystems implemented. For instance,
6240 if the design described an implementation of access control based on UNIX-style protection bits,
6241 the design would not be an accurate instantiation of those access control requirements present in
6242 the ST example used above. If the evaluator could not determine that the mechanism was
6243 accurately implemented because of a lack of detail, the evaluator would have to assess whether all
6244 of the SFR-enforcing subsystems have been identified, or if adequate detail had been provided for
6245 those subsystems.

6246 **11.8.2 Evaluation of sub-activity (ADV_TDS.2)**

6247 **11.8.2.1 Input**

6248 The evaluation evidence for this sub-activity is:

- 6249 a) the ST;
- 6250 b) the functional specification;
- 6251 c) security architecture description;
- 6252 d) the TOE design.

6253 **11.8.2.2 Action ADV_TDS.2.1E**

6254 ISO/IEC 15408-3 ADV_TDS.2.1C: *The design shall describe the structure of the TOE in terms of*
6255 *subsystems.*

6256 **11.8.2.2.1 Work unit ADV_TDS.2-1**

6257 The evaluator *shall examine* the TOE design to determine that the structure of the entire TOE is
6258 described in terms of subsystems.

6259 The evaluator ensures that all of the subsystems of the TOE are identified. This description of the
6260 TOE will be used as input to work unit ADV_TDS.2-2, where the parts of the TOE that make up the
6261 TSF are identified. That is, this requirement is on the entire TOE rather than on only the TSF.

6262 The TOE (and TSF) may be described in multiple layers of abstraction (i.e. subsystems and
6263 modules). Depending upon the complexity of the TOE, its design may be described in terms of
6264 subsystems and modules, as described in ISO/IEC 15408-3 **Annex A.4, ADV_TDS: Subsystems and**
6265 **Modules**. At this level of assurance, the decomposition only need be at the "subsystem" level.

- 6266 In performing this activity, the evaluator examines other evidence presented for the TOE (e.g., ST,
6267 operator user guidance) to determine that the description of the TOE in such evidence is consistent
6268 with the description contained in the TOE design.
- 6269 ISO/IEC 15408-3 ADV_TDS.2.2C: *The design shall identify all subsystems of the TSF.*
- 6270 **11.8.2.2.2 Work unit ADV_TDS.2-2**
- 6271 The evaluator **shall examine** the TOE design to determine that all subsystems of the TSF are
6272 identified.
- 6273 In work unit ADV_TDS.2-1 all of the subsystems of the TOE were identified, and a determination
6274 made that the non-TSF subsystems were correctly characterised. Building on that work, the
6275 subsystems that were not characterised as non-TSF subsystems should be precisely identified. The
6276 evaluator determines that, of the hardware and software installed and configured according to the
6277 Preparative procedures (AGD_PRE) guidance, each subsystem has been accounted for as either one
6278 that is part of the TSF, or one that is not.
- 6279 ISO/IEC 15408-3 ADV_TDS.2.3C: *The design shall describe the behaviour of each SFR non-interfering*
6280 *subsystem of the TSF in detail sufficient to determine that it is SFR non-interfering.*
- 6281 **11.8.2.2.3 Work unit ADV_TDS.2-3**
- 6282 The evaluator **shall examine** the TOE design to determine that each SFR-non-interfering
6283 subsystem of the TSF is described such that the evaluator can determine that the subsystem is SFR-
6284 non-interfering.
- 6285 SFR-non-interfering subsystems do not need to be described in detail as to how they function in
6286 the system. However, the evaluator makes a determination, based on the evidence provided by the
6287 developer, that the subsystems that do not have detailed descriptions are SFR-non-interfering.
6288 Note that if the developer provides a uniform level of detailed documentation then this work unit
6289 will be largely satisfied, since the point of categorising the subsystems is to allow the developer to
6290 provide less information for SFR-non-interfering subsystems than for SFR-enforcing and SFR-
6291 supporting subsystems.
- 6292 An SFR-non-interfering subsystem is one on which the SFR-enforcing and SFR-supporting
6293 subsystems have no dependence; that is, they play no role in implementing SFR functionality.
- 6294 ISO/IEC 15408-3 ADV_TDS.2.4C: *The design shall describe the SFR-enforcing behaviour of the SFR-*
6295 *enforcing subsystems.*
- 6296 **11.8.2.2.4 Work unit ADV_TDS.2-4**
- 6297 The evaluator **shall examine** the TOE design to determine that it provides a complete, accurate,
6298 and detailed description of the SFR-enforcing behaviour of the SFR-enforcing subsystems.
- 6299 The developer may designate subsystems as SFR-enforcing, SFR-supporting, and SFR non-
6300 interfering, but these “tags” are used only to describe the amount and type of information the
6301 developer must provide, and can be used to limit the amount of information the developer has to
6302 develop if their engineering process does not produce the documentation required. Whether the
6303 subsystems have been categorised by the developer or not, it is the evaluator's responsibility to
6304 determine that the subsystems have the appropriate information for their role (SFR-enforcing, etc.)
6305 in the TOE, and to obtain the appropriate information from the developer should the developer fail
6306 to provide the required information for a particular subsystem.
- 6307 SFR-enforcing behaviour refers to *how* a subsystem provides the functionality that implements an
6308 SFR. While not at the level of an algorithmic description, a detailed description of behaviour
6309 typically discusses how the functionality is provided in terms of what key data and data structures

6310 are, what control relationships exist within a subsystem, and how these elements work together to
6311 provide the SFR-enforcing behaviour. Such a description also references SFR-supporting behaviour,
6312 which the evaluator should consider in performing subsequent work units.

6313 To determine completeness and accuracy, the evaluator examines other information available (e.g.,
6314 functional specification, security architecture description). Descriptions of functionality in these
6315 documents should be consistent with what is provided for evidence for this work unit.

6316 ISO/IEC 15408-3 ADV_TDS.2.5C: *The design shall summarise the SFR-supporting and SFR-non-*
6317 *interfering behaviour of the SFR-enforcing subsystems.*

6318 **11.8.2.2.5 Work unit ADV_TDS.2-5**

6319 The evaluator **shall examine** the TOE design to determine that it provides a complete and accurate
6320 high-level summary of the SFR-supporting and SFR-non-interfering behaviour of the SFR-enforcing
6321 subsystems.

6322 The developer may designate subsystems as SFR-enforcing, SFR-supporting, and SFR non-
6323 interfering, but these “tags” are used only to describe the amount and type of information the
6324 developer must provide, and can be used to limit the amount of information the developer has to
6325 develop if their engineering process does not produce the documentation required. Whether the
6326 subsystems have been categorised by the developer or not, it is the evaluator's responsibility to
6327 determine that the subsystems have the appropriate information for their role (SFR-enforcing, etc.)
6328 in the TOE, and to obtain the appropriate information from the developer should the developer fail
6329 to provide the required information for a particular subsystem.

6330 In contrast to the previous work unit, this work unit calls for the evaluator to assess the
6331 information provided for SFR-enforcing subsystems that is SFR-supporting or SFR-non-interfering.
6332 The goal of this assessment is two-fold. First, it should provide the evaluator greater understanding
6333 of the way each subsystem works. Second, this assessment will help the evaluator to determine
6334 that all SFR-enforcing behaviour exhibited by a SFR-enforcing subsystem has been described.
6335 Unlike the previous work unit, the information provided for the SFR-supporting or SFR-non-
6336 interfering behaviour does not have to be as detailed as that provided by the SFR-enforcing
6337 behaviour. For example, data structures or data items that do not pertain to SFR-enforcing
6338 functionality will likely not need to be described in detail, if at all. It is the evaluator's
6339 determination, however, with respect to what “high-level” means for a particular TOE, and the
6340 evaluator obtains enough information from the developer (even if it turns out to be equivalent to
6341 information provided for the parts of the subsystem that are SFR-enforcing) to make a sound
6342 verdict for this work unit.

6343 The evaluator is cautioned, however, that “perfect” assurance is not a goal nor required by this
6344 work unit, so judgement will have to be exercised in determine the amount and composition of the
6345 evidence required to make a verdict on this work unit.

6346 To determine completeness and accuracy, the evaluator examines other information available (e.g.,
6347 functional specification, security architecture description). Summaries of functionality in these
6348 documents should be consistent with what is provided for evidence for this work unit. In particular,
6349 the functional specification should be used to determine that the behaviour required to implement
6350 the TSF Interfaces described by the functional specification are completely described by the
6351 subsystem, since the behaviour will either be SFR-enforcing, SFR-supporting or SFR-non-
6352 interfering.

6353 ISO/IEC 15408-3 ADV_TDS.2.6C: *The design shall summarise the behaviour of the SFR-supporting*
6354 *subsystems.*

6355 **11.8.2.2.6 Work unit ADV_TDS.2-6**

6356 The evaluator *shall examine* the TOE design to determine that it provides a complete and accurate
6357 high-level summary of the behaviour of the SFR-supporting subsystems.

6358 The developer may designate subsystems as SFR-enforcing, SFR-supporting, and SFR non-
6359 interfering, but these “tags” are used only to describe the amount and type of information the
6360 developer must provide, and can be used to limit the amount of information the developer has to
6361 develop if their engineering process does not produce the documentation required. Whether the
6362 subsystems have been categorised by the developer or not, it is the evaluator’s responsibility to
6363 determine that the subsystems have the appropriate information for their role (SFR-enforcing, etc.)
6364 in the TOE, and to obtain the appropriate information from the developer should the developer fail
6365 to provide the required information for a particular subsystem.

6366 In contrast to the previous two work units, this work unit calls for the developer to provide (and
6367 the evaluator to assess) information about SFR supporting subsystems. Such subsystems should be
6368 referenced by the descriptions of the SFR-enforcing subsystems, as well as by the descriptions of
6369 interactions in work unit ADV_TDS.2-7. The goal of evaluator’s assessment, like that for the
6370 previous work unit, is two-fold. First, it should provide the evaluator with an understanding of the
6371 way each SFR-supporting subsystem works. Second, the evaluator determines that the behaviour is
6372 summarized in enough detail so that the way in which the subsystem supports the SFR-enforcing
6373 behaviour is clear, and that the behaviour is not itself SFR-enforcing. The information provided for
6374 SFR-supporting subsystem’s behaviour does not have to be as detailed as that provided by the SFR-
6375 enforcing behaviour. For example, data structures or data items that do not pertain to SFR-
6376 enforcing functionality will likely not need to be described in detail, if at all. It is the evaluator’s
6377 determination, however, with respect to what “high-level” means for a particular TOE, and the
6378 evaluator obtains enough information from the developer (even if it turns out to be equivalent to
6379 information provided for the parts of the subsystem that are SFR-enforcing) to make a sound
6380 verdict for this work unit.

6381 The evaluator is cautioned, however, that “perfect” assurance is not a goal nor required by this
6382 work unit, so judgement will have to be exercised in determine the amount and composition of the
6383 evidence required to make a verdict on this work unit.

6384 To determine completeness and accuracy, the evaluator examines other information available (e.g.,
6385 functional specification, security architecture description). Summaries of functionality in these
6386 documents should be consistent with what is provided for evidence for this work unit.

6387 ISO/IEC 15408-3 ADV_TDS.2.7C: *The design shall provide a description of the interactions among all*
6388 *subsystems of the TSF.*

6389 **11.8.2.2.7 Work unit ADV_TDS.2-7**

6390 The evaluator *shall examine* the TOE design to determine that interactions between the
6391 subsystems of the TSF are described.

6392 The goal of describing the interactions between the subsystems is to help provide the reader a
6393 better understanding of how the TSF performs its functions. These interactions do not need to be
6394 characterised at the implementation level (e.g., parameters passed from one routine in a subsystem
6395 to a routine in a different subsystem; global variables; hardware signals (e.g., interrupts) from a
6396 hardware subsystem to an interrupt-handling subsystem), but the data elements identified for a
6397 particular subsystem that are going to be used by another subsystem need to be covered in this
6398 discussion. Any control relationships between subsystems (e.g., a subsystem responsible for
6399 configuring a rule base for a firewall system and the subsystem that actually implements these
6400 rules) should also be described.

6401 It should be noted while the developer should characterise all interactions between subsystems,
6402 the evaluators need to use their own judgement in assessing the completeness of the description. If

6403 the reason for an interaction is unclear, or if there are SFR-related interactions (discovered, for
6404 instance, in examining the descriptions of subsystem behaviour) that do not appear to be described,
6405 the evaluator ensures that this information is provided by the developer. However, if the evaluator
6406 can determine that interactions among a particular set of subsystems, while incompletely
6407 described by the developer, will not aid in understanding the overall functionality nor security
6408 functionality provided by the TSF, then the evaluator may choose to consider the description
6409 sufficient, and not pursue completeness for its own sake.

6410 ISO/IEC 15408-3 ADV_TDS.2.8C: *The mapping shall demonstrate that all TSFIs trace to the*
6411 *behaviour described in the TOE design that they invoke.*

6412 **11.8.2.2.8 Work unit ADV_TDS.2-8**

6413 The evaluator **shall examine** the TOE design to determine that it contains a complete and accurate
6414 mapping from the TSFI described in the functional specification to the subsystems of the TSF
6415 described in the TOE design.

6416 The subsystems described in the TOE design provide a description of how the TSF works at a
6417 detailed level for SFR-enforcing portions of the TSF, and at a higher level for other portions of the
6418 TSF. The TSFI provide a description of how the implementation is exercised. The evidence from the
6419 developer identifies the subsystem that is initially involved when an operation is requested at the
6420 TSFI, and identify the various subsystems that are primarily responsible for implementing the
6421 functionality. Note that a complete “call tree” for each TSFI is not required for this work unit.

6422 The evaluator assesses the completeness of the mapping by ensuring that all of the TSFI map to at
6423 least one subsystem. The verification of accuracy is more complex.

6424 The first aspect of accuracy is that each TSFI is mapped to a subsystem at the TSF boundary. This
6425 determination can be made by reviewing the subsystem description and interactions, and from this
6426 information determining its place in the architecture. The next aspect of accuracy is that the
6427 mapping makes sense. For instance, mapping a TSFI dealing with access control to a subsystem
6428 that checks passwords is not accurate. The evaluator should again use judgement in making this
6429 determination. The goal is that this information aids the evaluator in understanding the system and
6430 implementation of the SFRs, and ways in which entities at the TSF boundary can interact with the
6431 TSF. The bulk of the assessment of whether the SFRs are described accurately by the subsystems is
6432 performed in other work units.

6433 **11.8.2.3 Action ADV_TDS.2.2E**

6434 **11.8.2.3.1 Work unit ADV_TDS.2-9**

6435 The evaluator **shall examine** the TOE security functional requirements and the TOE design, to
6436 determine that all ST security functional requirements are covered by the TOE design.

6437 The evaluator may construct a map between the TOE security functional requirements and the TOE
6438 design. This map will likely be from a functional requirement to a set of subsystems. Note that this
6439 map may have to be at a level of detail below the component or even element level of the
6440 requirements, because of operations (assignments, refinements, selections) performed on the
6441 functional requirement by the ST author.

6442 For example, the **FDP_ACC.1 Subset access control** component contains an element with
6443 assignments. If the ST contained, for instance, ten rules in the **FDP_ACC.1 Subset access control**
6444 assignment, and these ten rules were implemented in specific places within fifteen modules, it
6445 would be inadequate for the evaluator to map **FDP_ACC.1 Subset access control** to one subsystem
6446 and claim the work unit had been completed. Instead, the evaluator would map **FDP_ACC.1 Subset**
6447 **access control** (rule 1) to subsystem A, behaviours x, y, and z; **FDP_ACC.1 Subset access control**
6448 (rule 2) to subsystem A, behaviours x, p, and q; etc.

6449 **11.8.2.3.2 Work unit ADV_TDS.2-10**

6450 The evaluator *shall examine* the TOE design to determine that it is an accurate instantiation of all
6451 security functional requirements.

6452 The evaluator ensures that each security requirement listed in the TOE security functional
6453 requirements subclause of the ST has a corresponding design description in the TOE design that
6454 accurately details how the TSF meets that requirement. This requires that the evaluator identify a
6455 collection of subsystems that are responsible for implementing a given functional requirement, and
6456 then examine those subsystems to understand how the requirement is implemented. Finally, the
6457 evaluator would assess whether the requirement was accurately implemented.

6458 As an example, if the ST requirements specified a role-based access control mechanism, the
6459 evaluator would first identify the subsystems that contribute to this mechanism's implementation.
6460 This could be done by in-depth knowledge or understanding of the TOE design or by work done in
6461 the previous work unit. Note that this trace is only to identify the subsystems, and is not the
6462 complete analysis.

6463 The next step would be to understand what mechanism the subsystems implemented. For instance,
6464 if the design described an implementation of access control based on UNIX-style protection bits,
6465 the design would not be an accurate instantiation of those access control requirements present in
6466 the ST example used above. If the evaluator could not determine that the mechanism was
6467 accurately implemented because of a lack of detail, the evaluator would have to assess whether all
6468 of the SFR-enforcing subsystems have been identified, or if adequate detail had been provided for
6469 those subsystems.

6470 **11.8.3 Evaluation of sub-activity (ADV_TDS.3)**6471 **11.8.3.1 Objectives**

6472 The objective of this sub-activity is to determine whether the TOE design provides a description of
6473 the TOE in terms of subsystems sufficient to determine the TSF boundary, and provides a
6474 description of the TSF internals in terms of modules (and optionally higher-level abstractions). It
6475 provides a detailed description of the SFR-enforcing modules and enough information about the
6476 SFR-supporting and SFR-non-interfering modules for the evaluator to determine that the SFRs are
6477 completely and accurately implemented; as such, the TOE design provides an explanation of the
6478 implementation representation.

6479 **11.8.3.2 Input**

6480 The evaluation evidence for this sub-activity is:

- 6481 a) the ST;
- 6482 b) the functional specification;
- 6483 c) security architecture description;
- 6484 d) the TOE design.

6485 **11.8.3.3 Application notes**

6486 There are three types of activity that the evaluator must undertake with respect to the TOE design.
6487 First, the evaluator determines that the TSF boundary has been adequately described. Second, the
6488 evaluator determines that the developer has provided documentation that conforms to the content
6489 and presentation requirements for this subsystem, and that is consistent with other documentation
6490 provided for the TOE. Finally, the evaluator must analyse the design information provided for the
6491 SFR-enforcing modules (at a detailed level) and the SFR-supporting and SFR-non-interfering

6492 modules (at a less detailed level) to understand how the system is implemented, and with that
6493 knowledge ensure that the TSFI in the functional specification are adequately described, and that
6494 the test information adequately tests the TSF (done in the Class ATE: Tests work units).

6495 It is important to note that while the developer is obligated to provide a complete description of
6496 the TSF (although SFR-enforcing modules will have more detail than the SFR-supporting or SFR-
6497 non-interfering modules), the evaluator is expected to use their judgement in performing their
6498 analysis. While the evaluator is expected to look at every module, the detail to which they examine
6499 each module may vary. The evaluator analyses each module in order to gain enough understanding
6500 to determine the effect of the functionality of the module on the security of the system, and the
6501 depth to which they need to analyse the module may vary depending on the module's role in the
6502 system. An important aspect of this analysis is that the evaluator should use the other
6503 documentation provided (TSS, functional specification, security architecture description, and the
6504 TSF internal document) in order to determine that the functionality that is described is correct, and
6505 that the implicit designation of SFR-supporting or SFR-non-interfering modules (see below) is
6506 supported by their role in the system architecture.

6507 The developer may designate modules as SFR-enforcing, SFR-supporting, and SFR non-interfering,
6508 but these “tags” are used only to describe the amount and type of information the developer must
6509 provide, and can be used to limit the amount of information the developer has to develop if their
6510 engineering process does not produce the documentation required. Whether the modules have
6511 been categorised by the developer or not, it is the evaluator's responsibility to determine that the
6512 modules have the appropriate information for their role (SFR-enforcing, etc.) in the TOE, and to
6513 obtain the appropriate information from the developer should the developer fail to provide the
6514 required information for a particular module.

6515 **11.8.3.4 Action ADV_TDS.3.1E**

6516 ISO/IEC 15408-3 ADV_TDS.3.1C: *The design shall describe the structure of the TOE in terms of*
6517 *subsystems.*

6518 **11.8.3.4.1 Work unit ADV_TDS.3-1**

6519 The evaluator **shall examine** the TOE design to determine that the structure of the entire TOE is
6520 described in terms of subsystems.

6521 The evaluator ensures that all of the subsystems of the TOE are identified. This description of the
6522 TOE will be used as input to work unit ADV_TDS.3-2, where the parts of the TOE that make up the
6523 TSF are identified. That is, this requirement is on the entire TOE rather than on only the TSF.

6524 The TOE (and TSF) may be described in multiple layers of abstraction (i.e. subsystems and
6525 modules). Depending upon the complexity of the TOE, its design may be described in terms of
6526 subsystems and modules, as described in ISO/IEC 15408-3 **Annex A.4, ADV_TDS: Subsystems and**
6527 **Modules**. For a very simple TOE that can be described solely at the “module” level (see ADV_TDS.3-
6528 2), this work unit is not applicable and therefore considered to be satisfied.

6529 In performing this activity, the evaluator examines other evidence presented for the TOE (e.g., ST,
6530 operator user guidance) to determine that the description of the TOE in such evidence is consistent
6531 with the description contained in the TOE design.

6532 ISO/IEC 15408-3 ADV_TDS.3.2C: *The design shall describe the TSF in terms of modules.*

6533 **11.8.3.4.2 Work unit ADV_TDS.3-2**

6534 The evaluator **shall examine** the TOE design to determine that the entire TSF is described in terms
6535 of modules.

6536 The evaluator will examine the modules for specific properties in other work units; in this work
 6537 unit the evaluator determines that the modular description covers the entire TSF, and not just a
 6538 portion of the TSF. The evaluator uses other evidence provided for the evaluation (e.g., functional
 6539 specification, security architecture description) in making this determination. For example, if the
 6540 functional specification contains interfaces to functionality that does not appear to be described in
 6541 the TOE design description, it may be the case that a portion of the TSF has not been included
 6542 appropriately. Making this determination will likely be an iterative process, where as more analysis
 6543 is done on the other evidence, more confidence can be gained with respect to the completeness of
 6544 the documentation.

6545 Unlike subsystems, modules describe the implementation in a level of detail that can serve as a
 6546 guide to reviewing the implementation representation. A description of a module should be such
 6547 that one could create an implementation of the module from the description, and the resulting
 6548 implementation would be 1) identical to the actual TSF implementation in terms of the interfaces
 6549 presented, 2) identical in the use of interfaces that are mentioned in the design, and 3) functionally
 6550 equivalent to the description of the purpose of the TSF module. For instance, RFC 793 provides a
 6551 high-level description of the TCP protocol. It is necessarily implementation independent. While it
 6552 provides a wealth of detail, it is *not* a suitable design description because it is not specific to an
 6553 implementation. An actual implementation can add to the protocol specified in the RFC, and
 6554 implementation choices (for instance, the use of global data vs. local data in various parts of the
 6555 implementation) may have an impact on the analysis that is performed. The design description of
 6556 the TCP module would list the interfaces presented by the implementation (rather than just those
 6557 defined in RFC 793), as well as an algorithm description of the processing associated with the
 6558 modules implementing TCP (assuming it was part of the TSF).

6559 ISO/IEC 15408-3 ADV_TDS.3.3C: *The design shall identify all subsystems of the TSF.*

6560 **11.8.3.4.3 Work unit ADV_TDS.3-3**

6561 The evaluator *shall examine* the TOE design to determine that all subsystems of the TSF are
 6562 identified.

6563 If the design is presented solely in terms of modules, then subsystems in these requirements are
 6564 equivalent to modules and the activity should be performed at the module level.

6565 In work unit ADV_TDS.3-1 all of the subsystems of the TOE were identified, and a determination
 6566 made that the non-TSF subsystems were correctly characterised. Building on that work, the
 6567 subsystems that were not characterised as non-TSF subsystems should be precisely identified. The
 6568 evaluator determines that, of the hardware and software installed and configured according to the
 6569 Preparative procedures (AGD_PRE) guidance, each subsystem has been accounted for as either one
 6570 that is part of the TSF, or one that is not.

6571 ISO/IEC 15408-3 ADV_TDS.3.4C: *The design shall provide a description of each subsystem of the TSF.*

6572 **11.8.3.4.4 Work unit ADV_TDS.3-4**

6573 The evaluator *shall examine* the TOE design to determine that each subsystem of the TSF
 6574 describes its role in the enforcement of SFRs described in the ST.

6575 If the design is presented solely in terms of modules, then this work unit will be considered
 6576 satisfied by the assessment done in subsequent work units; no explicit action on the part of the
 6577 evaluator is necessary in this case.

6578 On systems that are complex enough to warrant a subsystem-level description of the TSF in
 6579 addition to the modular description, the goal of the subsystem-level description is to give the
 6580 evaluator context for the modular description that follows. Therefore, the evaluator ensures that
 6581 the subsystem-level description contains a description of how the security functional requirements
 6582 are achieved in the design, but at a level of abstraction above the modular description. This

6583 description should discuss the mechanisms used at a level that is aligned with the module
6584 description; this will provide the evaluators the road map needed to intelligently assess the
6585 information contained in the module description. A well-written set of subsystem descriptions will
6586 help guide the evaluator in determining the modules that are most important to examine, thus
6587 focusing the evaluation activity on the portions of the TSF that have the most relevance with
6588 respect to the enforcement of the SFRs.

6589 The evaluator ensures that all subsystems of the TSF have a description. While the description
6590 should focus on the role that the subsystem plays in enforcing or supporting the implementation of
6591 the SFRs, enough information must be present so that a context for understanding the SFR-related
6592 functionality is provided.

6593 **11.8.3.4.5 Work unit ADV_TDS.3-5**

6594 The evaluator *shall examine* the TOE design to determine that each SFR-non-interfering
6595 subsystem of the TSF is described such that the evaluator can determine that the subsystem is SFR-
6596 non-interfering.

6597 If the design is presented solely in terms of modules, then this work unit will be considered
6598 satisfied by the assessment done in subsequent work units; no explicit action on the part of the
6599 evaluator is necessary in this case.

6600 An SFR-non-interfering subsystem is one on which the SFR-enforcing and SFR-supporting
6601 subsystems have no dependence; that is, they play no role in implementing SFR functionality.

6602 The evaluator ensures that all subsystems of the TSF have a description. While the description
6603 should focus on the role that the subsystem do not plays in enforcing or supporting the
6604 implementation of the SFRs, enough information must be present so that a context for
6605 understanding the SFR-non-interfering functionality is provided.

6606 ISO/IEC 15408-3 ADV_TDS.3.5C: *The design shall provide a description of the interactions among all*
6607 *subsystems of the TSF.*

6608 **11.8.3.4.6 Work unit ADV_TDS.3-6**

6609 The evaluator *shall examine* the TOE design to determine that interactions between the
6610 subsystems of the TSF are described.

6611 If the design is presented solely in terms of modules, then this work unit will be considered
6612 satisfied by the assessment done in subsequent work units; no explicit action on the part of the
6613 evaluator is necessary in this case.

6614 On systems that are complex enough to warrant a subsystem-level description of the TSF in
6615 addition to the modular description, the goal of describing the interactions between the
6616 subsystems is to help provide the reader a better understanding of how the TSF performs its
6617 functions. These interactions do not need to be characterised at the implementation level (e.g.,
6618 parameters passed from one routine in a subsystem to a routine in a different subsystem; global
6619 variables; hardware signals (e.g., interrupts) from a hardware subsystem to an interrupt-handling
6620 subsystem), but the data elements identified for a particular subsystem that are going to be used by
6621 another subsystem should be covered in this discussion. Any control relationships between
6622 subsystems (e.g., a subsystem responsible for configuring a rule base for a firewall system and the
6623 subsystem that actually implements these rules) should also be described.

6624 It should be noted while the developer should characterise all interactions between subsystems,
6625 the evaluators need to use their own judgement in assessing the completeness of the description. If
6626 the reason for an interaction is unclear, or if there are SFR-related interactions (discovered, for
6627 instance, in examining the module-level documentation) that do not appear to be described, the
6628 evaluator ensures that this information is provided by the developer. However, if the evaluator can

6629 determine that interactions among a particular set of subsystems, while incompletely described by
 6630 the developer, and a complete description will not aid in understanding the overall functionality
 6631 nor security functionality provided by the TSF, then the evaluator may choose to consider the
 6632 description sufficient, and not pursue completeness for its own sake.

6633 ISO/IEC 15408-3 ADV_TDS.3.6C: *The design shall provide a mapping from the subsystems of the TSF*
 6634 *to the modules of the TSF.*

6635 **11.8.3.4.7 Work unit ADV_TDS.3-7**

6636 The evaluator **shall examine** the TOE design to determine that the mapping between the
 6637 subsystems of the TSF and the modules of the TSF is complete.

6638 If the design is presented solely in terms of modules, then this work unit is considered satisfied.

6639 For TOEs that are complex enough to warrant a subsystem-level description of the TSF in addition
 6640 to the modular description, the developer provides a simple mapping showing how the modules of
 6641 the TSF are allocated to the subsystems. This will provide the evaluator a guide in performing their
 6642 module-level assessment. To determine completeness, the evaluator examines each mapping and
 6643 determines that all subsystems map to at least one module, and that all modules map to exactly one
 6644 subsystem.

6645 **11.8.3.4.8 Work unit ADV_TDS.3-8**

6646 The evaluator **shall examine** the TOE design to determine that the mapping between the
 6647 subsystems of the TSF and the modules of the TSF is accurate.

6648 If the design is presented solely in terms of modules, then this work unit is considered satisfied.

6649 For TOEs that are complex enough to warrant a subsystem-level description of the TSF in addition
 6650 to the modular description, the developer provides a simple mapping showing how the modules of
 6651 the TSF are allocated to the subsystems. This will provide the evaluator a guide in performing their
 6652 module-level assessment. The evaluator may choose to check the accuracy of the mapping in
 6653 conjunction with performing other work units. An “inaccurate” mapping is one where the module
 6654 is mistakenly associated with a subsystem where its functions are not used within the subsystem.
 6655 Because the mapping is intended to be a guide supporting more detailed analysis, the evaluator is
 6656 cautioned to apply appropriate effort to this work unit. Expending extensive evaluator resources
 6657 verifying the accuracy of the mapping is not necessary. Inaccuracies that lead to mis-
 6658 understandings related to the design that are uncovered as part of this or other work units are the
 6659 ones that should be associated with this work unit and corrected.

6660 ISO/IEC 15408-3 ADV_TDS.3.7C: *The design shall describe each SFR-enforcing module in terms of its*
 6661 *purpose and relationship with other modules.*

6662 **11.8.3.4.9 Work unit ADV_TDS.3-9**

6663 The evaluator **shall examine** the TOE design to determine that the description of the purpose of
 6664 each SFR-enforcing module and relationship with other modules is complete and accurate.

6665 The developer may designate modules as SFR-enforcing, SFR-supporting, and SFR non-interfering,
 6666 but these “tags” are used only to describe the amount and type of information the developer must
 6667 provide, and can be used to limit the amount of information the developer has to develop if their
 6668 engineering process does not produce the documentation required. Whether the modules have
 6669 been categorised by the developer or not, it is the evaluator's responsibility to determine that the
 6670 modules have the appropriate information for their role (SFR-enforcing, etc.) in the TOE, and to
 6671 obtain the appropriate information from the developer should the developer fail to provide the
 6672 required information for a particular module.

6673 The purpose of a module provides a description indicating what function the module is fulfilling. A
 6674 word of caution to evaluator is in order. The focus of this work unit should be to provide the
 6675 evaluator an understanding of how the module works so that determinations can be made about
 6676 the soundness of the implementation of the SFRs, as well as to support architectural analysis
 6677 performed for ADV_ARC component. As long as the evaluator has a sound understanding of the
 6678 module's operation, and its relationship to other modules and the TOE as a whole, the evaluator
 6679 should consider the objective of the work achieved and not engage in a documentation exercise for
 6680 the developer (by requiring, for example, a complete algorithmic description for a self-evident
 6681 implementation representation).

6682 Because the modules are at such a low level, it may be difficult determine completeness and
 6683 accuracy impacts from other documentation, such as operational user guidance, the functional
 6684 specification, the TSF internals, or the security architecture description. However, the evaluator
 6685 uses the information present in those documents to the extent possible to help ensure that the
 6686 purpose is accurately and completely described. This analysis can be aided by the analysis
 6687 performed for the work units for the **ADV_TDS.3.10C** element, which maps the TSFI in the
 6688 functional specification to the modules of the TSF.

6689 ISO/IEC 15408-3 ADV_TDS.3.8C: *The design shall describe each SFR-enforcing module in terms of its*
 6690 *SFR-related interfaces, return values from those interfaces, interaction with other modules and called*
 6691 *SFR-related interfaces to other SFR-enforcing modules.*

6692 **11.8.3.4.10 Work unit ADV_TDS.3-10**

6693 The evaluator **shall examine** the TOE design to determine that the description of the interfaces
 6694 presented by each SFR-enforcing module contain an accurate and complete description of the SFR-
 6695 related parameters, the calling conventions for each interface, and any values returned directly by
 6696 the interface.

6697 The SFR-related interfaces of a module are those interfaces used by other modules as a means to
 6698 invoke the SFR-related operations provided, and to provide inputs to or receive outputs from the
 6699 module. The purpose in the specification of these interfaces is to permit the exercise of them
 6700 during testing. Inter-module interfaces that are not SFR-related need not be specified or described,
 6701 since they are not a factor in testing. Likewise, other internal interfaces that are not a factor in
 6702 traversing SFR-related paths of execution (such as those internal paths that are fixed) need not be
 6703 specified or described, since they are not a factor in testing.

6704 SFR-related interfaces are described in terms of how they are invoked, and any values that are
 6705 returned. This description would include a list of SFR-related parameters, and descriptions of these
 6706 parameters. Note that global data would also be considered parameters if used by the module
 6707 (either as inputs or outputs) when invoked. If a parameter were expected to take on a set of values
 6708 (e.g., a “flag” parameter), the complete set of values the parameter could take on that would have
 6709 an effect on module processing would be specified. Likewise, parameters representing data
 6710 structures are described such that each field of the data structure is identified and described. Note
 6711 that different programming languages may have additional “interfaces” that would be non-obvious;
 6712 an example would be operator/function overloading in C++. This “implicit interface” in the class
 6713 description would also be described as part of the low-level TOE design. Note that although a
 6714 module could present only one interface, it is more common that a module presents a small set of
 6715 related interfaces.

6716 In terms of the assessment of parameters (inputs and outputs) to a module, any use of global data
 6717 must also be considered. A module “uses” global data if it either reads or writes the data. In order
 6718 to assure the description of such parameters (if used) is complete, the evaluator uses other
 6719 information provided about the module in the TOE design (interfaces, algorithmic description, etc.),
 6720 as well as the description of the particular set of global data assessed in work unit ADV_TDS.3-10.
 6721 For instance, the evaluator could first determine the processing the module performs by examining
 6722 its function and interfaces presented (particularly the parameters of the interfaces). They could
 6723 then check to see if the processing appears to “touch” any of the global data areas identified in the

6724 TOE design. The evaluator then determines that, for each global data area that appears to be
 6725 “touched”, that global data area is listed as a means of input or output by the module the evaluator
 6726 is examining.

6727 Invocation conventions are a programming-reference-type description that one could use to
 6728 correctly invoke a module’s interface if one were writing a program to make use of the module’s
 6729 functionality through that interface. This includes necessary inputs and outputs, including any set-
 6730 up that may need to be performed with respect to global variables.

6731 Values returned through the interface refer to values that are either passed through parameters or
 6732 messages; values that the function call itself returns in the style of a “C” program function call; or
 6733 values passed through global means (such as certain error routines in *ix-style operating systems).

6734 In order to assure the description is complete, the evaluator uses other information provided about
 6735 the module in the TOE design (e.g., algorithmic description, global data used) to ensure that it
 6736 appears all data necessary for performing the functions of the module is presented to the module,
 6737 and that any values that other modules expect the module under examination to provide are
 6738 identified as being returned by the module. The evaluator determines accuracy by ensuring that
 6739 the description of the processing matches the information listed as being passed to or from an
 6740 interface.

6741 ISO/IEC 15408-3 ADV_TDS.3.9C: *The design shall describe each SFR-supporting or SFR-non-*
 6742 *interfering module in terms of its purpose and interaction with other modules.*

6743 **11.8.3.4.11 Work unit ADV_TDS.3-11**

6744 The evaluator ***shall examine*** the TOE design to determine that SFR-supporting and SFR-non-
 6745 interfering modules are correctly categorised.

6746 In the cases where the developer has provided different amounts of information for different
 6747 modules, an implicit categorisation has been done. That is, modules (for instance) with detail
 6748 presented on their SFR-related interfaces (see [ADV_TDS.3.10C](#)) are candidate SFR-enforcing
 6749 modules, although examination by the evaluator may lead to a determination that some set of them
 6750 are SFR-supporting or SFR-non-interfering. Those with only a description of their purpose and
 6751 interaction with other modules (for instance) are “implicitly categorised” as SFR-supporting or
 6752 SFR-non-interfering.

6753 In these cases, a key focus of the evaluator for this work unit is attempting to determine from the
 6754 evidence provided for each module implicitly categorised as SFR-supporting or SFR-non-
 6755 interfering and the evaluation information about other modules (in the TOE design, the functional
 6756 specification, the security architecture description, and the operational user guidance), whether
 6757 the module is indeed SFR-supporting or SFR-non-interfering. At this level of assurance some error
 6758 should be tolerated; the evaluator does not have to be absolutely sure that a given module is SFR-
 6759 supporting or SFR-non-interfering, even though it is labelled as such. However, if the evidence
 6760 provided indicates that a SFR-supporting or SFR-non-interfering module is SFR-enforcing, the
 6761 evaluator requests additional information from the developer in order to resolve the apparent
 6762 inconsistency. For instance, suppose the documentation for Module A (an SFR-enforcing module)
 6763 indicates that it calls Module B to perform an access check on a certain type of construct. When the
 6764 evaluator examines the information associated with Module B, they find that all the developer has
 6765 provided is a purpose and a set of interactions (thus implicitly categorising Module B as SFR-
 6766 supporting or SFR-non-interfering). On examining the purpose and interactions from Module A, the
 6767 evaluator finds no mention of Module B performing any access checks, and Module A is not listed
 6768 as a module with which Module B interacts. At this point the evaluator should approach the
 6769 developer to resolve the discrepancies between the information provided in Module A and that in
 6770 Module B.

6771 Another example would be where the evaluator examines the mapping of the TSFI to the modules
 6772 as provided by [ADV_TDS.3.2D](#). This examination shows that Module C is associated with an SFR

6773 requiring identification of the user. Again, when the evaluator examines the information associated
6774 with Module C, they find that all the developer has provided is a purpose and a set of interactions
6775 (thus implicitly categorising Module C as SFR-supporting or SFR-non-interfering). Examining the
6776 purpose and interactions presented for Module C, the evaluator is unable to determine why Module
6777 C, listed as mapping to a TSFI concerned with user identification, would not be classified as SFR-
6778 enforcing. Again, the evaluator should approach the developer to resolve this discrepancy.

6779 A final example is from the opposite point of view. As before, the developer has provided
6780 information associated with Module D consisting of a purpose and a set of interactions (thus
6781 implicitly categorising Module D as SFR-supporting or SFR-non-interfering). The evaluator
6782 examines all of the evidence provided, including the purpose and interactions for Module D. The
6783 purpose appears to give a meaningful description of Module D's function in the TOE, the
6784 interactions are consistent with that description, and there is nothing to indicate that Module D is
6785 SFR-enforcing. In this case, the evaluator should not demand more information about Module D
6786 "just be to sure" it is correctly categorised. The developer has met their obligations and the
6787 resulting assurance the evaluator has in the implicit categorisation of Module D is (by definition)
6788 appropriate for this assurance level.

6789 **11.8.3.4.12 Work unit ADV_TDS.3-12**

6790 The evaluator *shall examine* the TOE design to determine that the description of the purpose of
6791 each SFR-supporting or SFR-non-interfering module is complete and accurate.

6792 The description of the purpose of a module indicates what function the module is fulfilling. From
6793 the description, the evaluator should be able to obtain a general idea of the module's role. In order
6794 to assure the description is complete, the evaluator uses the information provided about the
6795 module's interactions with other modules to assess whether the reasons for the module being
6796 called are consistent with the module's purpose. If the interaction description contains
6797 functionality that is not apparent from, or in conflict with, the module's purpose, the evaluator
6798 needs to determine whether the problem is one of accuracy or of completeness. The evaluator
6799 should be wary of purposes that are too short, since meaningful analysis based on a one-sentence
6800 purpose is likely to be impossible.

6801 Because the modules are at such a low level, it may be difficult determine completeness and
6802 accuracy impacts from other documentation, such as administrative guidance, the functional
6803 specification, the security architecture description, or the TSF internals document. However, the
6804 evaluator uses the information present in those documents to the extent possible to help ensure
6805 that the function is accurately and completely described. This analysis can be aided by the analysis
6806 performed for the work units for the ADV_TDS.3.10C element, which maps the TSFI in the
6807 functional specification to the modules of the TSF.

6808 **11.8.3.4.13 Work unit ADV_TDS.3-13**

6809 The evaluator *shall examine* the TOE design to determine that the description of a SFR-supporting
6810 or SFR-non-interfering module's interaction with other modules is complete and accurate.

6811 It is important to note that, in terms of the Part 3 requirement and this work unit, the term
6812 *interaction* is intended to convey less rigour than *interface*. An interaction does not need to be
6813 characterised at the implementation level (e.g., parameters passed from one routine in a module to
6814 a routine in a different module; global variables; hardware signals (e.g., interrupts) from a
6815 hardware subsystem to an interrupt-handling subsystem), but the data elements identified for a
6816 particular module that are going to be used by another module should be covered in this discussion.
6817 Any control relationships between modules (e.g., a module responsible for configuring a rule base
6818 for a firewall system and the module that actually implements these rules) should also be
6819 described.

6820 Because the modules are at such a low level, it may be difficult determine completeness and
6821 accuracy impacts from other documentation, such as operational user guidance, the functional

6822 specification, the security architecture description, or the TSF internals document. However, the
 6823 evaluator uses the information present in those documents to the extent possible to help ensure
 6824 that the function is accurately and completely described. This analysis can be aided by the analysis
 6825 performed for the work units for the **ADV_TDS.3.10C** element, which maps the TSFI in the
 6826 functional specification to the modules of the TSF.

6827 A module's interaction with other modules goes beyond just a call-tree-type document. The
 6828 interaction is described from a functional perspective of why a module interacts with other
 6829 modules. The module's purpose describes what functions the module provides to other modules;
 6830 the interactions should describe what the module depends on from other modules in order to
 6831 accomplish this function.

6832 ISO/IEC 15408-3 ADV_TDS.3.10C: *The mapping shall demonstrate that all TSFIs trace to the*
 6833 *behaviour described in the TOE design that they invoke.*

6834 **11.8.3.4.14 Work unit ADV_TDS.3-14**

6835 The evaluator **shall examine** the TOE design to determine that it contains a complete and accurate
 6836 mapping from the TSFI described in the functional specification to the modules of the TSF
 6837 described in the TOE design.

6838 The modules described in the TOE design provide a description of the implementation of the TSF.
 6839 The TSFI provide a description of how the implementation is exercised. The evidence from the
 6840 developer identifies the module that is initially invoked when an operation is requested at the TSFI,
 6841 and identifies the chain of modules invoked up to the module that is primarily responsible for
 6842 implementing the functionality. However, a complete call tree for each TSFI is not required for this
 6843 work unit. The cases in which more than one module would have to be identified are where there
 6844 are "entry point" modules or wrapper modules that have no functionality other than conditioning
 6845 inputs or de-multiplexing an input. Mapping to one of these modules would not provide any useful
 6846 information to the evaluator.

6847 The evaluator assesses the completeness of the mapping by ensuring that all of the TSFI map to at
 6848 least one module. The verification of accuracy is more complex.

6849 The first aspect of accuracy is that each TSFI is mapped to a module at the TSF boundary. This
 6850 determination can be made by reviewing the module description and its interfaces/interactions.
 6851 The next aspect of accuracy is that each TSFI identifies a chain of modules between the initial
 6852 module identified and a module that is primarily responsible for implementing the function
 6853 presented at the TSF. Note that this may be the initial module, or there may be several modules,
 6854 depending on how much pre-conditioning of the inputs is done. It should be noted that one
 6855 indicator of a pre-conditioning module is that it is invoked for a large number of the TSFI, where
 6856 the TSFI are all of similar type (e.g., system call). The final aspect of accuracy is that the mapping
 6857 makes sense. For instance, mapping a TSFI dealing with access control to a module that checks
 6858 passwords is not accurate. The evaluator should again use judgement in making this determination.
 6859 The goal is that this information aids the evaluator in understanding the system and
 6860 implementation of the SFRs, and ways in which entities at the TSF boundary can interact with the
 6861 TSF. The bulk of the assessment of whether the SFRs are described accurately by the modules is
 6862 performed in other work units.

6863 **11.8.3.5 Action ADV_TDS.3.2E**

6864 **11.8.3.5.1 Work unit ADV_TDS.3-15**

6865 The evaluator **shall examine** the TOE security functional requirements and the TOE design, to
 6866 determine that all ST security functional requirements are covered by the TOE design.

6867 The evaluator may construct a map between the TOE security functional requirements and the TOE
 6868 design. This map will likely be from a functional requirement to a set of subsystems, and later to

6869 modules. Note that this map may have to be at a level of detail below the component or even
6870 element level of the requirements, because of operations (assignments, refinements, selections)
6871 performed on the functional requirement by the ST author.

6872 For example, the **FDP_ACC.1 Subset access control** component contains an element with
6873 assignments. If the ST contained, for instance, ten rules in the **FDP_ACC.1 Subset access control**
6874 assignment, and these ten rules were implemented in specific places within fifteen modules, it
6875 would be inadequate for the evaluator to map **FDP_ACC.1 Subset access control** to one subsystem
6876 and claim the work unit had been completed. Instead, the evaluator would map **FDP_ACC.1 Subset**
6877 **access control** (rule 1) to modules x, y, and z of subsystem A; **FDP_ACC.1 Subset access control** (rule
6878 2) to modules x, p, and q of subsystem A; etc.

6879 **11.8.3.5.2 Work unit ADV_TDS.3-16**

6880 The evaluator *shall examine* the TOE design to determine that it is an accurate instantiation of all
6881 security functional requirements.

6882 The evaluator may construct a map between the TOE security functional requirements and the TOE
6883 design. This map will likely be from a functional requirement to a set of subsystems. Note that this
6884 map may have to be at a level of detail below the component or even element level of the
6885 requirements, because of operations (assignments, refinements, selections) performed on the
6886 functional requirement by the ST author.

6887 As an example, if the ST requirements specified a role-based access control mechanism, the
6888 evaluator would first identify the subsystems, and modules that contribute to this mechanism's
6889 implementation. This could be done by in-depth knowledge or understanding of the TOE design or
6890 by work done in the previous work unit. Note that this trace is only to identify the subsystems, and
6891 modules, and is not the complete analysis.

6892 The next step would be to understand what mechanism the subsystems, and modules implemented.
6893 For instance, if the design described an implementation of access control based on UNIX-style
6894 protection bits, the design would not be an accurate instantiation of those access control
6895 requirements present in the ST example used above. If the evaluator could not determine that the
6896 mechanism was accurately implemented because of a lack of detail, the evaluator would have to
6897 assess whether all of the SFR-enforcing subsystems and modules have been identified, or if
6898 adequate detail had been provided for those subsystems and modules.

6899 **11.8.4 Evaluation of sub-activity (ADV_TDS.4)**

6900 **11.8.4.1 Objectives**

6901 The objective of this sub-activity is to determine whether the TOE design provides a description of
6902 the TOE in terms of subsystems sufficient to determine the TSF boundary, and provides a
6903 description of the TSF internals in terms of modules (and optionally higher-level abstractions). It
6904 provides a detailed description of the SFR-enforcing and SFR-supporting modules and enough
6905 information about the SFR-non-interfering modules for the evaluator to determine that the SFRs
6906 are completely and accurately implemented; as such, the TOE design provides an explanation of the
6907 implementation representation.

6908 **11.8.4.2 Input**

6909 The evaluation evidence for this sub-activity is:

- 6910 a) the ST;
- 6911 b) the functional specification;
- 6912 c) security architecture description;

6913 d) the TOE design.

6914 **11.8.4.3 Application notes**

6915 There are three types of activity that the evaluator must undertake with respect to the TOE design.
 6916 First, the evaluator determines that the TSF boundary has been adequately described. Second, the
 6917 evaluator determines that the developer has provided documentation that conforms to the content
 6918 and presentation requirements this subsystem, and that is consistent with other documentation
 6919 provided for the TOE. Finally, the evaluator must analyse the design information provided for the
 6920 SFR-enforcing modules (at a detailed level) and the SFR-supporting and SFR-non-interfering
 6921 modules (at a less detailed level) to understand how the system is implemented, and with that
 6922 knowledge ensure that the TSFI in the functional specification are adequately described, and that
 6923 the test information adequately tests the TSF (done in the Class ATE: Tests work units).

6924 **11.8.4.4 Action ADV_TDS.4.1E**

6925 ISO/IEC 15408-3 ADV_TDS.4.1C: *The design shall describe the structure of the TOE in terms of*
 6926 *subsystems.*

6927 **11.8.4.4.1 Work unit ADV_TDS.4-1**

6928 The evaluator **shall examine** the TOE design to determine that the structure of the entire TOE is
 6929 described in terms of subsystems.

6930 The evaluator ensures that all of the subsystems of the TOE are identified. This description of the
 6931 TOE will be used as input to work unit ADV_TDS.4-4, where the parts of the TOE that make up the
 6932 TSF are identified. That is, this requirement is on the entire TOE rather than on only the TSF.

6933 The TOE (and TSF) may be described in multiple layers of abstraction (i.e. subsystems and
 6934 modules). Depending upon the complexity of the TOE, its design may be described in terms of
 6935 subsystems and modules, as described in ISO/IEC 15408-3 **Annex A.4, ADV_TDS: Subsystems and**
 6936 **Modules**. For a very simple TOE that can be described solely at the “module” level (see ADV_TDS.4-
 6937 2), this work unit is not applicable and therefore considered to be satisfied.

6938 In performing this activity, the evaluator examines other evidence presented for the TOE (e.g., ST,
 6939 operator user guidance) to determine that the description of the TOE in such evidence is consistent
 6940 with the description contained in the TOE design.

6941 ISO/IEC 15408-3 ADV_TDS.4.2C: *The design shall describe the TSF in terms of modules, designating*
 6942 *each module as SFR-enforcing, SFR-supporting, or SFR-non-interfering.*

6943 **11.8.4.4.2 Work unit ADV_TDS.4-2**

6944 The evaluator **shall examine** the TOE design to determine that the entire TSF is described in terms
 6945 of modules.

6946 The evaluator will examine the modules for specific properties in other work units; in this work
 6947 unit the evaluator determines that the modular description covers the entire TSF, and not just a
 6948 portion of the TSF. The evaluator uses other evidence provided for the evaluation (e.g., functional
 6949 specification, architectural description) in making this determination. For example, if the functional
 6950 specification contains interfaces to functionality that does not appear to be described in the TOE
 6951 design description, it may be the case that a portion of the TSF has not been included appropriately.
 6952 Making this determination will likely be an iterative process, where as more analysis is done on the
 6953 other evidence, more confidence can be gained with respect to the completeness of the
 6954 documentation.

6955 Unlike subsystems, modules describe the implementation in a level of detail that can serve as a
 6956 guide to reviewing the implementation representation. A description of a module should be such

6957 that one could create an implementation of the module from the description, and the resulting
6958 implementation would be 1) identical to the actual TSF implementation in terms of the interfaces
6959 presented, 2) identical in the use of interfaces that are mentioned in the design, and 3) functionally
6960 equivalent to the description of the purpose of the TSF module. For instance, RFC 793 provides a
6961 high-level description of the TCP protocol. It is necessarily implementation independent. While it
6962 provides a wealth of detail, it is **not** a suitable design description because it is not specific to an
6963 implementation. An actual implementation can add to the protocol specified in the RFC, and
6964 implementation choices (for instance, the use of global data vs. local data in various parts of the
6965 implementation) may have an impact on the analysis that is performed. The design description of
6966 the TCP module would list the interfaces presented by the implementation (rather than just those
6967 defined in RFC 793), as well as an algorithm description of the processing associated with the
6968 modules implementing TCP (assuming it was part of the TSF).

6969 **11.8.4.4.3 Work unit ADV_TDS.4-3**

6970 The evaluator **shall check** the TOE design to determine that the TSF modules are identified as
6971 either SFR-enforcing, SFR-supporting, or SFR-non-interfering.

6972 The purpose of designating each module (according to the role a particular module plays in the
6973 enforcement of the SFRs) is to allow developers to provide less information about the parts of the
6974 TSF that have little role in security. It is always permissible for the developer to provide more
6975 information or detail than the requirements demand, as might occur when the information has
6976 been gathered outside the evaluation context. In such cases the developer must still designate the
6977 modules as either SFR-enforcing, SFR-supporting, or SFR-non-interfering.

6978 The accuracy of these designations is continuously reviewed as the evaluation progresses. The
6979 concern is the mis-designation of modules as being less important (and hence, having less
6980 information) than is really the case. While blatant mis-designations may be immediately apparent
6981 (e.g., designating an authentication module as anything but SFR-enforcing when **User identification**
6982 **(FIA_UID)** is one of the SFRs being claimed), other mis-designations might not be discovered until
6983 the TSF is better understood. The evaluator must therefore keep in mind that these designations
6984 are the developer's initial best effort, but are subject to change. Further guidance is provided under
6985 work unit ADV_TDS.4-17, which examines the accuracy of these designations.

6986 ISO/IEC 15408-3 ADV_TDS.4.3C: *The design shall identify all subsystems of the TSF.*

6987 **11.8.4.4.4 Work unit ADV_TDS.4-4**

6988 The evaluator **shall examine** the TOE design to determine that all subsystems of the TSF are
6989 identified.

6990 If the design is presented solely in terms of modules, then subsystems in these requirements are
6991 equivalent to modules and the activity should be performed at the module level.

6992 In work unit ADV_TDS.4-1 all of the subsystems of the TOE were identified, and a determination
6993 made that the non-TSF subsystems were correctly characterised. Building on that work, the
6994 subsystems that were not characterised as non-TSF subsystems should be precisely identified. The
6995 evaluator determines that, of the hardware and software installed and configured according to the
6996 Preparative procedures (AGD_PRE) guidance, each subsystem has been accounted for as either one
6997 that is part of the TSF, or one that is not.

6998 ISO/IEC 15408-3 ADV_TDS.4.4C: *The design shall provide a semiformal description of each*
6999 *subsystem of the TSF, supported by informal, explanatory text where appropriate.*

7000 **11.8.4.4.5 Work unit ADV_TDS.4-5**

7001 The evaluator **shall examine** the TDS documentation to determine that the semiformal notation
7002 used for describing the subsystems, modules and their interfaces is defined or referenced.

7003 A semiformal notation can be either defined by the sponsor or a corresponding standard be
 7004 referenced. The evaluator should provide a mapping of security functions and their interfaces
 7005 outlining in what part of the documentation a function or interface is semiformal described and
 7006 what notation is used. The evaluator examines all semiformal notations used to make sure that
 7007 they are of a semiformal style and to justify the appropriateness of the manner how the semiformal
 7008 notations are used for the TOE.

7009 The evaluator is reminded that a semi-formal presentation is characterised by a standardised
 7010 format with a well-defined syntax that reduces ambiguity that may occur in informal presentations.
 7011 The syntax of all semiformal notations used in the functional specification shall be defined or a
 7012 corresponding standard be referenced. The evaluator verifies that the semiformal notations used
 7013 for expressing the functional specification are capable of expressing features relevant to security.
 7014 In order to determine this, the evaluator can refer to the SFR and compare the TSF security
 7015 features stated in the ST and those described in the FSP using the semiformal notations.

7016 **11.8.4.4.6 Work unit ADV_TDS.4-6**

7017 The evaluator *shall examine* the TOE design to determine that each subsystem of the TSF
 7018 describes its role in the enforcement of SFRs described in the ST.

7019 If the design is presented solely in terms of modules, then this work unit will be considered
 7020 satisfied by the assessment done in subsequent work units; no explicit action on the part of the
 7021 evaluator is necessary in this case.

7022 On systems that are complex enough to warrant a subsystem-level description of the TSF in
 7023 addition to the modular description, the goal of the subsystem-level description is to give the
 7024 evaluator context for the modular description that follows. Therefore, the evaluator ensures that
 7025 the subsystem-level description contains a description of how the security functional requirements
 7026 are achieved in the design, but at a level of abstraction above the modular description. This
 7027 description should discuss the mechanisms used at a level that is aligned with the module
 7028 description; this will provide the evaluators the road map needed to intelligently assess the
 7029 information contained in the module description. A well-written set of subsystem descriptions will
 7030 help guide the evaluator in determining the modules that are most important to examine, thus
 7031 focusing the evaluation activity on the portions of the TSF that have the most relevance with
 7032 respect to the enforcement of the SFRs.

7033 The evaluator ensures that all subsystems of the TSF have a description. While the description
 7034 should focus on the role that the subsystem plays in enforcing or supporting the implementation of
 7035 the SFRs, enough information must be present so that a context for understanding the SFR-related
 7036 functionality is provided.

7037 **11.8.4.4.7 Work unit ADV_TDS.4-7**

7038 The evaluator *shall examine* the TOE design to determine that each SFR-non-interfering
 7039 subsystem of the TSF is described such that the evaluator can determine that the subsystem is SFR-
 7040 non-interfering.

7041 If the design is presented solely in terms of modules, then this work unit will be considered
 7042 satisfied by the assessment done in subsequent work units; no explicit action on the part of the
 7043 evaluator is necessary in this case.

7044 An SFR-non-interfering subsystem is one on which the SFR-enforcing and SFR-supporting
 7045 subsystems have no dependence; that is, they play no role in implementing SFR functionality.

7046 The evaluator ensures that all subsystems of the TSF have a description. While the description
 7047 should focus on the role that the subsystem do not plays in enforcing or supporting the
 7048 implementation of the SFRs, enough information must be present so that a context for
 7049 understanding the SFR-non-interfering functionality is provided.

7050 ISO/IEC 15408-3 ADV_TDS.4.5C: *The design shall provide a description of the interactions among all*
7051 *subsystems of the TSF.*

7052 **11.8.4.4.8 Work unit ADV_TDS.4-8**

7053 The evaluator *shall examine* the TOE design to determine that interactions between the
7054 subsystems of the TSF are described.

7055 If the design is presented solely in terms of modules, then this work unit will be considered
7056 satisfied by the assessment done in subsequent work units; no explicit action on the part of the
7057 evaluator is necessary in this case.

7058 On systems that are complex enough to warrant a subsystem-level description of the TSF in
7059 addition to the modular description, the goal of describing the interactions between the
7060 subsystems is to help provide the reader a better understanding of how the TSF performs its
7061 functions. These interactions do not need to be characterised at the implementation level (e.g.,
7062 parameters passed from one routine in a subsystem to a routine in a different subsystem; global
7063 variables; hardware signals (e.g., interrupts) from a hardware subsystem to an interrupt-handling
7064 subsystem), but the data elements identified for a particular subsystem that are going to be used by
7065 another subsystem need to be covered in this discussion. Any control relationships between
7066 subsystems (e.g., a subsystem responsible for configuring a rule base for a firewall system and the
7067 subsystem that actually implements these rules) should also be described.

7068 It should be noted while the developer should characterise all interactions between subsystems,
7069 the evaluators need to use their own judgement in assessing the completeness of the description. If
7070 the reason for an interaction is unclear, or if there are SFR-related interactions (discovered, for
7071 instance, in examining the module-level documentation) that do not appear to be described, the
7072 evaluator ensures that this information is provided by the developer. However, if the evaluator can
7073 determine that interactions among a particular set of subsystems, while incompletely described by
7074 the developer, and a complete description will not aid in understanding the overall functionality
7075 nor security functionality provided by the TSF, then the evaluator may choose to consider the
7076 description sufficient, and not pursue completeness for its own sake.

7077 ISO/IEC 15408-3 ADV_TDS.4.6C: *The design shall provide a mapping from the subsystems of the TSF*
7078 *to the modules of the TSF.*

7079 **11.8.4.4.9 Work unit ADV_TDS.4-9**

7080 The evaluator *shall examine* the TOE design to determine that the mapping between the
7081 subsystems of the TSF and the modules of the TSF is complete.

7082 If the design is presented solely in terms of modules, then this work unit is considered satisfied.

7083 For TOEs that are complex enough to warrant a subsystem-level description of the TSF in addition
7084 to the modular description, the developer provides a simple mapping showing how the modules of
7085 the TSF are allocated to the subsystems. This will provide the evaluator a guide in performing their
7086 module-level assessment. To determine completeness, the evaluator examines each mapping and
7087 determines that all subsystems map to at least one module, and that all modules map to exactly one
7088 subsystem.

7089 **11.8.4.4.10 Work unit ADV_TDS.4-10**

7090 The evaluator *shall examine* the TOE design to determine that the mapping between the
7091 subsystems of the TSF to the modules of the TSF is accurate.

7092 If the design is presented solely in terms of modules, then this work unit is considered satisfied.

7093 For TOEs that are complex enough to warrant a subsystem-level description of the TSF in addition
 7094 to the modular description, the developer provides a simple mapping showing how the modules of
 7095 the TSF are allocated to the subsystems. This will provide the evaluator a guide in performing their
 7096 module-level assessment. The evaluator may choose to check the accuracy of the mapping in
 7097 conjunction with performing other work units. An “inaccurate” mapping is one where the module
 7098 is mistakenly associated with a subsystem where its functions are not used within the subsystem.
 7099 Because the mapping is intended to be a guide supporting more detailed analysis, the evaluator is
 7100 cautioned to apply appropriate effort to this work unit. Expending extensive evaluator resources
 7101 verifying the accuracy of the mapping is not necessary. Inaccuracies that lead to mis-
 7102 understandings related to the design that are uncovered as part of this or other work units are the
 7103 ones that should be associated with this work unit and corrected.

7104 ISO/IEC 15408-3 ADV_TDS.4.7C: *The design shall describe each SFR-enforcing and SFR-supporting*
 7105 *module in terms of its purpose and relationship with other modules.*

7106 **11.8.4.4.11 Work unit ADV_TDS.4-11**

7107 The evaluator *shall examine* the TOE design to determine that the description of the purpose of
 7108 each SFR-enforcing and SFR-supporting module, and relationship with other modules is complete
 7109 and accurate.

7110 The developer may designate modules as SFR-enforcing, SFR-supporting, and SFR non-interfering,
 7111 but these “tags” are used only to describe the amount and type of information the developer must
 7112 provide, and can be used to limit the amount of information the developer has to develop if their
 7113 engineering process does not produce the documentation required. Whether the modules have
 7114 been categorised by the developer or not, it is the evaluator's responsibility to determine that the
 7115 modules have the appropriate information for their role (SFR-enforcing, etc.) in the TOE, and to
 7116 obtain the appropriate information from the developer should the developer fail to provide the
 7117 required information for a particular module.

7118 The purpose of a module provides a description indicating what function the module is fulfilling. A
 7119 word of caution to evaluator is in order. The focus of this work unit should be to provide the
 7120 evaluator an understanding of how the module works so that determinations can be made about
 7121 the soundness of the implementation of the SFRs, as well as to support architectural analysis
 7122 performed for ADV_ARC subsystems. As long as the evaluator has a sound understanding of the
 7123 module's operation, and its relationship to other modules and the TOE as a whole, the evaluator
 7124 should consider the objective of the work achieved and not engage in a documentation exercise for
 7125 the developer (by requiring, for example, a complete algorithmic description for a self-evident
 7126 implementation representation).

7127 Because the modules are at such a low level, it may be difficult determine completeness and
 7128 accuracy impacts from other documentation, such as operational user guidance, the functional
 7129 specification, the TSF internals, or the security architecture description. However, the evaluator
 7130 uses the information present in those documents to the extent possible to help ensure that the
 7131 purpose is accurately and completely described. This analysis can be aided by the analysis
 7132 performed for the work units for the **ADV_TDS.4.10C** element, which maps the TSFI in the
 7133 functional specification to the modules of the TSF.

7134 ISO/IEC 15408-3 ADV_TDS.4.8C: *The design shall describe each SFR-enforcing and SFR-supporting*
 7135 *module in terms of its SFR-related interfaces, return values from those interfaces, interaction with*
 7136 *other modules and called SFR-related interfaces to other SFR-enforcing or SFR-supporting modules.*

7137 **11.8.4.4.12 Work unit ADV_TDS.4-12**

7138 The evaluator *shall examine* the TOE design to determine that the description of the interfaces
 7139 presented by each SFR-enforcing and SFR-supporting module contain an accurate and complete
 7140 description of the SFR-related parameters, the invocation conventions for each interface, and any
 7141 values returned directly by the interface.

7142 The SFR-related interfaces of a module are those interfaces used by other modules as a means to
7143 invoke the SFR-related operations provided, and to provide inputs to or receive outputs from the
7144 module. The purpose in the specification of these interfaces is to permit the exercise of them
7145 during testing. Inter-module interfaces that are not SFR-related need not be specified or described,
7146 since they are not a factor in testing. Likewise, other internal interfaces that are not a factor in
7147 traversing SFR-related paths of execution (such as those internal paths that are fixed).

7148 SFR-related interfaces of SFR-supporting modules are all interfaces of SFR-supporting modules
7149 that are called directly or indirectly from SFR-enforcing modules. Those interfaces need to be
7150 described with all the parameter used in such a call. This allows the evaluator to understand the
7151 purpose of the call to the SFR-supporting module in the context of operation of the SFR-enforcing
7152 modules.

7153 SFR-related interfaces are described in terms of how they are invoked, and any values that are
7154 returned. This description would include a list of parameters, and descriptions of these parameters.
7155 Note that global data would also be considered parameters if used by the module (either as inputs
7156 or outputs) when invoked. If a parameter were expected to take on a set of values (e.g., a “flag”
7157 parameter), the complete set of values the parameter could take on that would have an effect on
7158 module processing would be specified. Likewise, parameters representing data structures are
7159 described such that each field of the data structure is identified and described. Note that different
7160 programming languages may have additional “interfaces” that would be non-obvious; an example
7161 would be operator/function overloading in C++. This “implicit interface” in the class description
7162 would also be described as part of the low-level TOE design. Note that although a module could
7163 present only one interface, it is more common that a module presents a small set of related
7164 interfaces.

7165 In terms of the assessment of parameters (inputs and outputs) to a module, any use of global data
7166 must also be considered. A module “uses” global data if it either reads or writes the data. In order
7167 to assure the description of such parameters (if used) is complete, the evaluator uses other
7168 information provided about the module in the TOE design (interfaces, algorithmic description, etc.),
7169 as well as the description of the particular set of global data assessed in work unit ADV_TDS.4-12.
7170 For instance, the evaluator could first determine the processing the module performs by examining
7171 its function and interfaces presented (particularly the parameters of the interfaces). They could
7172 then check to see if the processing appears to “touch” any of the global data areas identified in the
7173 TDS design. The evaluator then determines that, for each global data area that appears to be
7174 “touched”, that global data area is listed as a means of input or output by the module the evaluator
7175 is examining.

7176 Invocation conventions are a programming-reference-type description that one could use to
7177 correctly invoke a module's interface if one were writing a program to make use of the module's
7178 functionality through that interface. This includes necessary inputs and outputs, including any set-
7179 up that may need to be performed with respect to global variables.

7180 Values returned through the interface refer to values that are either passed through parameters or
7181 messages; values that the function call itself returns in the style of a “C” program function call; or
7182 values passed through global means (such as certain error routines in *ix-style operating systems).

7183 In order to assure the description is complete, the evaluator uses other information provided about
7184 the module in the TOE design (e.g., algorithmic description, global data used) to ensure that it
7185 appears all data necessary for performing the functions of the module is presented to the module,
7186 and that any values that other modules expect the module under examination to provide are
7187 identified as being returned by the module. The evaluator determines accuracy by ensuring that
7188 the description of the processing matches the information listed as being passed to or from an
7189 interface.

7190 ISO/IEC 15408-3 ADV_TDS.4.9C: *The design shall describe each SFR-non-interfering module in terms*
7191 *of its purpose and interaction with other modules.*

7192 **11.8.4.4.13 Work unit ADV_TDS.4-13**

7193 The evaluator *shall examine* the TOE design to determine that SFR-non-interfering modules are
7194 correctly categorised.

7195 As mentioned in work unit ADV_TDS.4-2, less information is required about modules that are SFR-
7196 non-interfering. A key focus of the evaluator for this work unit is attempting to determine from the
7197 evidence provided for each module implicitly categorised as SFR-non-interfering and the
7198 evaluation (information about other modules in the TOE design, the functional specification, the
7199 security architecture description, the operational user guidance, the TSF internals document, and
7200 perhaps even the implementation representation) whether the module is indeed SFR-non-
7201 interfering. At this level of assurance some error should be tolerated; the evaluator does not have
7202 to be absolutely sure that a given module is SFR-non-interfering, even though it is labelled as such.
7203 However, if the evidence provided indicates that a SFR-non-interfering module is SFR-enforcing or
7204 SFR-supporting, the evaluator requests additional information from the developer in order to
7205 resolve the apparent inconsistency. For example, suppose the documentation for Module A (an
7206 SFR-enforcing module) indicates that it calls Module B to perform an access check on a certain type
7207 of construct. When the evaluator examines the information associated with Module B, it is
7208 discovered that the only information the developer has provided is a purpose and a set of
7209 interactions (thus implicitly categorising Module B as SFR-supporting or SFR-non-interfering). On
7210 examining the purpose and interactions from Module A, the evaluator finds no mention of Module
7211 B performing any access checks, and Module A is not listed as a module with which Module B
7212 interacts. At this point the evaluator should approach the developer to resolve the discrepancies
7213 between the information provided in Module A and that in Module B.

7214 Another example would be where the evaluator examines the mapping of the TSFI to the modules
7215 as provided by [ADV_TDS.4.2D](#). This examination shows that Module C is associated with an SFR
7216 requiring identification of the user. Again, when the evaluator examines the information associated
7217 with Module C, they find that all the developer has provided is a purpose and a set of interactions
7218 (thus implicitly categorising Module C as SFR-non-interfering). Examining the purpose and
7219 interactions presented for Module C, the evaluator is unable to determine why Module C, listed as
7220 mapping to a TSFI concerned with user identification, would not be classified as SFR-enforcing or
7221 SFR-supporting. Again, the evaluator should approach the developer to resolve this discrepancy.

7222 A final example illustrates the opposite situation. As before, the developer has provided
7223 information associated with Module D consisting of a purpose and a set of interactions (thus
7224 implicitly categorising Module D as SFR-non-interfering). The evaluator examines all of the
7225 evidence provided, including the purpose and interactions for Module D. The purpose appears to
7226 give a meaningful description of Module D's function in the TOE, the interactions are consistent
7227 with that description, and there is nothing to indicate that Module D is SFR-enforcing or SFR-
7228 supporting. In this case, the evaluator should not demand more information about Module D "just
7229 be to sure" it is correctly categorised. The developer has met the obligations and the resulting
7230 assurance the evaluator has in the implicit categorisation of Module D is (by definition)
7231 appropriate for this assurance level.

7232 **11.8.4.4.14 Work unit ADV_TDS.4-14**

7233 The evaluator *shall examine* the TOE design to determine that the description of the purpose of
7234 each SFR-non-interfering module is complete and accurate.

7235 The description of the purpose of a module indicates what function the module is fulfilling. From
7236 the description, the evaluator should be able to obtain a general idea of the module's role. In order
7237 to assure the description is complete, the evaluator uses the information provided about the
7238 module's interactions with other modules to assess whether the reasons for the module being
7239 called are consistent with the module's purpose. If the interaction description contains
7240 functionality that is not apparent from, or in conflict with, the module's purpose, the evaluator
7241 needs to determine whether the problem is one of accuracy or of completeness. The evaluator

7242 should be wary of purposes that are too short, since meaningful analysis based on a one-sentence
7243 purpose is likely to be impossible.

7244 Because the modules are at such a low level, it may be difficult determine completeness and
7245 accuracy impacts from other documentation, such as operational user guidance, the functional
7246 specification, the security architecture description, or the TSF internals document. However, the
7247 evaluator uses the information present in those documents to the extent possible to help ensure
7248 that the function is accurately and completely described. This analysis can be aided by the analysis
7249 performed for the work units for the **ADV_TDS.4.10C** element, which maps the TSFI in the
7250 functional specification to the modules of the TSF.

7251 **11.8.4.4.15 Work unit ADV_TDS.4-15**

7252 The evaluator **shall examine** the TOE design to determine that the description of a SFR-non-
7253 interfering module's interaction with other modules is complete and accurate.

7254 It is important to note that, in terms of the Part 3 requirement and this work unit, the term
7255 *interaction* is intended to convey less rigour than *interface*. An interaction does not need to be
7256 characterised at the implementation level (e.g., parameters passed from one routine in a module to
7257 a routine in a different module; global variables; hardware signals (e.g., interrupts) from a
7258 hardware subsystem to an interrupt-handling subsystem), but the data elements identified for a
7259 particular module that are going to be used by another module should be covered in this discussion.
7260 Any control relationships between modules (e.g., a module responsible for configuring a rule base
7261 for a firewall system and the module that actually implements these rules) should also be
7262 described.

7263 A module's interaction with other modules can be captured in many ways. The intent for the TOE
7264 design is to allow the evaluator to understand (in part through analysis of module interactions) the
7265 role of the SFR-supporting and SFR-non-interfering modules in the overall TOE design.
7266 Understanding of this role will aid the evaluator in performing work unit ADV_TDS.4-8.

7267 A module's interaction with other modules goes beyond just a call-tree-type document. The
7268 interaction is described from a functional perspective of why a module interacts with other
7269 modules. The module's purpose describes what functions the module provides to other modules;
7270 the interactions should describe what the module depends on from other modules in order to
7271 accomplish this function.

7272 Because the modules are at such a low level, it may be difficult determine completeness and
7273 accuracy impacts from other documentation, such as operational user guidance, the functional
7274 specification, the security architecture description, or the TSF internals document. However, the
7275 evaluator uses the information present in those documents to the extent possible to help ensure
7276 that the interactions are accurately and completely described.

7277 ISO/IEC 15408-3 ADV_TDS.4.10C: *The mapping shall demonstrate that all TSFIs trace to the*
7278 *behaviour described in the TOE design that they invoke.*

7279 **11.8.4.4.16 Work unit ADV_TDS.4-16**

7280 The evaluator **shall examine** the TOE design to determine that it contains a complete and accurate
7281 mapping from the TSFI described in the functional specification to the modules of the TSF
7282 described in the TOE design.

7283 The modules described in the TOE design provide a description of the implementation of the TSF.
7284 The TSFI provide a description of how the implementation is exercised. The evidence from the
7285 developer identifies the module that is initially invoked when an operation is requested at the TSFI,
7286 and identify the chain of modules invoked up to the module that is primarily responsible for
7287 implementing the functionality. However, a complete call tree for each TSFI is not required for this
7288 work unit. The cases in which more than one module would have to be identified are where there

7289 are “entry point” modules or wrapper modules that have no functionality other than conditioning
7290 inputs or de-multiplexing an input. Mapping to one of these modules would not provide any useful
7291 information to the evaluator.

7292 The evaluator assesses the completeness of the mapping by ensuring that all of the TSFI map to at
7293 least one module. The verification of accuracy is more complex.

7294 The first aspect of accuracy is that each TSFI is mapped to a module at the TSF boundary. This
7295 determination can be made by reviewing the module description and its interfaces/interactions.
7296 The next aspect of accuracy is that each TSFI identifies a chain of modules between the initial
7297 module identified and a module that is primarily responsible for implementing the function
7298 presented at the TSF. Note that this may be the initial module, or there may be several modules,
7299 depending on how much pre-conditioning of the inputs is done. It should be noted that one
7300 indicator of a pre-conditioning module is that it is invoked for a large number of the TSFI, where
7301 the TSFI are all of similar type (e.g., system call). The final aspect of accuracy is that the mapping
7302 makes sense. For instance, mapping a TSFI dealing with access control to a module that checks
7303 passwords is not accurate. The evaluator should again use judgement in making this determination.
7304 The goal is that this information aids the evaluator in understanding the system and
7305 implementation of the SFRs, and ways in which entities at the TSF boundary can interact with the
7306 TSF. The bulk of the assessment of whether the SFRs are described accurately by the modules is
7307 performed in other work units.

7308 **11.8.4.5 Action ADV_TDS.4.2E**

7309 **11.8.4.5.1 Work unit ADV_TDS.4-17**

7310 The evaluator *shall examine* the TOE security functional requirements and the TOE design, to
7311 determine that all ST security functional requirements are covered by the TOE design.

7312 The evaluator may construct a map between the TOE security functional requirements and the TOE
7313 design. This map will likely be from a functional requirement to a set of subsystems, and later to
7314 modules. Note that this map may have to be at a level of detail below the component or even
7315 element level of the requirements, because of operations (assignments, refinements, selections)
7316 performed on the functional requirement by the ST author.

7317 For example, the **FDP_ACC.1 Subset access control** component contains an element with
7318 assignments. If the ST contained, for instance, ten rules in the **FDP_ACC.1 Subset access control**
7319 assignment, and these ten rules were implemented in specific places within fifteen modules, it
7320 would be inadequate for the evaluator to map **FDP_ACC.1 Subset access control** to one subsystem
7321 and claim the work unit had been completed. Instead, the evaluator would map **FDP_ACC.1 Subset**
7322 **access control** (rule 1) to modules x, y and z of subsystem A; **FDP_ACC.1 Subset access control** (rule
7323 2) to x, p, and q of subsystem A; etc.

7324 **11.8.4.5.2 Work unit ADV_TDS.4-18**

7325 The evaluator *shall examine* the TOE design to determine that it is an accurate instantiation of all
7326 security functional requirements.

7327 The evaluator may construct a map between the TOE security functional requirements and the TOE
7328 design. This map will likely be from a functional requirement to a set of subsystems. Note that this
7329 map may have to be at a level of detail below the component or even element level of the
7330 requirements, because of operations (assignments, refinements, selections) performed on the
7331 functional requirement by the ST author.

7332 As an example, if the ST requirements specified a role-based access control mechanism, the
7333 evaluator would first identify the subsystems, and modules that contribute to this mechanism's
7334 implementation. This could be done by in-depth knowledge or understanding of the TOE design or

7335 by work done in the previous work unit. Note that this trace is only to identify the subsystems, and
7336 modules, and is not the complete analysis.

7337 The next step would be to understand what mechanism the subsystems, and modules implemented.
7338 For instance, if the design described an implementation of access control based on UNIX-style
7339 protection bits, the design would not be an accurate instantiation of those access control
7340 requirements present in the ST example used above. If the evaluator could not determine that the
7341 mechanism was accurately implemented because of a lack of detail, the evaluator would have to
7342 assess whether all of the SFR-enforcing subsystems and modules have been identified, or if
7343 adequate detail had been provided for those subsystems and modules.

7344 **11.8.5 Evaluation of sub-activity (ADV_TDS.5)**

7345 **11.8.5.1 Objectives**

7346 The objectives of this sub-activity are to determine whether the TOE design provides a description
7347 of the TOE in terms of subsystems sufficient to determine the TSF boundary, and provides a
7348 description of the TSF internals in terms of modules (and optionally higher-level abstractions). It
7349 provides enough information about the modules for the evaluator to determine that the SFRs are
7350 completely and accurately implemented; as such, the TOE design provides an explanation of the
7351 implementation representation.

7352 **11.8.5.2 Input**

7353 The evaluation evidence for this sub-activity is:

- 7354 a) the ST;
- 7355 b) the functional specification;
- 7356 c) security architecture description;
- 7357 d) the TOE design.

7358 **11.8.5.3 Application notes**

7359 There are three types of activity that the evaluator must undertake with respect to the TOE design.
7360 First, the evaluator determines that the TSF boundary has been adequately described. Second, the
7361 evaluator determines that the developer has provided documentation that conforms to the content
7362 and presentation requirements this subsystem, and that is consistent with other documentation
7363 provided for the TOE. Finally, the evaluator must analyse the design information provided for the
7364 modules (at a detailed level) to understand how the system is implemented, and with that
7365 knowledge ensure that the TSFI in the functional specification are adequately described, and that
7366 the test information adequately tests the TSF (done in the Class ATE: Tests work units).

7367 **11.8.5.4 Action ADV_TDS.5.1E**

7368 ADV_TDS.5.1C *The design shall describe the structure of the TOE in terms of subsystems.*

7369 **11.8.5.4.1 Work unit ADV_TDS.5-1**

7370 The evaluator **shall examine** the TOE design to determine that the structure of the entire TOE is
7371 described in terms of subsystems.

7372 The evaluator ensures that all of the subsystems of the TOE are identified. This description of the
7373 TOE will be used as input to work unit ADV_TDS.5-4, where the parts of the TOE that make up the
7374 TSF are identified. That is, this requirement is on the entire TOE rather than on only the TSF.

7375 The TOE (and TSF) may be described in multiple layers of abstraction (i.e. subsystems and
7376 modules). Depending upon the complexity of the TOE, its design may be described in terms of
7377 subsystems and modules, as described in CC Part 3 Annex A.4, ADV_TDS: Subsystems and Modules.
7378 For a very simple TOE that can be described solely at the “module” level (see ADV_TDS.5-2), this
7379 work unit is not applicable and therefore considered to be satisfied.

7380 In performing this activity, the evaluator examines other evidence presented for the TOE (e.g., ST,
7381 operator user guidance) to determine that the description of the TOE in such evidence is consistent
7382 with the description contained in the TOE design.

7383 ADV_TDS.5.2C *The design shall describe the TSF in terms of modules, designating each module as*
7384 *SFR-enforcing, SFR-supporting, or SFR-non-interfering.*

7385 **11.8.5.4.2 Work unit ADV_TDS.5-2**

7386 The evaluator **shall examine** the TOE design to determine that the entire TSF is described in terms
7387 of modules.

7388 The evaluator will examine the modules for specific properties in other work units; in this work
7389 unit the evaluator determines that the modular description covers the entire TSF, and not just a
7390 portion of the TSF. The evaluator uses other evidence provided for the evaluation (e.g., functional
7391 specification, architectural description) in making this determination. For example, if the functional
7392 specification contains interfaces to functionality that does not appear to be described in the TOE
7393 design description, it may be the case that a portion of the TSF has not been included appropriately.
7394 Making this determination will likely be an iterative process, where as more analysis is done on the
7395 other evidence, more confidence can be gained with respect to the completeness of the
7396 documentation.

7397 Unlike subsystems, modules describe the implementation in a level of detail that can serve as a
7398 guide to reviewing the implementation representation. A description of a module should be such
7399 that one could create an implementation of the module from the description, and the resulting
7400 implementation would be 1) identical to the actual TSF implementation in terms of the interfaces
7401 presented, 2) identical in the use of interfaces that are mentioned in the design, and 3) functionally
7402 equivalent to the description of the purpose of the TSF module. For instance, RFC 793 provides a
7403 high-level description of the TCP protocol. It is necessarily implementation independent. While it
7404 provides a wealth of detail, it is **not** a suitable design description because it is not specific to an
7405 implementation. An actual implementation can add to the protocol specified in the RFC, and
7406 implementation choices (for instance, the use of global data vs. local data in various parts of the
7407 implementation) may have an impact on the analysis that is performed. The design description of
7408 the TCP module would list the interfaces presented by the implementation (rather than just those
7409 defined in RFC 793), as well as an algorithm description of the processing associated with the
7410 modules implementing TCP (assuming it was part of the TSF).

7411 **11.8.5.4.3 Work unit ADV_TDS.5-3**

7412 The evaluator **shall check** the TOE design to determine that the TSF modules are identified as
7413 either SFR-enforcing, SFR-supporting, or SFR-non-interfering.

7414 The purpose of designating each module (according to the role a particular module plays in the
7415 enforcement of the SFRs) is to allow developers to provide less information about the parts of the
7416 TSF that have little role in security. It is always permissible for the developer to provide more
7417 information or detail than the requirements demand, as might occur when the information has
7418 been gathered outside the evaluation context. In such cases the developer must still designate the
7419 modules as either SFR-enforcing, SFR-supporting, or SFR-non-interfering.

7420 The accuracy of these designations is continuously reviewed as the evaluation progresses. The
7421 concern is the mis-designation of modules as being less important (and hence, having less
7422 information) than is really the case. While blatant mis-designations may be immediately apparent

7423 (e.g., designating an authentication module as anything but SFR-enforcing when User identification
7424 (FIA_UID) is one of the SFRs being claimed), other mis-designations might not be discovered until
7425 the TSF is better understood. The evaluator must therefore keep in mind that these designations
7426 are the developer's initial best effort, but are subject to change. Further guidance is provided under
7427 work unit ADV_TDS.5-16, which examines the accuracy of these designations.

7428 *ADV_TDS.5.3C The design shall identify all subsystems of the TSF.*

7429 **11.8.5.4.4 Work unit ADV_TDS.5-4**

7430 The evaluator **shall examine** the TOE design to determine that all subsystems of the TSF are
7431 identified.

7432 If the design is presented solely in terms of modules, then subsystems in these requirements are
7433 equivalent to modules and the activity should be performed at the module level.

7434 In work unit ADV_TDS.5-1 all of the subsystems of the TOE were identified, and a determination
7435 made that the non-TSF subsystems were correctly characterised. Building on that work, the
7436 subsystems that were not characterised as non-TSF subsystems should be precisely identified. The
7437 evaluator determines that, of the hardware and software installed and configured according to the
7438 Preparative procedures (AGD_PRE) guidance, each subsystem has been accounted for as either one
7439 that is part of the TSF, or one that is not.

7440 *ADV_TDS.5.4C The design shall provide a semiformal description of each subsystem of the TSF,*
7441 *supported by informal, explanatory text where appropriate.*

7442 **11.8.5.4.5 Work unit ADV_TDS.5-5**

7443 The evaluator **shall examine** the TDS documentation to determine that the semiformal notation
7444 used for describing the subsystems, modules and their interfaces is defined or referenced.

7445 A semiformal notation can be either defined by the sponsor or a corresponding standard be
7446 referenced. The evaluator should provide a mapping of security functions and their interfaces
7447 outlining in what part of the documentation a function or interface is semiformal described and
7448 what notation is used. The evaluator examines all semiformal notations used to make sure that
7449 they are of a semiformal style and to justify the appropriateness of the manner how the semiformal
7450 notations are used for the TOE.

7451 The evaluator is reminded that a semi-formal presentation is characterised by a standardised
7452 format with a well-defined syntax that reduces ambiguity that may occur in informal presentations.
7453 The syntax of all semiformal notations used in the functional specification shall be defined or a
7454 corresponding standard be referenced. The evaluator verifies that the semiformal notations used
7455 for expressing the functional specification are capable of expressing features relevant to security.
7456 In order to determine this, the evaluator can refer to the SFR and compare the TSF security
7457 features stated in the ST and those described in the FSP using the semiformal notations.

7458 Note that ADV_TDS.5.7C requires the module description to be semiformal. This work unit
7459 therefore applies also to that description.

7460 **11.8.5.4.6 Work unit ADV_TDS.5-6**

7461 The evaluator **shall examine** the TOE design to determine that each subsystem of the TSF
7462 describes its role in the enforcement of SFRs described in the ST.

7463 If the design is presented solely in terms of modules, then this work unit will be considered
7464 satisfied by the assessment done in subsequent work units; no explicit action on the part of the
7465 evaluator is necessary in this case.

7466 On systems that are complex enough to warrant a subsystem-level description of the TSF in
 7467 addition to the modular description, the goal of the subsystem-level description is to give the
 7468 evaluator context for the modular description that follows. Therefore, the evaluator ensures that
 7469 the subsystem-level description contains a description of how the security functional requirements
 7470 are achieved in the design, but at a level of abstraction above the modular description. This
 7471 description should discuss the mechanisms used at a level that is aligned with the module
 7472 description; this will provide the evaluators the road map needed to intelligently assess the
 7473 information contained in the module description. A well-written set of subsystem descriptions will
 7474 help guide the evaluator in determining the modules that are most important to examine, thus
 7475 focusing the evaluation activity on the portions of the TSF that have the most relevance with
 7476 respect to the enforcement of the SFRs.

7477 The evaluator ensures that all subsystems of the TSF have a description. While the description
 7478 should focus on the role that the subsystem plays in enforcing or supporting the implementation of
 7479 the SFRs, enough information must be present so that a context for understanding the SFR-related
 7480 functionality is provided.

7481 **11.8.5.4.7 Work unit ADV_TDS.5-7**

7482 The evaluator *shall examine* the TOE design to determine that each SFR-non-interfering
 7483 subsystem of the TSF is described such that the evaluator can determine that the subsystem is SFR-
 7484 non-interfering.

7485 If the design is presented solely in terms of modules, then this work unit will be considered
 7486 satisfied by the assessment done in subsequent work units; no explicit action on the part of the
 7487 evaluator is necessary in this case.

7488 An SFR-non-interfering subsystem is one on which the SFR-enforcing and SFR-supporting
 7489 subsystems have no dependence; that is, they play no role in implementing SFR functionality.

7490 The evaluator ensures that all subsystems of the TSF have a description. While the description
 7491 should focus on the role that the subsystem do not plays in enforcing or supporting the
 7492 implementation of the SFRs, enough information must be present so that a context for
 7493 understanding the SFR-non-interfering functionality is provided.

7494 ADV_TDS.5.5C *The design shall provide a description of the interactions among all subsystems of*
 7495 *the TSF.*

7496 **11.8.5.4.8 Work unit ADV_TDS.5-8**

7497 The evaluator *shall examine* the TOE design to determine that interactions between the
 7498 subsystems of the TSF are described.

7499 If the design is presented solely in terms of modules, then this work unit will be considered
 7500 satisfied by the assessment done in subsequent work units; no explicit action on the part of the
 7501 evaluator is necessary in this case.

7502 On systems that are complex enough to warrant a subsystem-level description of the TSF in
 7503 addition to the modular description, the goal of describing the interactions between the
 7504 subsystems is to help provide the reader a better understanding of how the TSF performs its
 7505 functions. These interactions do not need to be characterised at the implementation level (e.g.,
 7506 parameters passed from one routine in a subsystem to a routine in a different subsystem; global
 7507 variables; hardware signals (e.g., interrupts) from a hardware subsystem to an interrupt-handling
 7508 subsystem), but the data elements identified for a particular subsystem that are going to be used by
 7509 another subsystem need to be covered in this discussion. Any control relationships between
 7510 subsystems (e.g., a subsystem responsible for configuring a rule base for a firewall system and the
 7511 subsystem that actually implements these rules) should also be described.

7512 It should be noted while the developer should characterise all interactions between subsystems,
7513 the evaluators need to use their own judgement in assessing the completeness of the description. If
7514 the reason for an interaction is unclear, or if there are SFR-related interactions (discovered, for
7515 instance, in examining the module-level documentation) that do not appear to be described, the
7516 evaluator ensures that this information is provided by the developer. However, if the evaluator can
7517 determine that interactions among a particular set of subsystems, while incompletely described by
7518 the developer, and a complete description will not aid in understanding the overall functionality
7519 nor security functionality provided by the TSF, then the evaluator may choose to consider the
7520 description sufficient, and not pursue completeness for its own sake.

7521 ADV_TDS.5.6C *The design shall provide a mapping from the subsystems of the TSF to the modules*
7522 *of the TSF.*

7523 **11.8.5.4.9 Work unit ADV_TDS.5-9**

7524 The evaluator **shall examine** the TOE design to determine that the mapping between the
7525 subsystems of the TSF and the modules of the TSF is complete.

7526 If the design is presented solely in terms of modules, then this work unit is considered satisfied.

7527 For TOEs that are complex enough to warrant a subsystem-level description of the TSF in addition
7528 to the modular description, the developer provides a simple mapping showing how the modules of
7529 the TSF are allocated to the subsystems. This will provide the evaluator a guide in performing their
7530 module-level assessment. To determine completeness, the evaluator examines each mapping and
7531 determines that all subsystems map to at least one module, and that all modules map to exactly one
7532 subsystem.

7533 **11.8.5.4.10 Work unit ADV_TDS.5-10**

7534 The evaluator **shall examine** the TOE design to determine that the mapping between the
7535 subsystems of the TSF to the modules of the TSF is accurate.

7536 If the design is presented solely in terms of modules, then this work unit is considered satisfied.

7537 For TOEs that are complex enough to warrant a subsystem-level description of the TSF in addition
7538 to the modular description, the developer provides a simple mapping showing how the modules of
7539 the TSF are allocated to the subsystems. This will provide the evaluator a guide in performing their
7540 module-level assessment. The evaluator may choose to check the accuracy of the mapping in
7541 conjunction with performing other work units. An “inaccurate” mapping is one where the module
7542 is mistakenly associated with a subsystem where its functions are not used within the subsystem.
7543 Because the mapping is intended to be a guide supporting more detailed analysis, the evaluator is
7544 cautioned to apply appropriate effort to this work unit. Expending extensive evaluator resources
7545 verifying the accuracy of the mapping is not necessary. Inaccuracies that lead to mis-
7546 understandings related to the design that are uncovered as part of this or other work units are the
7547 ones that should be associated with this work unit and corrected.

7548 ADV_TDS.5.7C *The design shall provide a semiformal description of each module in terms of its*
7549 *purpose, interaction, interfaces, return values from those interfaces, and called interfaces to other*
7550 *modules, supported by informal, explanatory text where appropriate.*

7551 **11.8.5.4.11 Work unit ADV_TDS.5-11**

7552 The evaluator **shall examine** the TOE design to determine that the semiformal description of the
7553 purpose of each module, and its relationship with other modules is complete and accurate.

7554 The developer may designate modules as SFR-enforcing, SFR-supporting, and SFR non-interfering,
7555 but these “tags” are used only to describe the amount and type of information the developer must
7556 provide, and can be used to limit the amount of information the developer has to develop if their

7557 engineering process does not produce the documentation required. Whether the modules have
 7558 been categorised by the developer or not, it is the evaluator's responsibility to determine that the
 7559 modules have the appropriate information for their role (SFR-enforcing, etc.) in the TOE, and to
 7560 obtain the appropriate information from the developer should the developer fail to provide the
 7561 required information for a particular module.

7562 The purpose of a module provides a description indicating what function the module is fulfilling. A
 7563 word of caution to the evaluator is in order. The focus of this work unit should be to provide the
 7564 evaluator an understanding of how the module works so that determinations can be made about
 7565 the soundness of the implementation of the SFRs, as well as to support architectural analysis
 7566 performed for ADV_ARC subsystems. As long as the evaluator has a sound understanding of the
 7567 module's operation, and its relationship to other modules and the TOE as a whole, the evaluator
 7568 should consider the objective of the work achieved and not engage in a documentation exercise for
 7569 the developer (by requiring, for example, a complete algorithmic description for a self-evident
 7570 implementation representation).

7571 Because the modules are at such a low level, it may be difficult determine completeness and
 7572 accuracy impacts from other documentation, such as operational user guidance, the functional
 7573 specification, the TSF internals, or the security architecture description. However, the evaluator
 7574 uses the information present in those documents to the extent possible to help ensure that the
 7575 purpose is accurately and completely described. This analysis can be aided by the analysis
 7576 performed for the work units for the ADV_TDS.5.8C element, which maps the TSFI in the functional
 7577 specification to the modules of the TSF.

7578 **11.8.5.4.12 Work unit ADV_TDS.5-12**

7579 The evaluator *shall examine* the TOE design to determine that the semiformal description of the
 7580 interfaces presented by each module contain an accurate and complete description of the related
 7581 parameters, the invocation conventions for each interface, and any values returned directly by the
 7582 interface.

7583 The interfaces of a module are those interfaces used by other modules as a means to invoke the
 7584 operations provided, and to provide inputs to or receive outputs from the module. The purpose in
 7585 the specification of these interfaces is to permit the exercise of them during testing. Inter-module
 7586 interfaces that are not SFR-related need not be specified or described, since they are not a factor in
 7587 testing. Likewise, other internal interfaces that are not a factor in traversing SFR-related paths of
 7588 execution (such as those internal paths that are fixed).

7589 SFR-related interfaces are all interfaces that are called directly or indirectly from SFR-enforcing
 7590 modules. Those interfaces need to be described with all the parameter used in such a call. This
 7591 allows the evaluator to understand the purpose of the call in the context of operation of the SFR-
 7592 enforcing modules.

7593 SFR-related interfaces are described in terms of how they are invoked, and any values that are
 7594 returned. This description would include a list of parameters, and descriptions of these parameters.
 7595 Note that global data would also be considered parameters if used by the module (either as inputs
 7596 or outputs) when invoked. If a parameter were expected to take on a set of values (e.g., a "flag"
 7597 parameter), the complete set of values the parameter could take on, that would have an effect on
 7598 module processing, would be specified. Likewise, parameters representing data structures are
 7599 described such that each field of the data structure is identified and described. Note that different
 7600 programming languages may have additional "interfaces" that would be non-obvious; an example
 7601 would be operator/function overloading in C++. This "implicit interface" in the class description
 7602 would also be described as part of the low-level TOE design. Note that although a module could
 7603 present only one interface, it is more common that a module presents a small set of related
 7604 interfaces.

7605 In terms of the assessment of parameters (inputs and outputs) to a module, any use of global data
 7606 must also be considered. A module "uses" global data if it either reads or writes the data. In order

7607 to assure the description of such parameters (if used) is complete, the evaluator uses other
7608 information provided about the module in the TOE design (interfaces, algorithmic description, etc.),
7609 as well as the description of the particular set of global data assessed in work unit ADV_TDS.5-10.
7610 For instance, the evaluator could first determine the processing the module performs by examining
7611 its function and interfaces presented (particularly the parameters of the interfaces). They could
7612 then check to see if the processing appears to “touch” any of the global data areas identified in the
7613 TDS design. The evaluator then determines that, for each global data area that appears to be
7614 “touched”, that global data area is listed as a means of input or output by the module the evaluator
7615 is examining.

7616 Invocation conventions are a programming-reference-type description that one could use to
7617 correctly invoke a module's interface if one were writing a program to make use of the module's
7618 functionality through that interface. This includes necessary inputs and outputs, including any set-
7619 up that may need to be performed with respect to global variables.

7620 Values returned through the interface refer to values that are either passed through parameters or
7621 messages; values that the function call itself returns in the style of a “C” program function call; or
7622 values passed through global means (such as certain error routines in *ix-style operating systems).

7623 In order to assure the description is complete, the evaluator uses other information provided about
7624 the module in the TOE design (e.g., algorithmic description, global data used) to ensure that it
7625 appears all data necessary for performing the functions of the module is presented to the module,
7626 and that any values that other modules expect the module under examination to provide are
7627 identified as being returned by the module. The evaluator determines accuracy by ensuring that
7628 the description of the processing matches the information listed as being passed to or from an
7629 interface.

7630 ADV_TDS.5.8C *The mapping shall demonstrate that all TSFIs trace to the behaviour described in*
7631 *the TOE design that they invoke.*

7632 **11.8.5.4.13 Work unit ADV_TDS.5-13**

7633 The evaluator **shall examine** the TOE design to determine that it contains a complete and accurate
7634 mapping from the TSFI described in the functional specification to the modules of the TSF
7635 described in the TOE design.

7636 The modules described in the TOE design provide a description of the implementation of the TSF.
7637 The TSFI provide a description of how the implementation is exercised. The evidence from the
7638 developer identifies the module that is initially invoked when an operation is requested at the TSFI,
7639 and identify the chain of modules invoked up to the module that is primarily responsible for
7640 implementing the functionality. However, a complete call tree for each TSFI is not required for this
7641 work unit. The cases in which more than one module would have to be identified are where there
7642 are “entry point” modules or wrapper modules that have no functionality other than conditioning
7643 inputs or de-multiplexing an input. Mapping to one of these modules would not provide any useful
7644 information to the evaluator.

7645 The evaluator assesses the completeness of the mapping by ensuring that all of the TSFI map to at
7646 least one module. The verification of accuracy is more complex.

7647 The first aspect of accuracy is that each TSFI is mapped to a module at the TSF boundary. This
7648 determination can be made by reviewing the module description and its interfaces/interactions.
7649 The next aspect of accuracy is that each TSFI identifies a chain of modules between the initial
7650 module identified and a module that is primarily responsible for implementing the function
7651 presented at the TSF. Note that this may be the initial module, or there may be several modules,
7652 depending on how much pre-conditioning of the inputs is done. It should be noted that one
7653 indicator of a pre-conditioning module is that it is invoked for a large number of the TSFI, where
7654 the TSFI are all of similar type (e.g., system call). The final aspect of accuracy is that the mapping
7655 makes sense. For instance, mapping a TSFI dealing with access control to a module that checks

7656 passwords is not accurate. The evaluator should again use judgement in making this determination.
 7657 The goal is that this information aids the evaluator in understanding the system and
 7658 implementation of the SFRs, and ways in which entities at the TSF boundary can interact with the
 7659 TSF. The bulk of the assessment of whether the SFRs are described accurately by the modules is
 7660 performed in other work units.

7661 **11.8.5.4.14 Work unit ADV_TDS.5-14**

7662 The evaluator shall examine the TOE security functional requirements and the TOE design, to
 7663 determine that all ST security functional requirements are covered by the TOE design. The
 7664 evaluator may construct a map between the TOE security functional requirements and the TOE
 7665 design. This map will likely be from a functional requirement to a set of subsystems, and later to
 7666 modules. Note that this map may have to be at a level of detail below the component or even
 7667 element level of the requirements, because of operations (assignments, refinements, selections)
 7668 performed on the functional requirement by the ST author.

7669 For example, the FDP_ACC.1 Subset access control component contains an element with
 7670 assignments. If the ST contained, for instance, ten rules in the FDP_ACC.1 Subset access
 7671 control assignment, and these ten rules were implemented in specific places within fifteen
 7672 modules, it would be inadequate for the evaluator to map FDP_ACC.1 Subset access control to
 7673 one subsystem and claim the work unit had been completed. Instead, the evaluator would map
 7674 FDP_ACC.1 Subset access control (rule 1) to modules x, y and z of subsystem A; FDP_ACC.1
 7675 Subset access control (rule 2) to x, p, and q of subsystem A; etc.

7676 **11.8.5.4.15 Work unit ADV_TDS.5-15**

7677 The evaluator shall examine the TOE design to determine that it is an accurate instantiation of all
 7678 security functional requirements.

7679 The evaluator may construct a map between the TOE security functional requirements and the TOE
 7680 design. This map will likely be from a functional requirement to a set of subsystems and modules.
 7681 Note that this map may have to be at a level of detail below the component or even element level of
 7682 the requirements, because of operations (assignments, refinements, selections) performed on the
 7683 functional requirement by the ST author.

7684 As an example, if the ST requirements specified a role-based access control mechanism, the
 7685 evaluator would first identify the subsystems, and modules that contribute to this mechanism's
 7686 implementation. This could be done by in-depth knowledge or understanding of the TOE design or
 7687 by work done in the previous work unit. Note that this trace is only to identify the subsystems, and
 7688 modules, and is not the complete analysis.

7689 The next step would be to understand what mechanism the subsystems, and modules implemented.
 7690 For instance, if the design described an implementation of access control based on UNIX-style
 7691 protection bits, the design would not be an accurate instantiation of those access control
 7692 requirements present in the ST example used above. If the evaluator could not determine that the
 7693 mechanism was accurately implemented because of a lack of detail, the evaluator would have to
 7694 assess whether all of the SFR-enforcing subsystems and modules have been identified, or if
 7695 adequate detail had been provided for those subsystems and modules.

7696 **11.8.6 Evaluation of sub-activity (ADV_TDS.6)**

7697 There is no general guidance; the scheme should be consulted for guidance on this sub-activity.

7698 **11.9 Composite design compliance (ADV_COMP)**

7699 The composite-specific work units defined in this chapter are intended to be integrated as
 7700 refinements to the evaluation activities of the ADV class listed in the following table. The other
 7701 activities of ADV class do not require composite-specific work units.

CC assurance family	Evaluation activity	Evaluation work unit	Composite-specific work unit
ADV_ARC	ADV_ARC.1.1E	ADV_ARC.1.1C/ ADV_ARC.1-1	ADV_COMP.1-1
ADV_IMP	ADV_IMP.1.1E	ADV_IMP.1.1C/ ADV_IMP.1-1	ADV_COMP.1-1
ADV_TDS	ADV_TDS.1.2E	ADV_TDS.1-7	ADV_COMP.1-1

7702 NB: If the level of the assurance requirement chosen is higher than those identified in this table, the
7703 composite-specific work unit is also applicable.

7704 **11.9.1 Evaluation of sub-activity (ADV_COMP.1)**

7705 **11.9.1.1 Objectives**

7706 The aim of this activity is to determine whether the requirements on the application, imposed by
7707 the underlying platform, are fulfilled in the composite product.

7708 **11.9.1.2 Application notes**

7709 The requirements on the application, imposed by the underlying platform, can be formulated in
7710 the relevant certification report (e.g. in form of constraints and recommendations), user guidance
7711 and ETR_COMP (in form of observations and recommendations) for the platform. The developer of
7712 the composite product shall regard each of these sources, if available and implement the composite
7713 product in such a way that the applicable requirements are fulfilled.

7714 The TSF of the composite product is represented at various levels of abstraction in the families of
7715 the development class ADV. Experiential, the appropriate levels of design representation for
7716 examining, whether the requirements of the platform are fulfilled by the composite product, are
7717 the TOE design (ADV_TDS), security architecture (ADV_ARC) and the implementation (ADV_IMP).
7718 In case, these design representation levels are not available (e.g. due to the assurance package
7719 chosen is EAL1), the current activity is not applicable (see the next paragraph for the reason)

7720 Due to the definition of the composite TOE the interface between the underlying platform and the
7721 application is the internal one, hence, a functional specification (ADV_FSP) as representation level
7722 is not appropriate for analysing the design compliance.

7723 Security architecture ADV_ARC as assurance family is dedicated to ensure that integrative security
7724 services like domain separation, self-protection and non-bypassability properly work. It is
7725 impossible and not the sense of the composite evaluation to have an insight into the architectural
7726 internals of the underlying platform (it is a matter of the platform evaluation). What the Composite
7727 Evaluator has to do in the context of ADV_ARC is

7728 i. to determine whether the application uses services of the underlying platform within its
7729 own Composite-ST to provide domain separation, self-protection, non-bypassability and
7730 protected start-up; if no, there is no further composite activities for ADV_ARC; if yes, then

7731 ii. the evaluator has to determine, whether the application uses these platform-services in
7732 an appropriate/secure way

7733 Since consistency of the composite product security policy has already been considered in the
7734 context of the Security Target in the assurance family ASE_COMP there is no necessity to consider
7735 non-contradictoriness of the security policy model (ADV_SPM) of the composite TOE and the
7736 security policy model of the underlying platform.

7737 **11.9.1.3 Action ADV_COMP.1.1E**

7738 The evaluator shall confirm that the rationale for design compliance is complete, coherent, and
7739 internally consistent.

7740 *ADV_COMP.1.1C The design compliance justification shall provide a rationale for design compliance –*
7741 *on an appropriate representation level – of how the requirements on the application, imposed by the*
7742 *underlying platform, are fulfilled in the composite product.*

7743 **11.9.1.3.1 Work unit ADV_COMP.1-1**

7744 The evaluator shall examine the rationale for design compliance to determine that all applicable
7745 requirements on the application, imposed by the underlying platform, are fulfilled by the
7746 composite product.

7747 In order to perform this work unit the evaluator shall use the rationale for design compliance as
7748 well as the TSF representation on the ADV_TDS, ADV_ARC and ADV_IMP levels on the one side and
7749 the input of the platform developer in form of the certification report, guidance and ETR_COMP on
7750 the other side. The evaluator shall analyse which platform requirements are applicable for the
7751 current composite product, based on the identified RP-SFR-MECH and RP-SFR-SERV. The
7752 evaluator shall compare each of the applicable requirements with the actual specification and/or
7753 implementation of the composite product and determine, for each requirement, whether it is
7754 fulfilled. As result, the evaluator confirms or disproves the rationale for design compliance.

7755 For example, platform guidance may require the application to perform a special start-up
7756 sequence testing the current state of the platform and initialising its self-protection mechanisms.
7757 Such information might be found in the description of secure architecture ADV_ARC of the
7758 composite TOE; see also the Application Note above.

7759 A second example, platform guidance may require the application to perform a DFA check on the
7760 DES operation, while the application is implementing BAC in an e-passport MRTD. The ADV_ARC
7761 will explain whether the platform guidance is followed up or not, and in case that the
7762 requirements in the platform guidance are not followed a corresponding reasoning will be
7763 provided. The arguments of the developer explain why a non-compliance will not introduce
7764 vulnerabilities.

7765 The appropriate representation level (ADV_TDS, ADV_ARC and/or ADV_IMP), what the analysis is
7766 being performed on, can be chosen and mixed flexibly depending on the concrete composite TOE
7767 and the requirement in question. Where it is not self-explaining, the evaluator shall justify why the
7768 representation level chosen is appropriate.

7769 The evaluator activities in the context of this work unit can be spread over different single
7770 evaluation aspects (e.g. over ADV_TDS and ADV_IMP). In this case the evaluator performs the
7771 partial activity in the context of the corresponding single evaluation aspect. Then the notation for
7772 this work unit shall be ADV_COMP.1-1-TDS, ADV_COMP.1-1-ARC and ADV_COMP.1-1-IMP,
7773 respectively.

7774 If the assurance package chosen does not contain the families ADV_TDS, ADV_ARC or ADV_IMP (e.g.
7775 EAL1), this work unit is not applicable (cf. Application Note above).

7776 The result of this work unit shall be integrated to the result of ADV_TDS.1-2E/ ADV_TDS.1-7,
7777 ADV_ARC.1.1E/ ADV_ARC.1.1C/ ADV_ARC.1-1, ADV_IMP.1.1E/ ADV_IMP.1.1C/ ADV_IMP.1-1 (or
7778 the equivalent higher components if a higher assurance level is selected).

7779

7780 **12 Class AGD: Guidance documents**

7781 **12.1 Introduction**

7782 The purpose of the guidance document activity is to judge the adequacy of the documentation
7783 describing how the user can handle the TOE in a secure manner. Such documentation should take
7784 into account the various types of users (e.g. those who accept, install, administrate or operate the
7785 TOE) whose incorrect actions could adversely affect the security of the TOE or of their own data.

7786 The guidance documents class is subdivided into two families which are concerned firstly with the
7787 preparative procedures (all that has to be done to transform the delivered TOE into its evaluated
7788 configuration in the environment as described in the ST, i.e. accepting and installing the TOE) and
7789 secondly with the operational user guidance (all that has to be done during the operation of the
7790 TOE in its evaluated configuration, i.e. operation and administration).

7791 **12.2 Application notes**

7792 The guidance documents activity applies to those functions and interfaces which are related to the
7793 security of the TOE. The secure configuration of the TOE is described in the ST.

7794 **12.3 Operational user guidance (AGD_OPE)**

7795 **12.3.1 Evaluation of sub-activity (AGD_OPE.1)**

7796 **12.3.1.1 Objectives**

7797 The objectives of this sub-activity are to determine whether the user guidance describes for each
7798 user role the security functionality and interfaces provided by the TSF, provides instructions and
7799 guidelines for the secure use of the TOE, addresses secure procedures for all modes of operation,
7800 facilitates prevention and detection of insecure TOE states, or whether it is misleading or
7801 unreasonable.

7802 **12.3.1.2 Input**

7803 The evaluation evidence for this sub-activity is:

- 7804 a) the ST;
- 7805 b) the functional specification;
- 7806 c) the TOE design, if applicable;
- 7807 d) the user guidance;

7808 **12.3.1.3 Action AGD_OPE.1.1E**

7809 ISO/IEC 15408-3 AGD_OPE.1.1C: *The operational user guidance shall describe, for each user role, the*
7810 *user-accessible functions and privileges that should be controlled in a secure processing environment,*
7811 *including appropriate warnings.*

7812 **12.3.1.3.1 Work unit AGD_OPE.1-1**

7813 The evaluator **shall examine** the operational user guidance to determine that it describes, for each
7814 user role, the user-accessible functions and privileges that should be controlled in a secure
7815 processing environment, including appropriate warnings.

- 7816 The configuration of the TOE may allow different user roles to have dissimilar privileges in making
7817 use of the different functions of the TOE. This means that some users are authorised to perform
7818 certain functions, while other users may not be so authorised. These functions and privileges
7819 should be described, for each user role, by the user guidance.
- 7820 The user guidance identifies, for each user role, the functions and privileges that must be
7821 controlled, the types of commands required for them, and the reasons for such commands. The
7822 user guidance should contain warnings regarding the use of these functions and privileges.
7823 Warnings should address expected effects, possible side effects, and possible interactions with
7824 other functions and privileges.
- 7825 ISO/IEC 15408-3 AGD_OPE.1.2C: *The operational user guidance shall describe, for each user role,*
7826 *how to use the available interfaces provided by the TOE in a secure manner.*
- 7827 **12.3.1.3.2 Work unit AGD_OPE.1-2**
- 7828 The evaluator **shall examine** the operational user guidance to determine that it describes, for each
7829 user role, the secure use of the available interfaces provided by the TOE.
- 7830 The user guidance should provide advice regarding effective use of the TSF (e.g. reviewing
7831 password composition practises, suggested frequency of user file backups, discussion on the effects
7832 of changing user access privileges).
- 7833 ISO/IEC 15408-3 AGD_OPE.1.3C: *The operational user guidance shall describe, for each user role, the*
7834 *available functions and interfaces, in particular all security parameters under the control of the user,*
7835 *indicating secure values as appropriate.*
- 7836 **12.3.1.3.3 Work unit AGD_OPE.1-3**
- 7837 The evaluator **shall examine** the operational user guidance to determine that it describes, for each
7838 user role, the available security functionality and interfaces, in particular all security parameters
7839 under the control of the user, indicating secure values as appropriate.
- 7840 The user guidance should contain an overview of the security functionality that is visible at the
7841 user interfaces.
- 7842 The user guidance should identify and describe the purpose, behaviour, and interrelationships of
7843 the security interfaces and functionality.
- 7844 For each user-accessible interface, the user guidance should:
- 7845 a) describe the method(s) by which the interface is invoked (e.g. command-line,
7846 programming-language system call, menu selection, command button);
- 7847 b) describe the parameters to be set by the user, their particular purposes, valid and default
7848 values, and secure and insecure use settings of such parameters, both individually or in
7849 combination;
- 7850 c) describe the immediate TSF response, message, or code returned.
- 7851 The evaluator should consider the functional specification and the ST to determine that the TSF
7852 described in these documents is consistent to the operational user guidance. The evaluator has to
7853 ensure that the operational user guidance is complete to allow the secure use through the TSFI
7854 available to all types of human users. The evaluator may, as an aid, prepare an informal mapping
7855 between the guidance and these documents. Any omissions in this mapping may indicate
7856 incompleteness.

7857 ISO/IEC 15408-3 AGD_OPE.1.4C: *The operational user guidance shall, for each user role, clearly*
7858 *present each type of security-relevant event relative to the user-accessible functions that need to be*
7859 *performed, including changing the security characteristics of entities under the control of the TSF.*

7860 **12.3.1.3.4 Work unit AGD_OPE.1-4**

7861 The evaluator **shall examine** the operational user guidance to determine that it describes, for each
7862 user role, each type of security-relevant event relative to the user functions that need to be
7863 performed, including changing the security characteristics of entities under the control of the TSF
7864 and operation following failure or operational error.

7865 All types of security-relevant events are detailed for each user role, such that each user knows
7866 what events may occur and what action (if any) they may have to take in order to maintain security.
7867 Security-relevant events that may occur during operation of the TOE (e.g. audit trail overflow,
7868 system crash, updates to user records, such as when a user account is removed when the user
7869 leaves the organisation) are adequately defined to allow user intervention to maintain secure
7870 operation.

7871 ISO/IEC 15408-3 AGD_OPE.1.5C: *The operational user guidance shall identify all possible modes of*
7872 *operation of the TOE (including operation following failure or operational error), their consequences*
7873 *and implications for maintaining secure operation.*

7874 **12.3.1.3.5 Work unit AGD_OPE.1-5**

7875 The evaluator **shall examine** the operational user guidance and other evaluation evidence to
7876 determine that the guidance identifies all possible modes of operation of the TOE (including, if
7877 applicable, operation following failure or operational error), their consequences and implications
7878 for maintaining secure operation.

7879 Other evaluation evidence, particularly the functional specification, provide an information source
7880 that the evaluator should use to determine that the guidance contains sufficient guidance
7881 information.

7882 If test documentation is included in the assurance package, then the information provided in this
7883 evidence can also be used to determine that the guidance contains sufficient guidance
7884 documentation. The detail provided in the test steps can be used to confirm that the guidance
7885 provided is sufficient for the use and administration of the TOE.

7886 The evaluator should focus on a single human visible TSFI at a time, comparing the guidance for
7887 securely using the TSFI with other evaluation evidence, to determine that the guidance related to
7888 the TSFI is sufficient for the secure usage (i.e. consistent with the SFRs) of that TSFI. The evaluator
7889 should also consider the relationships between interfaces, searching for potential conflicts.

7890 ISO/IEC 15408-3 AGD_OPE.1.6C: *The operational user guidance shall, for each user role, describe the*
7891 *security measures to be followed in order to fulfil the security objectives for the operational*
7892 *environment as described in the ST.*

7893 **12.3.1.3.6 Work unit AGD_OPE.1-6**

7894 The evaluator **shall examine** the operational user guidance to determine that it describes, for each
7895 user role, the security measures to be followed in order to fulfil the security objectives for the
7896 operational environment as described in the ST.

7897 The evaluator analyses the security objectives for the operational environment in the ST and
7898 determines that for each user role, the relevant security measures are described appropriately in
7899 the user guidance.

- 7900 The security measures described in the user guidance should include all relevant external
7901 procedural, physical, personnel and connectivity measures.
- 7902 Note that those measures relevant for secure installation of the TOE are examined in Preparative
7903 procedures (AGD_PRE).
- 7904 ISO/IEC 15408-3 AGD_OPE.1.7C: *The operational user guidance shall be clear and reasonable.*
- 7905 **12.3.1.3.7 Work unit AGD_OPE.1-7**
- 7906 The evaluator *shall examine* the operational user guidance to determine that it is clear.
- 7907 The guidance is unclear if it can reasonably be misconstrued by an administrator or user, and used
7908 in a way detrimental to the TOE, or to the security provided by the TOE.
- 7909 **12.3.1.3.8 Work unit AGD_OPE.1-8**
- 7910 The evaluator *shall examine* the operational user guidance to determine that it is reasonable.
- 7911 The guidance is unreasonable if it makes demands on the TOE's usage or operational environment
7912 that are inconsistent with the ST or unduly onerous to maintain security.
- 7913 **12.4 Preparative procedures (AGD_PRE)**
- 7914 **12.4.1 Evaluation of sub-activity (AGD_PRE.1)**
- 7915 **12.4.1.1 Objectives**
- 7916 The objective of this sub-activity is to determine whether the procedures and steps for the secure
7917 preparation of the TOE have been documented and result in a secure configuration.
- 7918 **12.4.1.2 Input**
- 7919 The evaluation evidence for this sub-activity is:
- 7920 a) the ST;
- 7921 b) the TOE including its preparative procedures;
- 7922 c) the description of developer's delivery procedures, if applicable;
- 7923 **12.4.1.3 Application notes**
- 7924 The preparative procedures refer to all acceptance and installation procedures, that are necessary
7925 to progress the TOE to the secure configuration as described in the ST.
- 7926 **12.4.1.4 Action AGD_PRE.1.1E**
- 7927 ISO/IEC 15408-3 AGD_PRE.1.1C: *The preparative procedures shall describe all the steps necessary*
7928 *for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.*
- 7929 **12.4.1.4.1 Work unit AGD_PRE.1-1**
- 7930 The evaluator *shall examine* the provided acceptance procedures to determine that they describe
7931 the steps necessary for secure acceptance of the TOE in accordance with the developer's delivery
7932 procedures.

7933 If it is not anticipated by the developer's delivery procedures that acceptance procedures will or
7934 can be applied, this work unit is not applicable, and is therefore considered to be satisfied.

7935 The acceptance procedures should include as a minimum, that the user has to check that all parts
7936 of the TOE as indicated in the ST have been delivered in the correct version.

7937 The acceptance procedures should reflect the steps the user has to perform in order to accept the
7938 delivered TOE that are implied by the developer's delivery procedures.

7939 The acceptance procedures should provide detailed information about the following, if applicable:

7940 a) making sure that the delivered TOE is the complete evaluated instance;

7941 b) detecting modification/masquerading of the delivered TOE.

7942 ISO/IEC 15408-3 AGD_PRE.1.2C: *The preparative procedures shall describe all the steps necessary*
7943 *for secure installation of the TOE and for the secure preparation of the operational environment in*
7944 *accordance with the security objectives for the operational environment as described in the ST.*

7945 **12.4.1.4.2 Work unit AGD_PRE.1-2**

7946 The evaluator **shall examine** the provided installation procedures to determine that they describe
7947 the steps necessary for secure installation of the TOE and the secure preparation of the operational
7948 environment in accordance with the security objectives in the ST.

7949 If it is not anticipated that installation procedures will or can be applied (e.g. because the TOE may
7950 already be delivered in an operational state), this work unit is not applicable, and is therefore
7951 considered to be satisfied.

7952 The installation procedures should provide detailed information about the following, if applicable:

7953 a) minimum system requirements for secure installation;

7954 b) requirements for the operational environment in accordance with the security objectives
7955 provided by the ST;

7956 c) the steps the user has to perform in order to get to an operational TOE being
7957 commensurate with its evaluated configuration. Such a description shall include - for
7958 each step - a clear scheme for the decision on the next step depended on success, failure
7959 or problems at the current step;

7960 d) changing the installation specific security characteristics of entities under the control of
7961 the TSF (for example parameters, settings, passwords);

7962 e) handling exceptions and problems.

7963 **12.4.1.5 Action AGD_PRE.1.2E**

7964 **12.4.1.5.1 Work unit AGD_PRE.1-3**

7965 The evaluator **shall perform** all user procedures necessary to prepare the TOE to determine that
7966 the TOE and its operational environment can be prepared securely using only the supplied
7967 preparative procedures.

7968 Preparation requires the evaluator to advance the TOE from a deliverable state to the state in
7969 which it is operational, including acceptance and installation of the TOE, and enforcing the SFRs
7970 consistent with the security objectives for the TOE specified in the ST.

7971 The evaluator should follow only the developer's procedures and may perform the activities that
 7972 customers are usually expected to perform to accept and install the TOE, using the supplied
 7973 preparative procedures only. Any difficulties encountered during such an exercise may be
 7974 indicative of incomplete, unclear or unreasonable guidance.

7975 This work unit may be performed in conjunction with the evaluation activities under Independent
 7976 testing (ATE_IND).

7977 If it is known that the TOE will be used as a dependent component for a composed TOE evaluation,
 7978 then the evaluator should ensure that the operational environment is satisfied by the base
 7979 component used in the composed TOE.

7980 **13 Class ALC: Life-cycle support**

7981 **13.1 Introduction**

7982 The purpose of the life-cycle support activity is to determine the adequacy of the security
 7983 procedures that the developer uses during the development and maintenance of the TOE. These
 7984 procedures include the life-cycle model used by the developer, the configuration management, the
 7985 security measures used throughout TOE development, the tools used by the developer throughout
 7986 the life-cycle of the TOE, the handling of security flaws, and the delivery activity.

7987 Poorly controlled development and maintenance of the TOE can result in vulnerabilities in the
 7988 implementation. Conformance to a defined life-cycle model can help to improve controls in this
 7989 area. A measurable life-cycle model used for the TOE can remove ambiguity in assessing the
 7990 development progress of the TOE.

7991 The purpose of the configuration management activity is to assist the consumer in identifying the
 7992 evaluated TOE, to ensure that configuration items are uniquely identified, and the adequacy of the
 7993 procedures that are used by the developer to control and track changes that are made to the TOE.
 7994 This includes details on what changes are tracked, how potential changes are incorporated, and the
 7995 degree to which automation is used to reduce the scope for error.

7996 Developer security procedures are intended to protect the TOE and its associated design
 7997 information from interference or disclosure. Interference in the development process may allow
 7998 the deliberate introduction of vulnerabilities. Disclosure of design information may allow
 7999 vulnerabilities to be more easily exploited. The adequacy of the procedures will depend on the
 8000 nature of the TOE and the development process.

8001 The use of well-defined development tools and the application of implementation standards by the
 8002 developer and by third parties involved in the development process help to ensure that
 8003 vulnerabilities are not inadvertently introduced during refinement.

8004 The flaw remediation activity is intended to track security flaws, to identify corrective actions, and
 8005 to distribute the corrective action information to TOE users.

8006 The purpose of the delivery activity is to judge the adequacy of the documentation of the
 8007 procedures used to ensure that the TOE is delivered to the consumer without modification.

8008 **ALC_TDA**

8009 Appropriate detailed comment/changes are invited.

8010 **ALC_COMP**

8011 Appropriate detailed comment/changes are invited.

8012

8013 **13.2 CM capabilities (ALC_CMC)**

8014 **13.2.1 Evaluation of sub-activity (ALC_CMC.1)**

8015 **13.2.1.1 Objectives**

8016 The objectives of this sub-activity are to determine whether the developer has clearly identified the
8017 TOE.

8018 **13.2.1.2 Input**

8019 The evaluation evidence for this sub-activity is:

8020 a) the ST;

8021 b) the TOE suitable for testing.

8022 **13.2.1.3 Action ALC_CMC.1.1E**

8023 ISO/IEC 15408-3 ALC_CMC.1.1C: *The TOE shall be labelled with its unique reference.*

8024 **13.2.1.3.1 Work unit ALC_CMC.1-1**

8025 The evaluator **shall check** that the TOE provided for evaluation is labelled with its reference.

8026 The evaluator should ensure that the TOE contains the unique reference which is stated in the ST.
8027 This could be achieved through labelled packaging or media, or by a label displayed by the
8028 operational TOE. This is to ensure that it would be possible for consumers to identify the TOE (e.g.
8029 at the point of purchase or use).

8030 The TOE may provide a method by which it can be easily identified. For example, a software TOE
8031 may display its name and version number during the start up routine, or in response to a command
8032 line entry. A hardware or firmware TOE may be identified by a part number physically stamped on
8033 the TOE.

8034 Alternatively, the unique reference provided for the TOE may be the combination of the unique
8035 reference of each component from which the TOE is comprised (e.g. in the case of a composed TOE).

8036 **13.2.1.3.2 Work unit ALC_CMC.1-2**

8037 The evaluator **shall check** that the TOE references used are consistent.

8038 If the TOE is labelled more than once then the labels have to be consistent. For example, it should
8039 be possible to relate any labelled guidance documentation supplied as part of the TOE to the
8040 evaluated operational TOE. This ensures that consumers can be confident that they have purchased
8041 the evaluated version of the TOE, that they have installed this version, and that they have the
8042 correct version of the guidance to operate the TOE in accordance with its ST.

8043 The evaluator also verifies that the TOE reference is consistent with the ST.

8044 If this work unit is applied to a composed TOE, the following will apply. The composed IT TOE will
8045 not be labelled with its unique (composite) reference, but only the individual components will be
8046 labelled with their appropriate TOE reference. It would require further development for the IT TOE
8047 to be labelled, i.e. during start-up and/or operation, with the composite reference. If the composed
8048 TOE is delivered as the constituent component TOEs, then the TOE items delivered will not contain
8049 the composite reference. However, the composed TOE ST will include the unique reference for the
8050 composed TOE and will identify the components comprising the composed TOE through which the
8051 consumers will be able to determine whether they have the appropriate items.

8052 **13.2.2 Evaluation of sub-activity (ALC_CMC.2)**8053 **13.2.2.1 Objectives**

8054 The objectives of this sub-activity are to determine whether the developer uses a CM system that
8055 uniquely identifies all configuration items.

8056 **13.2.2.2 Input**

8057 The evaluation evidence for this sub-activity is:

- 8058 a) the ST;
- 8059 b) the TOE suitable for testing;
- 8060 c) the configuration management documentation.

8061 **13.2.2.3 Application notes**

8062 This component contains an implicit evaluator action to determine that the CM system is being
8063 used. As the requirements here are limited to identification of the TOE and provision of a
8064 configuration list, this action is already covered by, and limited to, the existing work units. At
8065 Evaluation of sub-activity (ALC_CMC.3) the requirements are expanded beyond these two items,
8066 and more explicit evidence of operation is required.

8067 **13.2.2.4 Action ALC_CMC.2.1E**

8068 ISO/IEC 15408-3 ALC_CMC.2.1C: *The TOE shall be labelled with its unique reference.*

8069 **13.2.2.4.1 Work unit ALC_CMC.2-1**

8070 The evaluator ***shall check*** that the TOE provided for evaluation is labelled with its reference.

8071 The evaluator should ensure that the TOE contains the unique reference which is stated in the ST.
8072 This could be achieved through labelled packaging or media, or by a label displayed by the
8073 operational TOE. This is to ensure that it would be possible for consumers to identify the TOE (e.g.
8074 at the point of purchase or use).

8075 The TOE may provide a method by which it can be easily identified. For example, a software TOE
8076 may display its name and version number during the start up routine, or in response to a command
8077 line entry. A hardware or firmware TOE may be identified by a part number physically stamped on
8078 the TOE.

8079 Alternatively, the unique reference provided for the TOE may be the combination of the unique
8080 reference of each component from which the TOE is comprised (e.g. in the case of a composed TOE).

8081 **13.2.2.4.2 Work unit ALC_CMC.2-2**

8082 The evaluator ***shall check*** that the TOE references used are consistent.

8083 If the TOE is labelled more than once then the labels have to be consistent. For example, it should
8084 be possible to relate any labelled guidance documentation supplied as part of the TOE to the
8085 evaluated operational TOE. This ensures that consumers can be confident that they have purchased
8086 the evaluated version of the TOE, that they have installed this version, and that they have the
8087 correct version of the guidance to operate the TOE in accordance with its ST.

8088 The evaluator also verifies that the TOE reference is consistent with the ST.

8089 If this work unit is applied to a composed TOE, the following will apply. The composed IT TOE will
8090 not be labelled with its unique (composite) reference, but only the individual components will be
8091 labelled with their appropriate TOE reference. It would require further development for the IT TOE
8092 to be labelled, i.e. during start-up and/or operation, with the composite reference. If the composed
8093 TOE is delivered as the constituent component TOEs, then the TOE items delivered will not contain
8094 the composite reference. However, the composed TOE ST will include the unique reference for the
8095 composed TOE and will identify the components comprising the composed TOE through which the
8096 consumers will be able to determine whether they have the appropriate items.

8097 ISO/IEC 15408-3 ALC_CMC.2.2C: *The CM documentation shall describe the method used to uniquely*
8098 *identify the configuration items.*

8099 **13.2.2.4.3 Work unit ALC_CMC.2-3**

8100 The evaluator **shall examine** the method of identifying configuration items to determine that it
8101 describes how configuration items are uniquely identified.

8102 Procedures should describe how the status of each configuration item can be tracked throughout
8103 the life-cycle of the TOE. The procedures may be detailed in the CM plan or throughout the CM
8104 documentation. The information included should describe:

8105 a) the method how each configuration item is uniquely identified, such that it is possible to
8106 track versions of the same configuration item;

8107 b) the method how configuration items are assigned unique identifiers and how they are
8108 entered into the CM system;

8109 c) the method to be used to identify superseded versions of a configuration item.

8110 ISO/IEC 15408-3 ALC_CMC.2.3C: *The CM system shall uniquely identify all configuration items.*

8111 **13.2.2.4.4 Work unit ALC_CMC.2-4**

8112 The evaluator **shall examine** the configuration items to determine that they are identified in a way
8113 that is consistent with the CM documentation.

8114 Assurance that the CM system uniquely identifies all configuration items is gained by examining
8115 the identifiers for the configuration items. For both configuration items that comprise the TOE, and
8116 drafts of configuration items that are submitted by the developer as evaluation evidence, the
8117 evaluator confirms that each configuration item possesses a unique identifier in a manner
8118 consistent with the unique identification method that is described in the CM documentation.

8119 **13.2.3 Evaluation of sub-activity (ALC_CMC.3)**

8120 **13.2.3.1 Objectives**

8121 The objectives of this sub-activity are to determine whether the developer uses a CM system that
8122 uniquely identifies all configuration items, and whether the ability to modify these items is
8123 properly controlled.

8124 **13.2.3.2 Input**

8125 The evaluation evidence for this sub-activity is:

8126 a) the ST;

8127 b) the TOE suitable for testing;

- 8128 c) the configuration management documentation.
- 8129 **13.2.3.3 Action ALC_CMC.3.1E**
- 8130 ISO/IEC 15408-3 ALC_CMC.3.1C: *The TOE shall be labelled with its unique reference.*
- 8131 **13.2.3.3.1 Work unit ALC_CMC.3-1**
- 8132 The evaluator **shall check** that the TOE provided for evaluation is labelled with its reference.
- 8133 The evaluator should ensure that the TOE contains the unique reference which is stated in the ST.
8134 This could be achieved through labelled packaging or media, or by a label displayed by the
8135 operational TOE. This is to ensure that it would be possible for consumers to identify the TOE (e.g.
8136 at the point of purchase or use).
- 8137 The TOE may provide a method by which it can be easily identified. For example, a software TOE
8138 may display its name and version number during the start up routine, or in response to a command
8139 line entry. A hardware or firmware TOE may be identified by a part number physically stamped on
8140 the TOE.
- 8141 Alternatively, the unique reference provided for the TOE may be the combination of the unique
8142 reference of each component from which the TOE is comprised (e.g. in the case of a composed TOE).
- 8143 **13.2.3.3.2 Work unit ALC_CMC.3-2**
- 8144 The evaluator **shall check** that the TOE references used are consistent.
- 8145 If the TOE is labelled more than once then the labels have to be consistent. For example, it should
8146 be possible to relate any labelled guidance documentation supplied as part of the TOE to the
8147 evaluated operational TOE. This ensures that consumers can be confident that they have purchased
8148 the evaluated version of the TOE, that they have installed this version, and that they have the
8149 correct version of the guidance to operate the TOE in accordance with its ST.
- 8150 The evaluator also verifies that the TOE reference is consistent with the ST.
- 8151 If this work unit is applied to a composed TOE, the following will apply. The composed IT TOE will
8152 not be labelled with its unique (composite) reference, but only the individual components will be
8153 labelled with their appropriate TOE reference. It would require further development for the IT TOE
8154 to be labelled, i.e. during start-up and/or operation, with the composite reference. If the composed
8155 TOE is delivered as the constituent component TOEs, then the TOE items delivered will not contain
8156 the composite reference. However, the composed TOE ST will include the unique reference for the
8157 composed TOE and will identify the components comprising the composed TOE through which the
8158 consumers will be able to determine whether they have the appropriate items.
- 8159 ISO/IEC 15408-3 ALC_CMC.3.2C: *The CM documentation shall describe the method used to uniquely*
8160 *identify the configuration items.*
- 8161 **13.2.3.3.3 Work unit ALC_CMC.3-3**
- 8162 The evaluator **shall examine** the method of identifying configuration items to determine that it
8163 describes how configuration items are uniquely identified.
- 8164 Procedures should describe how the status of each configuration item can be tracked throughout
8165 the life-cycle of the TOE. The procedures may be detailed in the CM plan or throughout the CM
8166 documentation. The information included should describe:
- 8167 a) the method how each configuration item is uniquely identified, such that it is possible to
8168 track versions of the same configuration item;

- 8169 b) the method how configuration items are assigned unique identifiers and how they are
8170 entered into the CM system;
- 8171 c) the method to be used to identify superseded versions of a configuration item.
- 8172 ISO/IEC 15408-3 ALC_CMC.3.3C: *The CM system shall uniquely identify all configuration items.*
- 8173 **13.2.3.3.4 Work unit ALC_CMC.3-4**
- 8174 The evaluator **shall examine** the configuration items to determine that they are identified in a way
8175 that is consistent with the CM documentation.
- 8176 Assurance that the CM system uniquely identifies all configuration items is gained by examining
8177 the identifiers for the configuration items. For both configuration items that comprise the TOE, and
8178 drafts of configuration items that are submitted by the developer as evaluation evidence, the
8179 evaluator confirms that each configuration item possesses a unique identifier in a manner
8180 consistent with the unique identification method that is described in the CM documentation.
- 8181 ISO/IEC 15408-3 ALC_CMC.3.4C: *The CM system shall provide measures such that only authorised*
8182 *changes are made to the configuration items.*
- 8183 **13.2.3.3.5 Work unit ALC_CMC.3-5**
- 8184 The evaluator **shall examine** the CM access control measures described in the CM plan to
8185 determine that they are effective in preventing unauthorised access to the configuration items.
- 8186 The evaluator may use a number of methods to determine that the CM access control measures are
8187 effective. For example, the evaluator may exercise the access control measures to ensure that the
8188 procedures could not be bypassed. The evaluator may use the outputs generated by the CM system
8189 procedures required by **ALC_CMC.3.8C**. The evaluator may also witness a demonstration of the CM
8190 system to ensure that the access control measures employed are operating effectively.
- 8191 ISO/IEC 15408-3 ALC_CMC.3.5C: *The CM documentation shall include a CM plan.*
- 8192 **13.2.3.3.6 Work unit ALC_CMC.3-6**
- 8193 The evaluator **shall check** that the CM documentation provided includes a CM plan.
- 8194 The CM plan needs not to be a connected document, but it is recommended that there is a single
8195 document that describes where the various parts of the CM plan can be found. If the CM plan is no
8196 single document, the list in the following work unit gives hints regarding which context is expected.
- 8197 ISO/IEC 15408-3 ALC_CMC.3.6C: *The CM plan shall describe how the CM system is used for the*
8198 *development of the TOE.*
- 8199 **13.2.3.3.7 Work unit ALC_CMC.3-7**
- 8200 The evaluator **shall examine** the CM plan to determine that it describes how the CM system is used
8201 for the development of the TOE.
- 8202 The descriptions contained in a CM plan include, if applicable:
- 8203 a) all activities performed in the TOE development that are subject to configuration
8204 management procedures (e.g. creation, modification or deletion of a configuration item,
8205 data-backup, archiving);
- 8206 b) which means (e.g. CM tools, forms) have to be made available;

- 8207 c) the usage of the CM tools: the necessary details for a user of the CM system to be able to
8208 operate the CM tools correctly in order to maintain the integrity of the TOE;
- 8209 d) which other objects (development components, tools, assessment environments, etc) are
8210 taken under CM control;
- 8211 e) the roles and responsibilities of individuals required to perform operations on individual
8212 configuration items (different roles may be identified for different types of configuration
8213 items (e.g. design documentation or source code));
- 8214 f) how CM instances (e.g. change control boards, interface control working groups) are
8215 introduced and staffed;
- 8216 g) the description of the change management;
- 8217 h) the procedures that are used to ensure that only authorised individuals can make changes
8218 to configuration items;
- 8219 i) the procedures that are used to ensure that concurrency problems do not occur as a
8220 result of simultaneous changes to configuration items;
- 8221 j) the evidence that is generated as a result of application of the procedures. For example,
8222 for a change to a configuration item, the CM system might record a description of the
8223 change, accountability for the change, identification of all configuration items affected,
8224 status (e.g. pending or completed), and date and time of the change. This might be
8225 recorded in an audit trail of changes made or change control records;
- 8226 k) the approach to version control and unique referencing of TOE versions (e.g. covering the
8227 release of patches in operating systems, and the subsequent detection of their
8228 application).
- 8229 ISO/IEC 15408-3 ALC_CMC.3.7C: *The evidence shall demonstrate that all configuration items are*
8230 *being maintained under the CM system.*
- 8231 **13.2.3.3.8 Work unit ALC_CMC.3-8**
- 8232 The evaluator **shall check** that the configuration items identified in the configuration list are being
8233 maintained by the CM system.
- 8234 The CM system employed by the developer should maintain the integrity of the TOE. The evaluator
8235 should check that for each type of configuration item (e.g. design documents or source code
8236 modules) contained in the configuration list there are examples of the evidence generated by the
8237 procedures described in the CM plan. In this case, the approach to sampling will depend upon the
8238 level of granularity used in the CM system to control CM items. Where, for example, 10,000 source
8239 code modules are identified in the configuration list, a different sampling strategy needs to be
8240 applied compared to the case in which there are only 5, or even 1. The emphasis of this activity
8241 should be on ensuring that the CM system is being operated correctly, rather than on the detection
8242 of any minor error.
- 8243 For guidance on sampling see A.2, Sampling.
- 8244 ISO/IEC 15408-3 ALC_CMC.3.8C: *The evidence shall demonstrate that the CM system is being*
8245 *operated in accordance with the CM plan.*
- 8246 **13.2.3.3.9 Work unit ALC_CMC.3-9**
- 8247 The evaluator **shall check** the CM documentation to ascertain that it includes the CM system
8248 records identified by the CM plan.

8249 The output produced by the CM system should provide the evidence that the evaluator needs to be
8250 confident that the CM plan is being applied, and also that all configuration items are being
8251 maintained by the CM system as required by **ALC_CMC.3.7C**. Example output could include change
8252 control forms, or configuration item access approval forms.

8253 **13.2.3.3.10 Work unit ALC_CMC.3-10**

8254 The evaluator *shall examine* the evidence to determine that the CM system is being operated in
8255 accordance with the CM plan.

8256 The evaluator should select and examine a sample of evidence covering each type of CM-relevant
8257 operation that has been performed on a configuration item (e.g. creation, modification, deletion,
8258 reversion to an earlier version) to confirm that all operations of the CM system have been carried
8259 out in line with documented procedures. The evaluator confirms that the evidence includes all the
8260 information identified for that operation in the CM plan. Examination of the evidence may require
8261 access to a CM tool that is used. The evaluator may choose to sample the evidence.

8262 For guidance on sampling see A.2, Sampling.

8263 Further confidence in the correct operation of the CM system and the effective maintenance of
8264 configuration items may be established by means of interviews with selected development staff. In
8265 conducting such interviews, the evaluator aims to gain a deeper understanding of how the CM
8266 system is used in practise as well as to confirm that the CM procedures are being applied as
8267 described in the CM documentation. Note that such interviews should complement rather than
8268 replace the examination of documentary evidence, and may not be necessary if the documentary
8269 evidence alone satisfies the requirement. However, given the wide scope of the CM plan it is
8270 possible that some aspects (e.g. roles and responsibilities) may not be clear from the CM plan and
8271 records alone. This is one case where clarification may be necessary through interviews.

8272 It is expected that the evaluator will visit the development site in support of this activity.

8273 For guidance on site visits see A.4, Site Visits.

8274 **13.2.4 Evaluation of sub-activity (ALC_CMC.4)**

8275 **13.2.4.1 Objectives**

8276 The objectives of this sub-activity are to determine whether the developer has clearly identified the
8277 TOE and its associated configuration items, and whether the ability to modify these items is
8278 properly controlled by automated tools, thus making the CM system less susceptible to human
8279 error or negligence.

8280 **13.2.4.2 Input**

8281 The evaluation evidence for this sub-activity is:

- 8282 a) the ST;
- 8283 b) the TOE suitable for testing;
- 8284 c) the configuration management documentation.

8285 **13.2.4.3 Action ALC_CMC.4.1E**

8286 ISO/IEC 15408-3 ALC_CMC.4.1C: *The TOE shall be labelled with its unique reference.*

8287 **13.2.4.3.1 Work unit ALC_CMC.4-1**

8288 The evaluator **shall check** that the TOE provided for evaluation is labelled with its reference.

8289 The evaluator should ensure that the TOE contains the unique reference which is stated in the ST.
8290 This could be achieved through labelled packaging or media, or by a label displayed by the
8291 operational TOE. This is to ensure that it would be possible for consumers to identify the TOE (e.g.
8292 at the point of purchase or use).

8293 The TOE may provide a method by which it can be easily identified. For example, a software TOE
8294 may display its name and version number during the start up routine, or in response to a command
8295 line entry. A hardware or firmware TOE may be identified by a part number physically stamped on
8296 the TOE.

8297 Alternatively, the unique reference provided for the TOE may be the combination of the unique
8298 reference of each component from which the TOE is comprised (e.g. in the case of a composed TOE).

8299 **13.2.4.3.2 Work unit ALC_CMC.4-2**

8300 The evaluator **shall check** that the TOE references used are consistent.

8301 If the TOE is labelled more than once then the labels have to be consistent. For example, it should
8302 be possible to relate any labelled guidance documentation supplied as part of the TOE to the
8303 evaluated operational TOE. This ensures that consumers can be confident that they have purchased
8304 the evaluated version of the TOE, that they have installed this version, and that they have the
8305 correct version of the guidance to operate the TOE in accordance with its ST.

8306 The evaluator also verifies that the TOE reference is consistent with the ST.

8307 If this work unit is applied to a composed TOE, the following will apply. The composed TOE will not
8308 be labelled with its unique (composite) reference, but only the individual components will be
8309 labelled with their appropriate TOE reference. It would require further development for the
8310 composed TOE to be labelled, i.e. during start-up and/or operation, with the composite reference. If
8311 the composed TOE is delivered as the constituent component TOEs, then the TOE items delivered
8312 will not contain the composite reference. However, the composed TOE ST will include the unique
8313 reference for the composed TOE and will identify the components comprising the composed TOE
8314 through which the consumers will be able to determine whether they have the appropriate items.

8315 ISO/IEC 15408-3 ALC_CMC.4.2C: *The CM documentation shall describe the method used to uniquely*
8316 *identify the configuration items.*

8317 **13.2.4.3.3 Work unit ALC_CMC.4-3**

8318 The evaluator **shall examine** the method of identifying configuration items to determine that it
8319 describes how configuration items are uniquely identified.

8320 Procedures should describe how the status of each configuration item can be tracked throughout
8321 the life-cycle of the TOE. The procedures may be detailed in the CM plan or throughout the CM
8322 documentation. The information included should describe:

8323 a) the method how each configuration item is uniquely identified, such that it is possible to
8324 track versions of the same configuration item;

8325 b) the method how configuration items are assigned unique identifiers and how they are
8326 entered into the CM system;

8327 c) the method to be used to identify superseded versions of a configuration item.

8328 ISO/IEC 15408-3 ALC_CMC.4.3C: *The CM system shall uniquely identify all configuration items.*

8329 **13.2.4.3.4 Work unit ALC_CMC.4-4**

8330 The evaluator **shall examine** the configuration items to determine that they are identified in a way
8331 that is consistent with the CM documentation.

8332 Assurance that the CM system uniquely identifies all configuration items is gained by examining
8333 the identifiers for the configuration items. For configuration items identified under ALC_CMS, the
8334 evaluator confirms that each configuration item possesses a unique identifier in a manner
8335 consistent with the unique identification method that is described in the CM documentation.

8336 ISO/IEC 15408-3 ALC_CMC.4.4C: *The CM system shall provide automated measures such that only*
8337 *authorised changes are made to the configuration items.*

8338 **13.2.4.3.5 Work unit ALC_CMC.4-5**

8339 The evaluator **shall examine** the CM access control measures described in the CM plan (cf.
8340 **ALC_CMC.4.6C**) to determine that they are automated and effective in preventing unauthorised
8341 access to the configuration items.

8342 The evaluator may use a number of methods to determine that the CM access control measures are
8343 effective. For example, the evaluator may exercise the access control measures to ensure that the
8344 procedures could not be bypassed. The evaluator may use the outputs generated by the CM system
8345 procedures required by **ALC_CMC.4.10C**. The evaluator may also witness a demonstration of the
8346 CM system to ensure that the access control measures employed are operating effectively.

8347 ISO/IEC 15408-3 ALC_CMC.4.5C: *The CM system shall support the production of the TOE by*
8348 *automated means.*

8349 **13.2.4.3.6 Work unit ALC_CMC.4-6**

8350 The evaluator **shall check** the CM plan (cf. **ALC_CMC.4.6C**) for automated procedures for
8351 supporting the production of the TOE.

8352 The term “production” applies to those processes adopted by the developer to progress the TOE
8353 from the implementation representation to a state acceptable for delivery to the end customer.

8354 The evaluator verifies the existence of automated production support procedures within the CM
8355 plan.

8356 The following are examples for automated means supporting the production of the TOE:

8357 — a “make” tool (as provided with many software development tools) in the case of a software
8358 TOE;

8359 — a tool ensuring automatically (for example by means of bar codes) that only parts are
8360 combined which indeed belong together in the case of a hardware TOE.

8361 **13.2.4.3.7 Work unit ALC_CMC.4-7**

8362 The evaluator **shall examine** the TOE production support procedures to determine that they are
8363 effective in ensuring that a TOE is generated that reflects its implementation representation.

8364 The production support procedures should describe which tools have to be used to produce the
8365 final TOE from the implementation representation in a clearly defined way. The conventions,
8366 directives, or other necessary constructs are described under ALC_TAT.

8367 The evaluator determines that by following the production support procedures the correct
 8368 configuration items would be used to generate the TOE. For example, in a software TOE this may
 8369 include checking that the automated production procedures ensure that all source files and related
 8370 libraries are included in the compiled object code. Moreover, the procedures should ensure that
 8371 compiler options and comparable other options are defined uniquely. For a hardware TOE, this
 8372 work unit may include checking that the automatic production procedures ensure that the
 8373 belonging parts are built together and no parts are missing.

8374 The customer can then be confident that the version of the TOE delivered for installation is derived
 8375 from the implementation representation in an unambiguous way and implements the SFRs as
 8376 described in the ST.

8377 The evaluator should bear in mind that the CM system need not necessarily possess the capability
 8378 to produce the TOE, but should provide support for the process that will help reduce the
 8379 probability of human error.

8380 ISO/IEC 15408-3 ALC_CMC.4.6C: *The CM documentation shall include a CM plan.*

8381 **13.2.4.3.8 Work unit ALC_CMC.4-8**

8382 The evaluator **shall check** that the CM documentation provided includes a CM plan.

8383 The CM plan does not need to be contained within a single document, but it is recommended that
 8384 there is a separate document that describes where the various parts of the CM plan can be found. If
 8385 the CM plan is provided by a set of documents, the list in the following work unit gives guidance
 8386 regarding the required content.

8387 ISO/IEC 15408-3 ALC_CMC.4.7C: *The CM plan shall describe how the CM system is used for the*
 8388 *development of the TOE.*

8389 **13.2.4.3.9 Work unit ALC_CMC.4-9**

8390 The evaluator **shall examine** the CM plan to determine that it describes how the CM system is used
 8391 for the development of the TOE.

8392 The descriptions contained in a CM plan include, if applicable:

8393 a) all activities performed in the TOE development that are subject to configuration
 8394 management procedures (e.g. creation, modification or deletion of a configuration item,
 8395 data-backup, archiving);

8396 b) which means (e.g. CM tools, forms) have to be made available;

8397 c) the usage of the CM tools: the necessary details for a user of the CM system to be able to
 8398 operate the CM tools correctly in order to maintain the integrity of the TOE;

8399 d) the production support procedures;

8400 e) which other objects (development components, tools, assessment environments, etc) are
 8401 taken under CM control;

8402 f) the roles and responsibilities of individuals required to perform operations on individual
 8403 configuration items (different roles may be identified for different types of configuration
 8404 items (e.g. design documentation or source code));

8405 g) how CM instances (e.g. change control boards, interface control working groups) are
 8406 introduced and staffed;

- 8407 h) the description of the change management;
- 8408 i) the procedures that are used to ensure that only authorised individuals can make changes
8409 to configuration items;
- 8410 j) the procedures that are used to ensure that concurrency problems do not occur as a
8411 result of simultaneous changes to configuration items;
- 8412 k) the evidence that is generated as a result of application of the procedures. For example,
8413 for a change to a configuration item, the CM system might record a description of the
8414 change, accountability for the change, identification of all configuration items affected,
8415 status (e.g. pending or completed), and date and time of the change. This might be
8416 recorded in an audit trail of changes made or change control records;
- 8417 l) the approach to version control and unique referencing of TOE versions (e.g. covering the
8418 release of patches in operating systems, and the subsequent detection of their
8419 application).
- 8420 ISO/IEC 15408-3 ALC_CMC.4.8C: *The CM plan shall describe the procedures used to accept modified*
8421 *or newly created configuration items as part of the TOE.*
- 8422 **13.2.4.3.10 Work unit ALC_CMC.4-10**
- 8423 The evaluator ***shall examine*** the CM plan to determine that it describes the procedures used to
8424 accept modified or newly created configuration items as parts of the TOE.
- 8425 The descriptions of the acceptance procedures in the CM plan should include the developer roles or
8426 individuals responsible for the acceptance and the criteria to be used for acceptance. They should
8427 take into account all acceptance situations that may occur, in particular:
- 8428 a) accepting an item into the CM system for the first time, in particular inclusion of software,
8429 firmware and hardware components from other manufacturers into the TOE
8430 (“integration”);
- 8431 b) moving configuration items to the next life-cycle phase at each stage of the construction of
8432 the TOE (e.g. module, subsystem, system);
- 8433 c) subsequent to transports between different development sites.
- 8434 If this work unit is applied to a dependent component that is going to be integrated in a composed
8435 TOE, the CM plan should consider the control of base components obtained by the dependent TOE
8436 developer.
- 8437 When obtaining the components the evaluators are to verify the following:
- 8438 a) Transfer of each base component from the base component developer to the integrator
8439 (dependent TOE developer) was performed in accordance with the base component
8440 TOE's secure delivery procedures, as reported in the base component TOE certification
8441 report.
- 8442 b) The component received has the same identifiers as those stated in the ST and
8443 Certification Report for the component TOE.
- 8444 c) All additional material required by a developer for composition (integration) is provided.
8445 This is to include the necessary extract of the component TOE's functional specification.
- 8446 ISO/IEC 15408-3 ALC_CMC.4.9C: *The evidence shall demonstrate that all configuration items are*
8447 *being maintained under the CM system.*

8448 **13.2.4.3.11 Work unit ALC_CMC.4-11**

8449 The evaluator **shall check** that the configuration items identified in the configuration list are being
8450 maintained by the CM system.

8451 The CM system employed by the developer should maintain the integrity of the TOE. The evaluator
8452 should check that for each type of configuration item (e.g. design documents or source code
8453 modules) contained in the configuration list there are examples of the evidence generated by the
8454 procedures described in the CM plan. In this case, the approach to sampling will depend upon the
8455 level of granularity used in the CM system to control CM items. Where, for example, 10,000 source
8456 code modules are identified in the configuration list, a different sampling strategy needs to be
8457 applied compared to the case in which there are only 5, or even 1. The emphasis of this activity
8458 should be on ensuring that the CM system is being operated correctly, rather than on the detection
8459 of any minor error.

8460 For guidance on sampling see A.2, Sampling.

8461 ISO/IEC 15408-3 ALC_CMC.4.10C: *The evidence shall demonstrate that the CM system is being*
8462 *operated in accordance with the CM plan.*

8463 **13.2.4.3.12 Work unit ALC_CMC.4-12**

8464 The evaluator **shall check** the CM documentation to ascertain that it includes the CM system
8465 records identified by the CM plan.

8466 The output produced by the CM system should provide the evidence that the evaluator needs to be
8467 confident that the CM plan is being applied, and also that all configuration items are being
8468 maintained by the CM system as required by **ALC_CMC.4.9C**. Example output could include change
8469 control forms, or configuration item access approval forms.

8470 **13.2.4.3.13 Work unit ALC_CMC.4-13**

8471 The evaluator **shall examine** the evidence to determine that the CM system is being operated in
8472 accordance with the CM plan.

8473 The evaluator should select and examine a sample of evidence covering each type of CM-relevant
8474 operation that has been performed on a configuration item (e.g. creation, modification, deletion,
8475 reversion to an earlier version) to confirm that all operations of the CM system have been carried
8476 out in line with documented procedures. The evaluator confirms that the evidence includes all the
8477 information identified for that operation in the CM plan. Examination of the evidence may require
8478 access to a CM tool that is used. The evaluator may choose to sample the evidence.

8479 For guidance on sampling see A.2, Sampling.

8480 Further confidence in the correct operation of the CM system and the effective maintenance of
8481 configuration items may be established by means of interviews with selected development staff. In
8482 conducting such interviews, the evaluator aims to gain a deeper understanding of how the CM
8483 system is used in practise as well as to confirm that the CM procedures are being applied as
8484 described in the CM documentation. Note that such interviews should complement rather than
8485 replace the examination of documentary evidence, and may not be necessary if the documentary
8486 evidence alone satisfies the requirement. However, given the wide scope of the CM plan it is
8487 possible that some aspects (e.g. roles and responsibilities) may not be clear from the CM plan and
8488 records alone. This is one case where clarification may be necessary through interviews.

8489 It is expected that the evaluator will visit the development site in support of this activity.

8490 For guidance on site visits see A.4, Site Visits.

8491 **13.2.5 Evaluation of sub-activity (ALC_CMC.5)**

8492 **13.2.5.1 Objectives**

8493 The objectives of this sub-activity are to determine whether the developer has clearly identified the
8494 TOE and its associated configuration items, and whether the ability to modify these items is
8495 properly controlled by automated tools, thus making the CM system less susceptible to human
8496 error or negligence.

8497 **13.2.5.2 Input**

8498 The evaluation evidence for this sub-activity is:

- 8499 a) the ST;
- 8500 b) the TOE suitable for testing;
- 8501 c) the configuration management documentation.

8502 **13.2.5.3 Action ALC_CMC.5.1E**

8503 ISO/IEC 15408-3 ALC_CMC.5.1C: *The TOE shall be labelled with its unique reference.*

8504 **13.2.5.3.1 Work unit ALC_CMC.5-1**

8505 The evaluator **shall check** that the TOE provided for evaluation is labelled with its reference.

8506 The evaluator should ensure that the TOE contains the unique reference which is stated in the ST.
8507 This could be achieved through labelled packaging or media, or by a label displayed by the
8508 operational TOE. This is to ensure that it would be possible for consumers to identify the TOE (e.g.
8509 at the point of purchase or use).

8510 The TOE may provide a method by which it can be easily identified. For example, a software TOE
8511 may display its name and version number during the start up routine, or in response to a command
8512 line entry. A hardware or firmware TOE may be identified by a part number physically stamped on
8513 the TOE.

8514 Alternatively, the unique reference provided for the TOE may be the combination of the unique
8515 reference of each component from which the TOE is comprised (e.g. in the case of a composed TOE).

8516 **13.2.5.3.2 Work unit ALC_CMC.5-2**

8517 The evaluator **shall check** that the TOE references used are consistent.

8518 If the TOE is labelled more than once then the labels have to be consistent. For example, it should
8519 be possible to relate any labelled guidance documentation supplied as part of the TOE to the
8520 evaluated operational TOE. This ensures that consumers can be confident that they have purchased
8521 the evaluated version of the TOE, that they have installed this version, and that they have the
8522 correct version of the guidance to operate the TOE in accordance with its ST.

8523 The evaluator also verifies that the TOE reference is consistent with the ST.

8524 If this work unit is applied to a composed TOE, the following will apply. The composed IT TOE will
8525 not be labelled with its unique (composite) reference, but only the individual components will be
8526 labelled with their appropriate TOE reference. It would require further development for the IT TOE
8527 to be labelled, i.e. during start-up and/or operation, with the composite reference. If the composed
8528 TOE is delivered as the constituent component TOEs, then the TOE items delivered will not contain
8529 the composite reference. However, the composed TOE ST will include the unique reference for the

8530 composed TOE and will identify the components comprising the composed TOE through which the
8531 consumers will be able to determine whether they have the appropriate items.

8532 ISO/IEC 15408-3 ALC_CMC.5.2C: *The CM documentation shall describe the method used to uniquely*
8533 *identify the configuration items.*

8534 **13.2.5.3.3 Work unit ALC_CMC.5-3**

8535 The evaluator **shall examine** the method of identifying configuration items to determine that it
8536 describes how configuration items are uniquely identified.

8537 Procedures should describe how the status of each configuration item can be tracked throughout
8538 the life-cycle of the TOE. The procedures may be detailed in the CM plan or throughout the CM
8539 documentation. The information included should describe:

8540 a) the method how each configuration item is uniquely identified, such that it is possible to
8541 track versions of the same configuration item;

8542 b) the method how configuration items are assigned unique identifiers and how they are
8543 entered into the CM system;

8544 c) the method to be used to identify superseded versions of a configuration item.

8545 ISO/IEC 15408-3 ALC_CMC.5.3C: *The CM documentation shall justify that the acceptance procedures*
8546 *provide for an adequate and appropriate review of changes to all configuration items.*

8547 **13.2.5.3.4 Work unit ALC_CMC.5-4**

8548 The evaluator **shall examine** the CM documentation to determine that it justifies that the
8549 acceptance procedures provide for an adequate and appropriate review of changes to all
8550 configuration items.

8551 The CM documentation should make it sufficiently clear that by following the acceptance
8552 procedures only parts of adequate quality are incorporated into the TOE.

8553 ISO/IEC 15408-3 ALC_CMC.5.4C: *The CM system shall uniquely identify all configuration items.*

8554 **13.2.5.3.5 Work unit ALC_CMC.5-5**

8555 The evaluator **shall examine** the configuration items to determine that they are identified in a way
8556 that is consistent with the CM documentation.

8557 Assurance that the CM system uniquely identifies all configuration items is gained by examining
8558 the identifiers for the configuration items. For both configuration items that comprise the TOE, and
8559 drafts of configuration items that are submitted by the developer as evaluation evidence, the
8560 evaluator confirms that each configuration item possesses a unique identifier in a manner
8561 consistent with the unique identification method that is described in the CM documentation.

8562 ISO/IEC 15408-3 ALC_CMC.5.5C: *The CM system shall provide automated measures such that only*
8563 *authorised changes are made to the configuration items.*

8564 **13.2.5.3.6 Work unit ALC_CMC.5-6**

8565 The evaluator **shall examine** the CM access control measures described in the CM plan (cf.
8566 **ALC_CMC.5.12C**) to determine that they are automated and effective in preventing unauthorised
8567 access to the configuration items.

8568 The evaluator may use a number of methods to determine that the CM access control measures are
8569 effective. For example, the evaluator may exercise the access control measures to ensure that the
8570 procedures could not be bypassed. The evaluator may use the outputs generated by the CM system
8571 procedures required by [ALC_CMC.5.16C](#). The evaluator may also witness a demonstration of the
8572 CM system to ensure that the access control measures employed are operating effectively.

8573 ISO/IEC 15408-3 ALC_CMC.5.6C: *The CM system shall support the production of the TOE by*
8574 *automated means.*

8575 **13.2.5.3.7 Work unit ALC_CMC.5-7**

8576 The evaluator **shall check** the CM plan (cf. [ALC_CMC.5.12C](#)) for automated procedures for
8577 supporting the production of the TOE.

8578 The term “production” applies to those processes adopted by the developer to progress the TOE
8579 from the implementation representation to a state acceptable for delivery to the end customer.

8580 The evaluator verifies the existence of automated production support procedures within the CM
8581 plan.

8582 The following are examples for automated means supporting the production of the TOE:

8583 — a “make” tool (as provided with many software development tools) in the case of a software
8584 TOE;

8585 — a tool ensuring automatically (for example by means of bar codes) that only parts are
8586 combined which indeed belong together in the case of a hardware TOE.

8587 **13.2.5.3.8 Work unit ALC_CMC.5-8**

8588 The evaluator **shall examine** the TOE production support procedures to determine that they are
8589 effective in ensuring that a TOE is generated that reflects its implementation representation.

8590 The production support procedures should describe which tools have to be used to produce the
8591 final TOE from the implementation representation in a clearly defined way. The conventions,
8592 directives, or other necessary constructs are described under ALC_TAT.

8593 The evaluator determines that by following the production support procedures the correct
8594 configuration items would be used to generate the TOE. For example, in a software TOE this may
8595 include checking that the automated production procedures ensure that all source files and related
8596 libraries are included in the compiled object code. Moreover, the procedures should ensure that
8597 compiler options and comparable other options are defined uniquely. For a hardware TOE, this
8598 work unit may include checking that the automatic production procedures ensure that the
8599 belonging parts are built together and no parts are missing.

8600 The customer can then be confident that the version of the TOE delivered for installation is derived
8601 from the implementation representation in an unambiguous way and implements the SFRs as
8602 described in the ST.

8603 The evaluator should bear in mind that the CM system need not necessarily possess the capability
8604 to produce the TOE, but should provide support for the process that will help reduce the
8605 probability of human error.

8606 ISO/IEC 15408-3 ALC_CMC.5.7C: *The CM system shall ensure that the person responsible for*
8607 *accepting a configuration item into CM is not the person who developed it.*

8608 **13.2.5.3.9 Work unit ALC_CMC.5-9**

8609 The evaluator *shall examine* the CM system to determine that it ensures that the person
8610 responsible for accepting a configuration item is not the person who developed it.

8611 The acceptance procedures describe who is responsible for accepting a configuration item. From
8612 these descriptions, the evaluator should be able to determine that the person who developed a
8613 configuration item is in no case responsible for its acceptance.

8614 ISO/IEC 15408-3 ALC_CMC.5.8C: *The CM system shall identify the configuration items that comprise*
8615 *the TSF.*

8616 **13.2.5.3.10 Work unit ALC_CMC.5-10**

8617 The evaluator *shall examine* the CM system to determine that it identifies the configuration items
8618 that comprise the TSF.

8619 The CM documentation should describe how the CM system identifies the configuration items that
8620 comprise the TSF. The evaluator should select a sample of configuration items covering each type
8621 of items, particularly containing TSF and non-TSF items, and check that they are correctly classified
8622 by the CM system.

8623 For guidance on sampling see A.2, Sampling.

8624 ISO/IEC 15408-3 ALC_CMC.5.9C: *The CM system shall support the audit of all changes to the TOE by*
8625 *automated means, including the originator, date, and time in the audit trail.*

8626 **13.2.5.3.11 Work unit ALC_CMC.5-11**

8627 The evaluator *shall examine* the CM system to determine that it supports the audit of all changes
8628 to the TOE by automated means, including the originator, date, and time in the audit trail.

8629 The evaluator should inspect a sample of audit trails and check, if they contain the minimum
8630 information.

8631 ISO/IEC 15408-3 ALC_CMC.5.10C: *The CM system shall provide an automated means to identify all*
8632 *other configuration items that are affected by the change of a given configuration item.*

8633 **13.2.5.3.12 Work unit ALC_CMC.5-12**

8634 The evaluator *shall examine* the CM system to determine that it provides an automated means to
8635 identify all other configuration items that are affected by the change of a given configuration item.

8636 The CM documentation should describe how the CM system identifies all other configuration items
8637 that are affected by the change of a given configuration item. The evaluator should select a sample
8638 of configuration items, covering all types of items, and exercise the automated means to determine
8639 that it identifies all items that are affected by the change of the selected item.

8640 For guidance on sampling see A.2, Sampling.

8641 ISO/IEC 15408-3 ALC_CMC.5.11C: *The CM system shall be able to identify the version of the*
8642 *implementation representation from which the TOE is generated.*

8643 **13.2.5.3.13 Work unit ALC_CMC.5-13**

8644 The evaluator *shall examine* the CM system to determine that it is able to identify the version of
8645 the implementation representation from which the TOE is generated.

8646 The CM documentation should describe how the CM system identifies the version of the
8647 implementation representation from which the TOE is generated. The evaluator should select a
8648 sample of the parts used to produce the TOE and should apply the CM system to verify that it
8649 identifies the corresponding implementation representation in the correct version.

8650 For guidance on sampling see A.2, Sampling.

8651 ISO/IEC 15408-3 ALC_CMC.5.12C: *The CM documentation shall include a CM plan.*

8652 **13.2.5.3.14 Work unit ALC_CMC.5-14**

8653 The evaluator **shall check** that the CM documentation provided includes a CM plan.

8654 The CM plan needs not to be a connected document, but it is recommended that there is a single
8655 document that describes where the various parts of the CM plan can be found. If the CM plan is no
8656 single document, the list in the following work unit gives hints regarding which context is expected.

8657 ISO/IEC 15408-3 ALC_CMC.5.13C: *The CM plan shall describe how the CM system is used for the*
8658 *development of the TOE.*

8659 **13.2.5.3.15 Work unit ALC_CMC.5-15**

8660 The evaluator **shall examine** the CM plan to determine that it describes how the CM system is used
8661 for the development of the TOE.

8662 The descriptions contained in a CM plan include, if applicable:

8663 a) all activities performed in the TOE development that are subject to configuration
8664 management procedures (e.g. creation, modification or deletion of a configuration item,
8665 data-backup, archiving);

8666 b) which means (e.g. CM tools, forms) have to be made available;

8667 c) the usage of the CM tools: the necessary details for a user of the CM system to be able to
8668 operate the CM tools correctly in order to maintain the integrity of the TOE;

8669 d) the production support procedures;

8670 e) which other objects (development components, tools, assessment environments, etc) are
8671 taken under CM control;

8672 f) the roles and responsibilities of individuals required to perform operations on individual
8673 configuration items (different roles may be identified for different types of configuration
8674 items (e.g. design documentation or source code));

8675 g) how CM instances (e.g. change control boards, interface control working groups) are
8676 introduced and staffed;

8677 h) the description of the change management;

8678 i) the procedures that are used to ensure that only authorised individuals can make changes
8679 to configuration items;

8680 j) the procedures that are used to ensure that concurrency problems do not occur as a
8681 result of simultaneous changes to configuration items;

8682 k) the evidence that is generated as a result of application of the procedures. For example,
8683 for a change to a configuration item, the CM system might record a description of the

- 8684 change, accountability for the change, identification of all configuration items affected,
8685 status (e.g. pending or completed), and date and time of the change. This might be
8686 recorded in an audit trail of changes made or change control records;
- 8687 l) the approach to version control and unique referencing of TOE versions (e.g. covering the
8688 release of patches in operating systems, and the subsequent detection of their
8689 application).
- 8690 ISO/IEC 15408-3 ALC_CMC.5.14C: *The CM plan shall describe the procedures used to accept modified*
8691 *or newly created configuration items as part of the TOE.*
- 8692 **13.2.5.3.16 Work unit ALC_CMC.5-16**
- 8693 The evaluator **shall examine** the CM plan to determine that it describes the procedures used to
8694 accept modified or newly created configuration items as parts of the TOE.
- 8695 The descriptions of the acceptance procedures in the CM plan should include the developer roles or
8696 individuals responsible for the acceptance and the criteria to be used for acceptance. They should
8697 take into account all acceptance situations that may occur, in particular:
- 8698 a) accepting an item into the CM system for the first time, in particular inclusion of software,
8699 firmware and hardware components from other manufacturers into the TOE
8700 (“integration”);
- 8701 b) moving configuration items to the next life-cycle phase at each stage of the construction of
8702 the TOE (e.g. module, subsystem, system);
- 8703 c) subsequent to transports between different development sites.
- 8704 ISO/IEC 15408-3 ALC_CMC.5.15C: *The evidence shall demonstrate that all configuration items are*
8705 *being maintained under the CM system.*
- 8706 **13.2.5.3.17 Work unit ALC_CMC.5-17**
- 8707 The evaluator **shall check** that the configuration items identified in the configuration list are being
8708 maintained by the CM system.
- 8709 The CM system employed by the developer should maintain the integrity of the TOE. The evaluator
8710 should check that for each type of configuration item (e.g. design documents or source code
8711 modules) contained in the configuration list there are examples of the evidence generated by the
8712 procedures described in the CM plan. In this case, the approach to sampling will depend upon the
8713 level of granularity used in the CM system to control CM items. Where, for example, 10,000 source
8714 code modules are identified in the configuration list, a different sampling strategy needs to be
8715 applied compared to the case in which there are only 5, or even 1. The emphasis of this activity
8716 should be on ensuring that the CM system is being operated correctly, rather than on the detection
8717 of any minor error.
- 8718 For guidance on sampling see A.2, Sampling.
- 8719 ISO/IEC 15408-3 ALC_CMC.5.16C: *The evidence shall demonstrate that the CM system is being*
8720 *operated in accordance with the CM plan.*
- 8721 **13.2.5.3.18 Work unit ALC_CMC.5-18**
- 8722 The evaluator **shall check** the CM documentation to ascertain that it includes the CM system
8723 records identified by the CM plan.

8724 The output produced by the CM system should provide the evidence that the evaluator needs to be
8725 confident that the CM plan is being applied, and also that all configuration items are being
8726 maintained by the CM system as required by **ALC_CMC.5.15C**. Example output could include change
8727 control forms, or configuration item access approval forms.

8728 **13.2.5.3.19 Work unit ALC_CMC.5-19**

8729 The evaluator *shall examine* the evidence to determine that the CM system is being operated in
8730 accordance with the CM plan.

8731 The evaluator should select and examine a sample of evidence covering each type of CM-relevant
8732 operation that has been performed on a configuration item (e.g. creation, modification, deletion,
8733 reversion to an earlier version) to confirm that all operations of the CM system have been carried
8734 out in line with documented procedures. The evaluator confirms that the evidence includes all the
8735 information identified for that operation in the CM plan. Examination of the evidence may require
8736 access to a CM tool that is used. The evaluator may choose to sample the evidence.

8737 For guidance on sampling see A.2, Sampling.

8738 Further confidence in the correct operation of the CM system and the effective maintenance of
8739 configuration items may be established by means of interviews with selected development staff. In
8740 conducting such interviews, the evaluator aims to gain a deeper understanding of how the CM
8741 system is used in practise as well as to confirm that the CM procedures are being applied as
8742 described in the CM documentation. Note that such interviews should complement rather than
8743 replace the examination of documentary evidence, and may not be necessary if the documentary
8744 evidence alone satisfies the requirement. However, given the wide scope of the CM plan it is
8745 possible that some aspects (e.g. roles and responsibilities) may not be clear from the CM plan and
8746 records alone. This is one case where clarification may be necessary through interviews.

8747 It is expected that the evaluator will visit the development site in support of this activity.

8748 For guidance on site visits see A.4, Site Visits.

8749 **13.2.5.4 Action ALC_CMC.5.2E**

8750 **13.2.5.4.1 Work unit ALC_CMC.5-20**

8751 The evaluator *shall examine* the production support procedures to determine that by following
8752 these procedures a TOE would be produced like that one provided by the developer for testing
8753 activities.

8754 If the TOE is a small software TOE and production consists of compiling and linking, the evaluator
8755 might confirm the adequacy of the production support procedures by reapplying them himself.

8756 If the production process of the TOE is more complicated (as for example in the case of a smart
8757 card), but has already started, the evaluator should inspect the application of the production
8758 support procedures during a visit of the development site. They might compare a copy of the TOE
8759 produced in their presence with the samples used for their testing activities.

8760 For guidance on site visits see A.4, Site Visits.

8761 Otherwise the evaluator's determination should be based on the documentary evidence provided
8762 by the developer.

8763 This work unit may be performed in conjunction with the evaluation activities under
8764 Implementation representation (ADV_IMP).

8765 **13.3 CM scope (ALC_CMS)**8766 **13.3.1 Evaluation of sub-activity (ALC_CMS.1)**8767 **13.3.1.1 Objectives**

8768 The objective of this sub-activity is to determine whether the developer performs configuration
8769 management on the TOE and the evaluation evidence. These configuration items are controlled in
8770 accordance with CM capabilities (ALC_CMC).

8771 **13.3.1.2 Input**

8772 The evaluation evidence for this sub-activity is:

8773 a) the ST;

8774 b) the configuration list.

8775 **13.3.1.3 Action ALC_CMS.1.1E**

8776 ISO/IEC 15408-3 ALC_CMS.1.1C: *The configuration list shall include the following: the TOE itself; and*
8777 *the evaluation evidence required by the SARs.*

8778 **13.3.1.3.1 Work unit ALC_CMS.1-1**

8779 The evaluator **shall check** that the configuration list includes the following set of items:

8780 a) the TOE itself;

8781 b) the evaluation evidence required by the SARs in the ST.

8782 ISO/IEC 15408-3 ALC_CMS.1.2C: *The configuration list shall uniquely identify the configuration items.*

8783 **13.3.1.3.2 Work unit ALC_CMS.1-2**

8784 The evaluator **shall examine** the configuration list to determine that it uniquely identifies each
8785 configuration item.

8786 The configuration list contains sufficient information to uniquely identify which version of each
8787 item has been used (typically a version number). Use of this list will enable the evaluator to check
8788 that the correct configuration items, and the correct version of each item, have been used during
8789 the evaluation.

8790 **13.3.2 Evaluation of sub-activity (ALC_CMS.2)**8791 **13.3.2.1 Objectives**

8792 The objective of this sub-activity is to determine whether the configuration list includes the TOE,
8793 the parts that comprise the TOE, and the evaluation evidence. These configuration items are
8794 controlled in accordance with CM capabilities (ALC_CMC).

8795 **13.3.2.2 Input**

8796 The evaluation evidence for this sub-activity is:

8797 a) the ST;

8798 b) the configuration list.

8799 **13.3.2.3 Action ALC_CMS.2.1E**

8800 ISO/IEC 15408-3 ALC_CMS.2.1C: *The configuration list shall include the following: the TOE itself; the*
8801 *evaluation evidence required by the SARs; and the parts that comprise the TOE.*

8802 **13.3.2.3.1 Work unit ALC_CMS.2-1**

8803 The evaluator **shall check** that the configuration list includes the following set of items:

8804 a) the TOE itself;

8805 b) the parts that comprise the TOE;

8806 c) the evaluation evidence required by the SARs.

8807 ISO/IEC 15408-3 ALC_CMS.2.2C: *The configuration list shall uniquely identify the configuration items.*

8808 **13.3.2.3.2 Work unit ALC_CMS.2-2**

8809 The evaluator **shall examine** the configuration list to determine that it uniquely identifies each
8810 configuration item.

8811 The configuration list contains sufficient information to uniquely identify which version of each
8812 item has been used (typically a version number). Use of this list will enable the evaluator to check
8813 that the correct configuration items, and the correct version of each item, have been used during
8814 the evaluation.

8815 ISO/IEC 15408-3 ALC_CMS.2.3C: *For each TSF relevant configuration item, the configuration list*
8816 *shall indicate the developer of the item.*

8817 **13.3.2.3.3 Work unit ALC_CMS.2-3**

8818 The evaluator **shall check** that the configuration list indicates the developer of each TSF relevant
8819 configuration item.

8820 If only one developer is involved in the development of the TOE, this work unit is not applicable,
8821 and is therefore considered to be satisfied.

8822 **13.3.3 Evaluation of sub-activity (ALC_CMS.3)**

8823 **13.3.3.1 Objectives**

8824 The objective of this sub-activity is to determine whether the configuration list includes the TOE,
8825 the parts that comprise the TOE, the TOE implementation representation, and the evaluation
8826 evidence. These configuration items are controlled in accordance with CM capabilities (ALC_CMC).

8827 **13.3.3.2 Input**

8828 The evaluation evidence for this sub-activity is:

8829 a) the ST;

8830 b) the configuration list.

8831 **13.3.3.3 Action ALC_CMS.3.1E**

8832 ISO/IEC 15408-3 ALC_CMS.3.1C: *The configuration list shall include the following: the TOE itself; the*
 8833 *evaluation evidence required by the SARs; the parts that comprise the TOE; and the implementation*
 8834 *representation.*

8835 **13.3.3.3.1 Work unit ALC_CMS.3-1**

8836 The evaluator **shall check** that the configuration list includes the following set of items:

- 8837 a) the TOE itself;
- 8838 b) the parts that comprise the TOE;
- 8839 c) the TOE implementation representation;
- 8840 d) the evaluation evidence required by the SARs in the ST.

8841 ISO/IEC 15408-3 ALC_CMS.3.2C: *The configuration list shall uniquely identify the configuration items.*

8842 **13.3.3.3.2 Work unit ALC_CMS.3-2**

8843 The evaluator **shall examine** the configuration list to determine that it uniquely identifies each
 8844 configuration item.

8845 The configuration list contains sufficient information to uniquely identify which version of each
 8846 item has been used (typically a version number). Use of this list will enable the evaluator to check
 8847 that the correct configuration items, and the correct version of each item, have been used during
 8848 the evaluation.

8849 ISO/IEC 15408-3 ALC_CMS.3.3C: *For each TSF relevant configuration item, the configuration list*
 8850 *shall indicate the developer of the item.*

8851 **13.3.3.3.3 Work unit ALC_CMS.3-3**

8852 The evaluator **shall check** that the configuration list indicates the developer of each TSF relevant
 8853 configuration item.

8854 If only one developer is involved in the development of the TOE, this work unit is not applicable,
 8855 and is therefore considered to be satisfied.

8856 **13.3.4 Evaluation of sub-activity (ALC_CMS.4)**8857 **13.3.4.1 Objectives**

8858 The objective of this sub-activity is to determine whether the configuration list includes the TOE,
 8859 the parts that comprise the TOE, the TOE implementation representation, security flaws, and the
 8860 evaluation evidence. These configuration items are controlled in accordance with CM capabilities
 8861 (ALC_CMC).

8862 **13.3.4.2 Input**

8863 The evaluation evidence for this sub-activity is:

- 8864 a) the ST;
- 8865 b) the configuration list.

8866 **13.3.4.3 Action ALC_CMS.4.1E**

8867 ISO/IEC 15408-3 ALC_CMS.4.1C: *The configuration list shall include the following: the TOE itself; the*
8868 *evaluation evidence required by the SARs; the parts that comprise the TOE; the implementation*
8869 *representation; and security flaw reports and resolution status.*

8870 **13.3.4.3.1 Work unit ALC_CMS.4-1**

8871 The evaluator **shall check** that the configuration list includes the following set of items:

8872 a) the TOE itself;

8873 b) the parts that comprise the TOE;

8874 c) the TOE implementation representation;

8875 d) the evaluation evidence required by the SARs in the ST;

8876 e) the documentation used to record details of reported security flaws associated with the
8877 implementation (e.g., problem status reports derived from a developer's problem
8878 database).

8879 ISO/IEC 15408-3 ALC_CMS.4.2C: *The configuration list shall uniquely identify the configuration items.*

8880 **13.3.4.3.2 Work unit ALC_CMS.4-2**

8881 The evaluator **shall examine** the configuration list to determine that it uniquely identifies each
8882 configuration item.

8883 The configuration list contains sufficient information to uniquely identify which version of each
8884 item has been used (typically a version number). Use of this list will enable the evaluator to check
8885 that the correct configuration items, and the correct version of each item, have been used during
8886 the evaluation.

8887 ISO/IEC 15408-3 ALC_CMS.4.3C: *For each TSF relevant configuration item, the configuration list*
8888 *shall indicate the developer of the item.*

8889 **13.3.4.3.3 Work unit ALC_CMS.4-3**

8890 The evaluator **shall check** that the configuration list indicates the developer of each TSF relevant
8891 configuration item.

8892 If only one developer is involved in the development of the TOE, this work unit is not applicable,
8893 and is therefore considered to be satisfied.

8894 **13.3.5 Evaluation of sub-activity (ALC_CMS.5)**

8895 **13.3.5.1 Objectives**

8896 The objective of this sub-activity is to determine whether the configuration list includes the TOE,
8897 the parts that comprise the TOE, the TOE implementation representation, security flaws,
8898 development tools and related information, and the evaluation evidence. These configuration items
8899 are controlled in accordance with CM capabilities (ALC_CMC).

8900 **13.3.5.2 Input**

8901 The evaluation evidence for this sub-activity is:

8902 a) the ST;

8903 b) the configuration list.

8904 **13.3.5.3 Action ALC_CMS.5.1E**

8905 ISO/IEC 15408-3 ALC_CMS.5.1C: *The configuration list shall include the following: the TOE itself; the*
 8906 *evaluation evidence required by the SARs; the parts that comprise the TOE; the implementation*
 8907 *representation; security flaw reports and resolution status; and development tools and related*
 8908 *information.*

8909 **13.3.5.3.1 Work unit ALC_CMS.5-1**

8910 The evaluator **shall check** that the configuration list includes the following set of items:

8911 a) the TOE itself;

8912 b) the parts that comprise the TOE;

8913 c) the TOE implementation representation;

8914 d) the evaluation evidence required by the SARs in the ST;

8915 e) the documentation used to record details of reported security flaws associated with the
 8916 implementation (e.g., problem status reports derived from a developer's problem
 8917 database);

8918 f) all tools (incl. test software, if applicable) involved in the development and production of
 8919 the TOE including the names, versions, configurations and roles of each development tool,
 8920 and related documentation.

8921 For a software TOE, “development tools” are usually programming languages and compiler and
 8922 “related documentation” comprises compiler and linker options. For a hardware TOE,
 8923 “development tools” might be hardware design languages, simulation and synthesis tools,
 8924 compilers, and “related documentation” might comprise compiler options again.

8925 ISO/IEC 15408-3 ALC_CMS.5.2C: *The configuration list shall uniquely identify the configuration items.*

8926 **13.3.5.3.2 Work unit ALC_CMS.5-2**

8927 The evaluator **shall examine** the configuration list to determine that it uniquely identifies each
 8928 configuration item.

8929 The configuration list contains sufficient information to uniquely identify which version of each
 8930 item has been used (typically a version number). Use of this list will enable the evaluator to check
 8931 that the correct configuration items, and the correct version of each item, have been used during
 8932 the evaluation.

8933 ISO/IEC 15408-3 ALC_CMS.5.3C: *For each TSF relevant configuration item, the configuration list*
 8934 *shall indicate the developer of the item.*

8935 **13.3.5.3.3 Work unit ALC_CMS.5-3**

8936 The evaluator **shall check** that the configuration list indicates the developer of each TSF relevant
 8937 configuration item.

8938 If only one developer is involved in the development of the TOE, this work unit is not applicable,
 8939 and is therefore considered to be satisfied.

8940 **13.4 Delivery (ALC_DEL)**

8941 **13.4.1 Evaluation of sub-activity (ALC_DEL.1)**

8942 **13.4.1.1 Objectives**

8943 The objective of this sub-activity is to determine whether the delivery documentation describes all
8944 procedures used to maintain security of the TOE when distributing the TOE to the user.

8945 **13.4.1.2 Input**

8946 The evaluation evidence for this sub-activity is:

8947 a) the ST;

8948 b) the delivery documentation.

8949 **13.4.1.3 Action ALC_DEL.1.1E**

8950 ISO/IEC 15408-3 ALC_DEL.1.1C: *The delivery documentation shall describe all procedures that are*
8951 *necessary to maintain security when distributing versions of the TOE to the consumer.*

8952 **13.4.1.3.1 Work unit ALC_DEL.1-1**

8953 The evaluator *shall examine* the delivery documentation to determine that it describes all
8954 procedures that are necessary to maintain security when distributing versions of the TOE or parts
8955 of it to the consumer.

8956 The delivery documentation describes proper procedures to maintain security of the TOE during
8957 transfer of the TOE or its component parts and to determine the identification of the TOE.

8958 The delivery documentation should cover the entire TOE, but may contain different procedures for
8959 different parts of the TOE. The evaluation should consider the totality of procedures.

8960 The delivery procedures should be applicable across all phases of delivery from the production
8961 environment to the installation environment (e.g. packaging, storage and distribution). Standard
8962 commercial practise for packaging and delivery may be acceptable. This includes shrink wrapped
8963 packaging, a security tape or a sealed envelope. For the distribution, physical (e.g. public mail or a
8964 private distribution service) or electronic (e.g. electronic mail or downloading off the Internet)
8965 procedures may be used.

8966 Cryptographic checksums or a software signature may be used by the developer to ensure that
8967 tampering or masquerading can be detected. Tamper proof seals additionally indicate if the
8968 confidentiality has been broken. For software TOEs, confidentiality might be assured by using
8969 encryption. If availability is of concern, a secure transportation might be required.

8970 Interpretation of the term “necessary to maintain security” will need to consider:

8971 — The nature of the TOE (e.g. whether it is software or hardware).

8972 — The overall security level stated for the TOE by the chosen level of the Vulnerability
8973 Assessment. If the TOE is required to be resistant against attackers of a certain potential in its
8974 intended environment, this should also apply to the delivery of the TOE. The evaluator should
8975 determine that a balanced approach has been taken, such that delivery does not present a
8976 weak point in an otherwise secure development process.

8977 — The security objectives provided by the ST. The emphasis in the delivery documentation is
8978 likely to be on measures related to integrity, as integrity of the TOE is always important.

8979 However, confidentiality and availability of the delivery will be of concern in the delivery of
8980 some TOEs; procedures relating to these aspects of the secure delivery should also be
8981 discussed in the procedures.

8982 **13.4.1.4 Implied evaluator action**

8983 ISO/IEC 15408-3 ALC_DEL.1.2D: *The developer shall use the delivery procedures.*

8984 **13.4.1.4.1 Work unit ALC_DEL.1-2**

8985 The evaluator **shall examine** aspects of the delivery process to determine that the delivery
8986 procedures are used.

8987 The approach taken by the evaluator to check the application of delivery procedures will depend
8988 on the nature of the TOE, and the delivery process itself. In addition to examination of the
8989 procedures themselves, the evaluator seeks some assurance that they are applied in practise. Some
8990 possible approaches are:

8991 a) a visit to the distribution site(s) where practical application of the procedures may be
8992 observed;

8993 b) examination of the TOE at some stage during delivery, or after the user has received it (e.g.
8994 checking for tamper proof seals);

8995 c) observing that the process is applied in practise when the evaluator obtains the TOE
8996 through regular channels;

8997 d) questioning end users as to how the TOE was delivered.

8998 For guidance on site visits see A.4, Site Visits.

8999 It may be the case of a newly developed TOE that the delivery procedures have yet to be exercised.
9000 In these cases, the evaluator has to be satisfied that appropriate procedures and facilities are in
9001 place for future deliveries and that all personnel involved are aware of their responsibilities. The
9002 evaluator may request a “dry run” of a delivery if this is practical. If the developer has produced
9003 other similar products, then an examination of procedures in their use may be useful in providing
9004 assurance.

9005 **13.5 Development security (ALC_DVS)**

9006 **13.5.1 Evaluation of sub-activity (ALC_DVS.1)**

9007 **13.5.1.1 Objectives**

9008 The objective of this sub-activity is to determine whether the developer's security controls on the
9009 development environment are adequate to provide the confidentiality and integrity of the TOE
9010 design and implementation that is necessary to ensure that secure operation of the TOE is not
9011 compromised.

9012 **13.5.1.2 Input**

9013 The evaluation evidence for this sub-activity is:

9014 a) the ST;

9015 b) the development security documentation.

9016 In addition, the evaluator may need to examine other deliverables to determine that the security
9017 controls are well-defined and followed. Specifically, the evaluator may need to examine the
9018 developer's configuration management documentation (the input for the Evaluation of sub-activity
9019 (ALC_CMC.4) "Production support and acceptance procedures" and the Evaluation of sub-activity
9020 (ALC_CMS.4) "Problem tracking CM coverage"). Evidence that the procedures are being applied is
9021 also required.

9022 **13.5.1.3 Action ALC_DVS.1.1E**

9023 ISO/IEC 15408-3 ALC_DVS.1.1C: *The development security documentation shall describe all the*
9024 *physical, procedural, personnel, and other security measures that are necessary to protect the*
9025 *confidentiality and integrity of the TOE design and implementation in its development environment.*

9026 **13.5.1.3.1 Work unit ALC_DVS.1-1**

9027 The evaluator **shall examine** the development security documentation to determine that it details
9028 all security measures used in the development environment that are necessary to protect the
9029 confidentiality and integrity of the TOE design and implementation.

9030 The evaluator determines what is necessary by first referring to the ST for any information that
9031 may assist in the determination of necessary protection.

9032 If no explicit information is available from the ST the evaluator will need to make a determination
9033 of the necessary measures. In cases where the developer's measures are considered less than what
9034 is necessary, a clear justification should be provided for the assessment, based on a potential
9035 exploitable vulnerability.

9036 The following types of security measures are considered by the evaluator when examining the
9037 documentation:

- 9038 a) physical, for example physical access controls used to prevent unauthorised access to the
9039 TOE development environment (during normal working hours and at other times);
- 9040 b) procedural, for example covering:
- 9041 • granting of access to the development environment or to specific parts of the environment
9042 such as development machines
 - 9043 • revocation of access rights when a person leaves the development team
 - 9044 • transfer of protected material within and out of the development environment and between
9045 different development sites in accordance with defined acceptance procedures
 - 9046 • admitting and escorting visitors to the development environment
 - 9047 • roles and responsibilities in ensuring the continued application of security measures, and
9048 the detection of security breaches.
- 9049 c) personnel, for example any controls or checks made to establish the trustworthiness of
9050 new development staff;
- 9051 d) other security measures, for example the logical protections on any development
9052 machines.

9053 The development security documentation should identify the locations at which development
9054 occurs, and describe the aspects of development performed, along with the security measures
9055 applied at each location and for transports between different locations. For example, development

9056 could occur at multiple facilities within a single building, multiple buildings at the same site, or at
 9057 multiple sites. Transports of parts of the TOE or the unfinished TOE between different
 9058 development sites are to be covered by Development security (ALC_DVS), whereas the transport of
 9059 the finished TOE to the consumer is dealt with in Delivery (ALC_DEL).

9060 Development includes the production of the TOE.

9061 **13.5.1.3.2 Work unit ALC_DVS.1-2**

9062 The evaluator *shall examine* the development confidentiality and integrity policies in order to
 9063 determine the sufficiency of the security measures employed.

9064 The evaluator should examine whether the following is included in the policies:

9065 a) what information relating to the TOE development needs to be kept confidential, and
 9066 which members of the development staff are allowed to access such material;

9067 b) what material must be protected from unauthorised modification in order to preserve the
 9068 integrity of the TOE, and which members of the development staff are allowed to modify
 9069 such material.

9070 The evaluator should determine that these policies are described in the development security
 9071 documentation, that the security measures employed are consistent with the policies, and that they
 9072 are complete.

9073 It should be noted that configuration management procedures will help protect the integrity of the
 9074 TOE and the evaluator should avoid overlap with the work-units conducted for the CM capabilities
 9075 (ALC_CMC). For example, the CM documentation may describe the security procedures necessary
 9076 for controlling the roles or individuals who should have access to the development environment
 9077 and who may modify the TOE.

9078 Whereas the CM capabilities (ALC_CMC) requirements are fixed, those for the Development
 9079 security (ALC_DVS), mandating only necessary measures, are dependent on the nature of the TOE,
 9080 and on information that may be provided in the ST. The evaluators would then determine that such
 9081 a policy had been applied under this sub-activity.

9082 **13.5.1.4 Action ALC_DVS.1.2E**

9083 **13.5.1.4.1 Work unit ALC_DVS.1-3**

9084 The evaluator *shall examine* the development security documentation and associated evidence to
 9085 determine that the security measures are being applied.

9086 This work unit requires the evaluator to determine that the security measures described in the
 9087 development security documentation are being followed, such that the integrity of the TOE and the
 9088 confidentiality of associated documentation is being adequately protected. For example, this could
 9089 be determined by examination of the documentary evidence provided. Documentary evidence
 9090 should be supplemented by visiting the development environment. A visit to the development
 9091 environment will allow the evaluator to:

9092 a) observe the application of security measures (e.g. physical measures);

9093 b) examine documentary evidence of application of procedures;

9094 c) interview development staff to check awareness of the development security policies and
 9095 procedures, and their responsibilities.

9096 A development site visit is a useful means of gaining confidence in the measures being used. Any
9097 decision not to make such a visit should be determined in consultation with the evaluation
9098 authority.

9099 For guidance on site visits see A.4, Site Visits.

9100 **13.5.2 Evaluation of sub-activity (ALC_DVS.2)**

9101 **13.5.2.1 Objectives**

9102 The objective of this sub-activity is to determine whether the developer's security controls on the
9103 development environment are adequate to provide the confidentiality and integrity of the TOE
9104 design and implementation that is necessary to ensure that secure operation of the TOE is not
9105 compromised. Additionally, sufficiency of the measures as applied is intended be justified.

9106 **13.5.2.2 Input**

9107 The evaluation evidence for this sub-activity is:

9108 a) the ST;

9109 b) the development security documentation.

9110 In addition, the evaluator may need to examine other deliverables to determine that the security
9111 controls are well-defined and followed. Specifically, the evaluator may need to examine the
9112 developer's configuration management documentation (the input for the Evaluation of sub-activity
9113 (ALC_CMC.4) "Production support and acceptance procedures" and the Evaluation of sub-activity
9114 (ALC_CMS.4) "Problem tracking CM coverage"). Evidence that the procedures are being applied is
9115 also required.

9116 **13.5.2.3 Action ALC_DVS.2.1E**

9117 ISO/IEC 15408-3 ALC_DVS.2.1C: *The development security documentation shall describe all the*
9118 *physical, procedural, personnel, and other security measures that are necessary to protect the*
9119 *confidentiality and integrity of the TOE design and implementation in its development environment.*

9120 **13.5.2.3.1 Work unit ALC_DVS.2-1**

9121 The evaluator ***shall examine*** the development security documentation to determine that it details
9122 all security measures used in the development environment that are necessary to protect the
9123 confidentiality and integrity of the TOE design and implementation.

9124 The evaluator determines what is necessary by first referring to the ST for any information that
9125 may assist in the determination of necessary protection.

9126 If no explicit information is available from the ST the evaluator will need to make a determination
9127 of the necessary measures. In cases where the developer's measures are considered less than what
9128 is necessary, a clear justification should be provided for the assessment, based on a potential
9129 exploitable vulnerability.

9130 The following types of security measures are considered by the evaluator when examining the
9131 documentation:

9132 a) physical, for example physical access controls used to prevent unauthorised access to the
9133 TOE development environment (during normal working hours and at other times);

9134 b) procedural, for example covering:

- 9135 • granting of access to the development environment or to specific parts of the environment
9136 such as development machines
- 9137 • revocation of access rights when a person leaves the development team
- 9138 • transfer of protected material out of the development environment and between different
9139 development sites in accordance with defined acceptance procedures
- 9140 • admitting and escorting visitors to the development environment
- 9141 • roles and responsibilities in ensuring the continued application of security measures, and
9142 the detection of security breaches.
- 9143 c) personnel, for example any controls or checks made to establish the trustworthiness of
9144 new development staff;
- 9145 d) other security measures, for example the logical protections on any development
9146 machines.
- 9147 The development security documentation should identify the locations at which development
9148 occurs, and describe the aspects of development performed, along with the security measures
9149 applied at each location and for transports between different locations. For example, development
9150 could occur at multiple facilities within a single building, multiple buildings at the same site, or at
9151 multiple sites. Transports of parts of the TOE or the unfinished TOE between different
9152 development sites are to be covered by the Development security (ALC_DVS), whereas the
9153 transport of the finished TOE to the consumer is dealt with in the Delivery (ALC_DEL).
- 9154 Development includes the production of the TOE.
- 9155 ISO/IEC 15408-3 ALC_DVS.2.2C: *The development security documentation shall justify that the*
9156 *security measures provide the necessary level of protection to maintain the confidentiality and*
9157 *integrity of the TOE.*
- 9158 **13.5.2.3.2 Work unit ALC_DVS.2-2**
- 9159 The evaluator ***shall examine*** the development security documentation to determine that an
9160 appropriate justification is given why the security measures provide the necessary level of
9161 protection to maintain the confidentiality and integrity of the TOE.
- 9162 Since attacks on the TOE or its related information are assumed in different design and production
9163 stages, measures and procedures need to have an appropriate level necessary to prevent those
9164 attacks or to make them more difficult.
- 9165 Since this level depends on the overall attack potential claimed for the TOE (cf. the Vulnerability
9166 analysis (AVA_VAN) component chosen), the development security documentation should justify
9167 the necessary level of protection to maintain the confidentiality and integrity of the TOE. This level
9168 has to be achieved by the security measures applied.
- 9169 The concept of protection measures should be consistent, and the justification should include an
9170 analysis of how the measures are mutually supportive. All aspects of development and production
9171 on all the different sites with all roles involved up to delivery of the TOE should be analysed.
- 9172 Justification may include an analysis of potential vulnerabilities taking the applied security
9173 measures into account.
- 9174 There may be a convincing argument showing that e.g.

9175 — The technical measures and mechanisms of the developer's infrastructure are sufficient for
9176 keeping the appropriate security level (e.g. cryptographic mechanisms as well as physical
9177 protection mechanisms, properties of the CM system (cf. ALC_CMC.4-5));

9178 — The system containing the implementation representation of the TOE (including concerning
9179 guidance documents) provides effective protection against logical attacks e.g. by "Trojan" code
9180 or viruses. It might be adequate, if the implementation representation is kept on an isolated
9181 system where only the software necessary to maintain it is installed and where no additional
9182 software is installed afterwards.

9183 — Data brought into this system need to be carefully considered to prevent the installation of
9184 hidden functionality onto the system. The effectiveness of these measures need to be tested, e.g.
9185 by independently trying to get access to the machine, install some additional executable
9186 (program, macro etc.) or get some information out of the machine using logical attacks.

9187 — The appropriate organisational (procedural and personal) measures are unconditionally
9188 enforced.

9189 **13.5.2.3.3 Work unit ALC_DVS.2-3**

9190 The evaluator *shall examine* the development confidentiality and integrity policies in order to
9191 determine the sufficiency of the security measures employed.

9192 The evaluator should examine whether the following is included in the policies:

9193 a) what information relating to the TOE development needs to be kept confidential, and
9194 which members of the development staff are allowed to access such material;

9195 b) what material must be protected from unauthorised modification in order to preserve the
9196 integrity of the TOE, and which members of the development staff are allowed to modify
9197 such material.

9198 The evaluator should determine that these policies are described in the development security
9199 documentation, that the security measures employed are consistent with the policies, and that they
9200 are complete.

9201 It should be noted that configuration management procedures will help protect the integrity of the
9202 TOE and the evaluator should avoid overlap with the work-units conducted for the CM capabilities
9203 (ALC_CMC). For example, the CM documentation may describe the security procedures necessary
9204 for controlling the roles or individuals who should have access to the development environment
9205 and who may modify the TOE.

9206 Whereas the CM capabilities (ALC_CMC) requirements are fixed, those for the Development
9207 security (ALC_DVS), mandating only necessary measures, are dependent on the nature of the TOE,
9208 and on information that may be provided in the ST. For example, the ST may identify a security
9209 objective for the development environment that requires the TOE to be developed by staff that has
9210 security clearance. The evaluators would then determine that such a policy had been applied under
9211 this sub-activity.

9212 **13.5.2.4 Action ALC_DVS.2.2E**

9213 **13.5.2.4.1 Work unit ALC_DVS.2-4**

9214 The evaluator *shall examine* the development security documentation and associated evidence to
9215 determine that the security measures are being applied.

9216 This work unit requires the evaluator to determine that the security measures described in the
9217 development security documentation are being followed, such that the integrity of the TOE and the

9218 confidentiality of associated documentation is being adequately protected. For example, this could
 9219 be determined by examination of the documentary evidence provided. Documentary evidence
 9220 should be supplemented by visiting the development environment. A visit to the development
 9221 environment will allow the evaluator to:

- 9222 a) observe the application of security measures (e.g. physical measures);
- 9223 b) examine documentary evidence of application of procedures;
- 9224 c) interview development staff to check awareness of the development security policies and
 9225 procedures, and their responsibilities.

9226 A development site visit is a useful means of gaining confidence in the measures being used. Any
 9227 decision not to make such a visit should be determined in consultation with the evaluation
 9228 authority.

9229 For guidance on site visits see A.4, Site Visits.

9230 **13.6 Flaw remediation (ALC_FLR)**

9231 **13.6.1 Evaluation of sub-activity (ALC_FLR.1)**

9232 **13.6.1.1 Objectives**

9233 The objective of this sub-activity is to determine whether the developer has established flaw
 9234 remediation procedures that describe the tracking of security flaws, the identification of corrective
 9235 actions, and the distribution of corrective action information to TOE users.

9236 **13.6.1.2 Input**

9237 The evaluation evidence for this sub-activity is:

- 9238 a) the flaw remediation procedures documentation.

9239 **13.6.1.3 Action ALC_FLR.1.1E**

9240 ISO/IEC 15408-3 ALC_FLR.1.1C: *The flaw remediation procedures documentation shall describe the*
 9241 *procedures used to track all reported security flaws in each release of the TOE.*

9242 **13.6.1.3.1 Work unit ALC_FLR.1-1**

9243 The evaluator **shall examine** the flaw remediation procedures documentation to determine that it
 9244 describes the procedures used to track all reported security flaws in each release of the TOE.

9245 The procedures describe the actions that are taken by the developer from the time each suspected
 9246 security flaw is reported to the time that it is resolved. This includes the flaw's entire time frame,
 9247 from initial detection through ascertaining that the flaw is a security flaw, to resolution of the
 9248 security flaw.

9249 If a flaw is discovered not to be security-relevant, there is no need (for the purposes of the Flaw
 9250 remediation (ALC_FLR) requirements) for the flaw remediation procedures to track it further; only
 9251 that there be an explanation of why the flaw is not security-relevant.

9252 While these requirements do not mandate that there be a publicised means for TOE users to report
 9253 security flaws, they do mandate that all security flaws that are reported be tracked. That is, a
 9254 reported security flaw cannot be ignored simply because it comes from outside the developer's
 9255 organisation.

9256 ISO/IEC 15408-3 ALC_FLR.1.2C: *The flaw remediation procedures shall require that a description of*
9257 *the nature and effect of each security flaw be provided, as well as the status of finding a correction to*
9258 *that flaw.*

9259 **13.6.1.3.2 Work unit ALC_FLR.1-2**

9260 The evaluator **shall examine** the flaw remediation procedures to determine that the application of
9261 these procedures would produce a description of each security flaw in terms of its nature and
9262 effects.

9263 The procedures identify the actions that are taken by the developer to describe the nature and
9264 effects of each security flaw in sufficient detail to be able to reproduce it. The description of the
9265 nature of a security flaw addresses whether it is an error in the documentation, a flaw in the design
9266 of the TSF, a flaw in the implementation of the TSF, etc. The description of the security flaw's
9267 effects identifies the portions of the TSF that are affected and how those portions are affected. For
9268 example, a security flaw in the implementation might be found that affects the identification and
9269 authentication enforced by the TSF by permitting authentication with the password "BACK DOOR".

9270 **13.6.1.3.3 Work unit ALC_FLR.1-3**

9271 The evaluator **shall examine** the flaw remediation procedures to determine that the application of
9272 these procedures would identify the status of finding a correction to each security flaw.

9273 The flaw remediation procedures identify the different stages of security flaws. This differentiation
9274 includes at least: suspected security flaws that have been reported, suspected security flaws that
9275 have been confirmed to be security flaws, and security flaws whose solutions have been
9276 implemented. It is permissible that additional stages (e.g. flaws that have been reported but not yet
9277 investigated, flaws that are under investigation, security flaws for which a solution has been found
9278 but not yet implemented) be included.

9279 ISO/IEC 15408-3 ALC_FLR.1.3C: *The flaw remediation procedures shall require that corrective*
9280 *actions be identified for each of the security flaws.*

9281 **13.6.1.3.4 Work unit ALC_FLR.1-4**

9282 The evaluator **shall check** the flaw remediation procedures to determine that the application of
9283 these procedures would identify the corrective action for each security flaw.

9284 *Corrective action* may consist of a repair to the hardware, firmware, or software portions of the
9285 TOE, a modification of TOE guidance, or both. Corrective action that constitutes modifications to
9286 TOE guidance (e.g. details of procedural measures to be taken to obviate the security flaw) includes
9287 both those measures serving as only an interim solution (until the repair is issued) as well as those
9288 serving as a permanent solution (where it is determined that the procedural measure is the best
9289 solution).

9290 If the source of the security flaw is a documentation error, the corrective action consists of an
9291 update of the affected TOE guidance. If the corrective action is a procedural measure, this measure
9292 will include an update made to the affected TOE guidance to reflect these corrective procedures.

9293 ISO/IEC 15408-3 ALC_FLR.1.4C: *The flaw remediation procedures documentation shall describe the*
9294 *methods used to provide flaw information, corrections and guidance on corrective actions to TOE*
9295 *users.*

9296 **13.6.1.3.5 Work unit ALC_FLR.1-5**

9297 The evaluator **shall examine** the flaw remediation procedures documentation to determine that it
9298 describes a means of providing the TOE users with the necessary information on each security flaw.

9309 The *necessary information* about each security flaw consists of its description (not necessarily at the same level of detail as that provided as part of work unit ALC_FLR.1-2), the prescribed corrective action, and any associated guidance on implementing the correction.

9302 TOE users may be provided with such information, correction, and documentation updates in any of several ways, such as their posting to a website, their being sent to TOE users, or arrangements made for the developer to install the correction. In cases where the means of providing this information requires action to be initiated by the TOE user, the evaluator examines any TOE guidance to ensure that it contains instructions for retrieving the information.

9307 The only metric for assessing the adequacy of the method used for providing the information, corrections and guidance is that there be a reasonable expectation that TOE users can obtain or receive it. For example, consider the method of dissemination where the requisite data is posted to a website for one month, and the TOE users know that this will happen and when this will happen. This may not be especially reasonable or effective (as, say, a permanent posting to the website), yet it is feasible that the TOE user could obtain the necessary information. On the other hand, if the information were posted to the website for only one hour, yet TOE users had no way of knowing this or when it would be posted, it is infeasible that they would ever get the necessary information.

9315 **13.6.2 Evaluation of sub-activity (ALC_FLR.2)**

9316 **13.6.2.1 Objectives**

9317 The objective of this sub-activity is to determine whether the developer has established flaw remediation procedures that describe the tracking of security flaws, the identification of corrective actions, and the distribution of corrective action information to TOE users. Additionally, this sub-activity determines whether the developer's procedures provide for the corrections of security flaws, for the receipt of flaw reports from TOE users, and for assurance that the corrections introduce no new security flaws.

9323 In order for the developer to be able to act appropriately upon security flaw reports from TOE users, TOE users need to understand how to submit security flaw reports to the developer, and developers need to know how to receive these reports. Flaw remediation guidance addressed to the TOE user ensures that TOE users are aware of how to communicate with the developer; flaw remediation procedures describe the developer's role in such communication

9328 **13.6.2.2 Input**

9329 The evaluation evidence for this sub-activity is:

9330 a) the flaw remediation procedures documentation;

9331 b) flaw remediation guidance documentation.

9332 **13.6.2.3 Action ALC_FLR.2.1E**

9333 ISO/IEC 15408-3 ALC_FLR.2.1C: *The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.*

9335 **13.6.2.3.1 Work unit ALC_FLR.2-1**

9336 The evaluator *shall examine* the flaw remediation procedures documentation to determine that it describes the procedures used to track all reported security flaws in each release of the TOE.

9338 The procedures describe the actions that are taken by the developer from the time each suspected security flaw is reported to the time that it is resolved. This includes the flaw's entire time frame, from initial detection through ascertaining that the flaw is a security flaw, to resolution of the security flaw.

9342 If a flaw is discovered not to be security-relevant, there is no need (for the purposes of the Flaw
9343 remediation (ALC_FLR) requirements) for the flaw remediation procedures to track it further; only
9344 that there be an explanation of why the flaw is not security-relevant.

9345 ISO/IEC 15408-3 ALC_FLR.2.2C: *The flaw remediation procedures shall require that a description of*
9346 *the nature and effect of each security flaw be provided, as well as the status of finding a correction to*
9347 *that flaw.*

9348 **13.6.2.3.2 Work unit ALC_FLR.2-2**

9349 The evaluator **shall examine** the flaw remediation procedures to determine that the application of
9350 these procedures would produce a description of each security flaw in terms of its nature and
9351 effects.

9352 The procedures identify the actions that are taken by the developer to describe the nature and
9353 effects of each security flaw in sufficient detail to be able to reproduce it. The description of the
9354 nature of a security flaw addresses whether it is an error in the documentation, a flaw in the design
9355 of the TSF, a flaw in the implementation of the TSF, etc. The description of the security flaw's
9356 effects identifies the portions of the TSF that are affected and how those portions are affected. For
9357 example, a security flaw in the implementation might be found that affects the identification and
9358 authentication enforced by the TSF by permitting authentication with the password "BACKDOOR".

9359 **13.6.2.3.3 Work unit ALC_FLR.2-3**

9360 The evaluator **shall examine** the flaw remediation procedures to determine that the application of
9361 these procedures would identify the status of finding a correction to each security flaw.

9362 The flaw remediation procedures identify the different stages of security flaws. This differentiation
9363 includes at least: suspected security flaws that have been reported, suspected security flaws that
9364 have been confirmed to be security flaws, and security flaws whose solutions have been
9365 implemented. It is permissible that additional stages (e.g. flaws that have been reported but not yet
9366 investigated, flaws that are under investigation, security flaws for which a solution has been found
9367 but not yet implemented) be included.

9368 ISO/IEC 15408-3 ALC_FLR.2.3C: *The flaw remediation procedures shall require that corrective*
9369 *actions be identified for each of the security flaws.*

9370 **13.6.2.3.4 Work unit ALC_FLR.2-4**

9371 The evaluator **shall check** the flaw remediation procedures to determine that the application of
9372 these procedures would identify the corrective action for each security flaw.

9373 *Corrective action* may consist of a repair to the hardware, firmware, or software portions of the
9374 TOE, a modification of TOE guidance, or both. Corrective action that constitutes modifications to
9375 TOE guidance (e.g. details of procedural measures to be taken to obviate the security flaw) includes
9376 both those measures serving as only an interim solution (until the repair is issued) as well as those
9377 serving as a permanent solution (where it is determined that the procedural measure is the best
9378 solution).

9379 If the source of the security flaw is a documentation error, the corrective action consists of an
9380 update of the affected TOE guidance. If the corrective action is a procedural measure, this measure
9381 will include an update made to the affected TOE guidance to reflect these corrective procedures.

9382 ISO/IEC 15408-3 ALC_FLR.2.4C: *The flaw remediation procedures documentation shall describe the*
9383 *methods used to provide flaw information, corrections and guidance on corrective actions to TOE*
9384 *users.*

9385 **13.6.2.3.5 Work unit ALC_FLR.2-5**

9386 The evaluator *shall examine* the flaw remediation procedures documentation to determine that it
9387 describes a means of providing the TOE users with the necessary information on each security flaw.

9388 *The necessary information* about each security flaw consists of its description (not necessarily at
9389 the same level of detail as that provided as part of work unit ALC_FLR.2-2), the prescribed
9390 corrective action, and any associated guidance on implementing the correction.

9391 TOE users may be provided with such information, correction, and documentation updates in any
9392 of several ways, such as their posting to a website, their being sent to TOE users, or arrangements
9393 made for the developer to install the correction. In cases where the means of providing this
9394 information requires action to be initiated by the TOE user, the evaluator examines any TOE
9395 guidance to ensure that it contains instructions for retrieving the information.

9396 The only metric for assessing the adequacy of the method used for providing the information,
9397 corrections and guidance is that there be a reasonable expectation that TOE users can obtain or
9398 receive it. For example, consider the method of dissemination where the requisite data is posted to
9399 a website for one month, and the TOE users know that this will happen and when this will happen.
9400 This may not be especially reasonable or effective (as, say, a permanent posting to the website), yet
9401 it is feasible that the TOE user could obtain the necessary information. On the other hand, if the
9402 information were posted to the website for only one hour, yet TOE users had no way of knowing
9403 this or when it would be posted, it is infeasible that they would ever get the necessary information.

9404 ISO/IEC 15408-3 ALC_FLR.2.5C: *The flaw remediation procedures shall describe a means by which*
9405 *the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.*

9406 **13.6.2.3.6 Work unit ALC_FLR.2-6**

9407 The evaluator *shall examine* the flaw remediation procedures to determine that they describe
9408 procedures for the developer to accept reports of security flaws or requests for corrections to such
9409 flaws.

9410 The procedures ensure that TOE users have a means by which they can communicate with the TOE
9411 developer. By having a means of contact with the developer, the user can report security flaws,
9412 enquire about the status of security flaws, or request corrections to flaws. This means of contact
9413 may be part of a more general contact facility for reporting non-security related problems.

9414 The use of these procedures is not restricted to TOE users; however, only the TOE users are
9415 actively supplied with the details of these procedures. Others who might have access to or
9416 familiarity with the TOE can use the same procedures to submit reports to the developer, who is
9417 then expected to process them. Any means of submitting reports to the developer, other than those
9418 identified by the developer, are beyond the scope of this work unit; reports generated by other
9419 means need not be addressed.

9420 ISO/IEC 15408-3 ALC_FLR.2.6C: *The procedures for processing reported security flaws shall ensure*
9421 *that any reported flaws are remediated and the remediation procedures issued to TOE users.*

9422 **13.6.2.3.7 Work unit ALC_FLR.2-7**

9423 The evaluator *shall examine* the flaw remediation procedures to determine that the application of
9424 these procedures would help to ensure every reported flaw is corrected.

9425 The flaw remediation procedures cover not only those security flaws discovered and reported by
9426 developer personnel, but also those reported by TOE users. The procedures are sufficiently
9427 detailed so that they describe how it is ensured that each reported security flaw is corrected. The
9428 procedures contain reasonable steps that show progress leading to the eventual, inevitable
9429 resolution.

9430 The procedures describe the process that is taken from the point at which the suspected security
9431 flaw is determined to be a security flaw to the point at which it is resolved.

9432 **13.6.2.3.8 Work unit ALC_FLR.2-8**

9433 The evaluator *shall examine* the flaw remediation procedures to determine that the application of
9434 these procedures would help to ensure that the TOE users are issued remediation procedures for
9435 each security flaw.

9436 The procedures describe the process that is taken from the point at which a security flaw is
9437 resolved to the point at which the remediation procedures are provided. The procedures for
9438 delivering corrective actions should be consistent with the security objectives; they need not
9439 necessarily be identical to the procedures used for delivering the TOE, as documented to meet
9440 ALC_DEL, if included in the assurance requirements. For example, if the hardware portion of a TOE
9441 were originally delivered by bonded courier, updates to hardware resulting from flaw remediation
9442 would likewise be expected to be distributed by bonded courier. Updates unrelated to flaw
9443 remediation would follow the procedures set forth in the documentation meeting the Delivery
9444 (ALC_DEL) requirements.

9445 ISO/IEC 15408-3 ALC_FLR.2.7C: *The procedures for processing reported security flaws shall provide*
9446 *safeguards that any corrections to these security flaws do not introduce any new flaws.*

9447 **13.6.2.3.9 Work unit ALC_FLR.2-9**

9448 The evaluator *shall examine* the flaw remediation procedures to determine that the application of
9449 these procedures would result in safeguards that the potential correction contains no adverse
9450 effects.

9451 Through analysis, testing, or a combination of the two, the developer may reduce the likelihood
9452 that adverse effects will be introduced when a security flaw is corrected. The evaluator assesses
9453 whether the procedures provide detail in how the necessary mix of analysis and testing actions is
9454 to be determined for a given correction.

9455 The evaluator also determines that, for instances where the source of the security flaw is a
9456 documentation problem, the procedures include the means of safeguarding against the
9457 introduction of contradictions with other documentation.

9458 ISO/IEC 15408-3 ALC_FLR.2.8C: *The flaw remediation guidance shall describe a means by which TOE*
9459 *users report to the developer any suspected security flaws in the TOE.*

9460 **13.6.2.3.10 Work unit ALC_FLR.2-10**

9461 The evaluator *shall examine* the flaw remediation guidance to determine that the application of
9462 these procedures would result in a means for the TOE user to provide reports of suspected security
9463 flaws or requests for corrections to such flaws.

9464 The guidance ensures that TOE users have a means by which they can communicate with the TOE
9465 developer. By having a means of contact with the developer, the user can report security flaws,
9466 enquire about the status of security flaws, or request corrections to flaws.

9467 **13.6.3 Evaluation of sub-activity (ALC_FLR.3)**

9468 **13.6.3.1 Objectives**

9469 The objective of this sub-activity is to determine whether the developer has established flaw
9470 remediation procedures that describe the tracking of security flaws, the identification of corrective
9471 actions, and the distribution of corrective action information to TOE users. Additionally, this sub-
9472 activity determines whether the developer's procedures provide for the corrections of security

9473 flaws, for the receipt of flaw reports from TOE users, for assurance that the corrections introduce
 9474 no new security flaws, for the establishment of a point of contact for each TOE user, and for the
 9475 timely issue of corrective actions to TOE users.

9476 In order for the developer to be able to act appropriately upon security flaw reports from TOE
 9477 users, TOE users need to understand how to submit security flaw reports to the developer, and
 9478 developers need to know how to receive these reports. Flaw remediation guidance addressed to
 9479 the TOE user ensures that TOE users are aware of how to communicate with the developer; flaw
 9480 remediation procedures describe the developer's role in such communication.

9481 **13.6.3.2 Input**

9482 The evaluation evidence for this sub-activity is:

- 9483 a) the flaw remediation procedures documentation;
- 9484 b) flaw remediation guidance documentation.

9485 **13.6.3.3 Action ALC_FLR.3.1E**

9486 ISO/IEC 15408-3 ALC_FLR.3.1C: *The flaw remediation procedures documentation shall describe the*
 9487 *procedures used to track all reported security flaws in each release of the TOE.*

9488 **13.6.3.3.1 Work unit ALC_FLR.3-1**

9489 The evaluator **shall examine** the flaw remediation procedures documentation to determine that it
 9490 describes the procedures used to track all reported security flaws in each release of the TOE.

9491 The procedures describe the actions that are taken by the developer from the time each suspected
 9492 security flaw is reported to the time that it is resolved. This includes the flaw's entire time frame,
 9493 from initial detection through ascertaining that the flaw is a security flaw, to resolution of the
 9494 security flaw.

9495 If a flaw is discovered not to be security-relevant, there is no need (for the purposes of the Flaw
 9496 remediation (ALC_FLR) requirements) for the flaw remediation procedures to track it further; only
 9497 that there be an explanation of why the flaw is not security-relevant.

9498 ISO/IEC 15408-3 ALC_FLR.3.2C: *The flaw remediation procedures shall require that a description of*
 9499 *the nature and effect of each security flaw be provided, as well as the status of finding a correction to*
 9500 *that flaw.*

9501 **13.6.3.3.2 Work unit ALC_FLR.3-2**

9502 The evaluator **shall examine** the flaw remediation procedures to determine that the application of
 9503 these procedures would produce a description of each security flaw in terms of its nature and
 9504 effects.

9505 The procedures identify the actions that are taken by the developer to describe the nature and
 9506 effects of each security flaw in sufficient detail to be able to reproduce it. The description of the
 9507 nature of a security flaw addresses whether it is an error in the documentation, a flaw in the design
 9508 of the TSF, a flaw in the implementation of the TSF, etc. The description of the security flaw's
 9509 effects identifies the portions of the TSF that are affected and how those portions are affected. For
 9510 example, a security flaw in the implementation might be found that affects the identification and
 9511 authentication enforced by the TSF by permitting authentication with the password "BACKDOOR".

9512 **13.6.3.3.3 Work unit ALC_FLR.3-3**

9513 The evaluator *shall examine* the flaw remediation procedures to determine that the application of
9514 these procedures would identify the status of finding a correction to each security flaw.

9515 The flaw remediation procedures identify the different stages of security flaws. This differentiation
9516 includes at least: suspected security flaws that have been reported, suspected security flaws that
9517 have been confirmed to be security flaws, and security flaws whose solutions have been
9518 implemented. It is permissible that additional stages (e.g. flaws that have been reported but not yet
9519 investigated, flaws that are under investigation, security flaws for which a solution has been found
9520 but not yet implemented) be included.

9521 ISO/IEC 15408-3 ALC_FLR.3.3C: *The flaw remediation procedures shall require that corrective*
9522 *actions be identified for each of the security flaws.*

9523 **13.6.3.3.4 Work unit ALC_FLR.3-4**

9524 The evaluator *shall check* the flaw remediation procedures to determine that the application of
9525 these procedures would identify the corrective action for each security flaw.

9526 *Corrective action* may consist of a repair to the hardware, firmware, or software portions of the
9527 TOE, a modification of TOE guidance, or both. Corrective action that constitutes modifications to
9528 TOE guidance (e.g. details of procedural measures to be taken to obviate the security flaw) includes
9529 both those measures serving as only an interim solution (until the repair is issued) as well as those
9530 serving as a permanent solution (where it is determined that the procedural measure is the best
9531 solution).

9532 If the source of the security flaw is a documentation error, the corrective action consists of an
9533 update of the affected TOE guidance. If the corrective action is a procedural measure, this measure
9534 will include an update made to the affected TOE guidance to reflect these corrective procedures.

9535 ISO/IEC 15408-3 ALC_FLR.3.4C: *The flaw remediation procedures documentation shall describe the*
9536 *methods used to provide flaw information, corrections and guidance on corrective actions to TOE*
9537 *users.*

9538 **13.6.3.3.5 Work unit ALC_FLR.3-5**

9539 The evaluator *shall examine* the flaw remediation procedures documentation to determine that it
9540 describes a means of providing the TOE users with the necessary information on each security flaw.

9541 *The necessary information* about each security flaw consists of its description (not necessarily at
9542 the same level of detail as that provided as part of work unit ALC_FLR.3-2), the prescribed
9543 corrective action, and any associated guidance on implementing the correction.

9544 TOE users may be provided with such information, correction, and documentation updates in any
9545 of several ways, such as their posting to a website, their being sent to TOE users, or arrangements
9546 made for the developer to install the correction. In cases where the means of providing this
9547 information requires action to be initiated by the TOE user, the evaluator examines any TOE
9548 guidance to ensure that it contains instructions for retrieving the information.

9549 The only metric for assessing the adequacy of the method used for providing the information,
9550 corrections and guidance is that there be a reasonable expectation that TOE users can obtain or
9551 receive it. For example, consider the method of dissemination where the requisite data is posted to
9552 a website for one month, and the TOE users know that this will happen and when this will happen.
9553 This may not be especially reasonable or effective (as, say, a permanent posting to the website), yet
9554 it is feasible that the TOE user could obtain the necessary information. On the other hand, if the
9555 information were posted to the website for only one hour, yet TOE users had no way of knowing
9556 this or when it would be posted, it is infeasible that they would ever get the necessary information.

9557 For TOE users who register with the developer (see work unit ALC_FLR.3-12), the passive
9558 availability of this information is not sufficient. Developers must actively send the information (or a
9559 notification of its availability) to registered TOE users.

9560 ISO/IEC 15408-3 ALC_FLR.3.5C: *The flaw remediation procedures shall describe a means by which*
9561 *the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.*

9562 **13.6.3.3.6 Work unit ALC_FLR.3-6**

9563 The evaluator ***shall examine*** the flaw remediation procedures to determine that the application of
9564 these procedures would result in a means for the developer to receive from TOE user reports of
9565 suspected security flaws or requests for corrections to such flaws.

9566 The procedures ensure that TOE users have a means by which they can communicate with the TOE
9567 developer. By having a means of contact with the developer, the user can report security flaws,
9568 enquire about the status of security flaws, or request corrections to flaws. This means of contact
9569 may be part of a more general contact facility for reporting non-security related problems.

9570 The use of these procedures is not restricted to TOE users; however, only the TOE users are
9571 actively supplied with the details of these procedures. Others who might have access to or
9572 familiarity with the TOE can use the same procedures to submit reports to the developer, who is
9573 then expected to process them. Any means of submitting reports to the developer, other than those
9574 identified by the developer, are beyond the scope of this work unit; reports generated by other
9575 means need not be addressed.

9576 ISO/IEC 15408-3 ALC_FLR.3.6C: *The flaw remediation procedures shall include a procedure*
9577 *requiring timely response and the automatic distribution of security flaw reports and the associated*
9578 *corrections to registered users who might be affected by the security flaw.*

9579 **13.6.3.3.7 Work unit ALC_FLR.3-7**

9580 The evaluator ***shall examine*** the flaw remediation procedures to determine that the application of
9581 these procedures would result in a timely means of providing the registered TOE users who might
9582 be affected with reports about, and associated corrections to, each security flaw.

9583 The issue of timeliness applies to the issuance of both security flaw reports and the associated
9584 corrections. However, these need not be issued at the same time. It is recognised that flaw reports
9585 should be generated and issued as soon as an interim solution is found, even if that solution is as
9586 drastic as turn off the TOE. Likewise, when a more permanent (and less drastic) solution is found, it
9587 should be issued without undue delay.

9588 It is unnecessary to restrict the recipients of the reports and associated corrections to only those
9589 TOE users who might be affected by the security flaw; it is permissible that all TOE users be given
9590 such reports and corrections for all security flaws, provided such is done in a timely manner.

9591 **13.6.3.3.8 Work unit ALC_FLR.3-8**

9592 The evaluator ***shall examine*** the flaw remediation procedures to determine that the application of
9593 these procedures would result in automatic distribution of the reports and associated corrections
9594 to the registered TOE users who might be affected.

9595 *Automatic distribution* does not mean that human interaction with the distribution method is not
9596 permitted. In fact, the distribution method could consist entirely of manual procedures, perhaps
9597 through a closely monitored procedure with prescribed escalation upon the lack of issue of reports
9598 or corrections.

9599 It is unnecessary to restrict the recipients of the reports and associated corrections to only those
9600 TOE users who might be affected by the security flaw; it is permissible that all TOE users be given
9601 such reports and corrections for all security flaws, provided such is done automatically.

9602 ISO/IEC 15408-3 ALC_FLR.3.7C: *The procedures for processing reported security flaws shall ensure*
9603 *that any reported flaws are remediated and the remediation procedures issued to TOE users.*

9604 **13.6.3.3.9 Work unit ALC_FLR.3-9**

9605 The evaluator **shall examine** the flaw remediation procedures to determine that the application of
9606 these procedures would help to ensure that every reported flaw is corrected.

9607 The flaw remediation procedures cover not only those security flaws discovered and reported by
9608 developer personnel, but also those reported by TOE users. The procedures are sufficiently
9609 detailed so that they describe how it is ensured that each reported security flaw is remediated. The
9610 procedures contain reasonable steps that show progress leading to the eventual, inevitable
9611 resolution.

9612 The procedures describe the process that is taken from the point at which the suspected security
9613 flaw is determined to be a security flaw to the point at which it is resolved.

9614 **13.6.3.3.10 Work unit ALC_FLR.3-10**

9615 The evaluator **shall examine** the flaw remediation procedures to determine that the application of
9616 these procedures would help to ensure that the TOE users are issued remediation procedures for
9617 each security flaw.

9618 The procedures describe the process that is taken from the point at which a security flaw is
9619 resolved to the point at which the remediation procedures are provided. The procedures for
9620 delivering remediation procedures should be consistent with the security objectives; they need not
9621 necessarily be identical to the procedures used for delivering the TOE, as documented to meet
9622 Delivery (ALC_DEL), if included in the assurance requirements. For example, if the hardware
9623 portion of a TOE were originally delivered by bonded courier, updates to hardware resulting from
9624 flaw remediation would likewise be expected to be distributed by bonded courier. Updates
9625 unrelated to flaw remediation would follow the procedures set forth in the documentation meeting
9626 the Delivery (ALC_DEL) requirements.

9627 ISO/IEC 15408-3 ALC_FLR.3.8C: *The procedures for processing reported security flaws shall provide*
9628 *safeguards that any corrections to these security flaws do not introduce any new flaws.*

9629 **13.6.3.3.11 Work unit ALC_FLR.3-11**

9630 The evaluator **shall examine** the flaw remediation procedures to determine that the application of
9631 these procedures would result in safeguards that the potential correction contains no adverse
9632 effects.

9633 Through analysis, testing, or a combination of the two, the developer may reduce the likelihood
9634 that adverse effects will be introduced when a security flaw is corrected. The evaluator assesses
9635 whether the procedures provide detail in how the necessary mix of analysis and testing actions is
9636 to be determined for a given correction.

9637 The evaluator also determines that, for instances where the source of the security flaw is a
9638 documentation problem, the procedures include the means of safeguarding against the
9639 introduction of contradictions with other documentation.

9640 ISO/IEC 15408-3 ALC_FLR.3.9C: *The flaw remediation guidance shall describe a means by which TOE*
9641 *users report to the developer any suspected security flaws in the TOE.*

9642 **13.6.3.3.12 Work unit ALC_FLR.3-12**

9643 The evaluator *shall examine* the flaw remediation guidance to determine that the application of
 9644 these procedures would result in a means for the TOE user to provide reports of suspected security
 9645 flaws or requests for corrections to such flaws.

9646 The guidance ensures that TOE users have a means by which they can communicate with the TOE
 9647 developer. By having a means of contact with the developer, the user can report security flaws,
 9648 enquire about the status of security flaws, or request corrections to flaws.

9649 ISO/IEC 15408-3 ALC_FLR.3.10C: *The flaw remediation guidance shall describe a means by which*
 9650 *TOE users may register with the developer, to be eligible to receive security flaw reports and*
 9651 *corrections.*

9652 **13.6.3.3.13 Work unit ALC_FLR.3-13**

9653 The evaluator *shall examine* the flaw remediation guidance to determine that it describes a means
 9654 of enabling the TOE users to register with the developer.

9655 *Enabling the TOE users to register with the developer* simply means having a way for each TOE user
 9656 to provide the developer with a point of contact; this point of contact is to be used to provide the
 9657 TOE user with information related to security flaws that might affect that TOE user, along with any
 9658 corrections to the security flaw. Registering the TOE user may be accomplished as part of the
 9659 standard procedures that TOE users undergo to identify themselves to the developer, for the
 9660 purposes of registering a software licence, or for obtaining update and other useful information.

9661 There need not be one registered TOE user per installation of the TOE; it would be sufficient if
 9662 there were one registered TOE user for an organisation. For example, a corporate TOE user might
 9663 have a centralised acquisition office for all of its sites. In this case, the acquisition office would be a
 9664 sufficient point of contact for all of that TOE user's sites, so that all of the TOE user's installations of
 9665 the TOE have a registered point of contact.

9666 In either case, it must be possible to associate each TOE that is delivered with an organisation in
 9667 order to ensure that there is a registered user for each TOE. For organisations that have many
 9668 different addresses, this assures that there will be no user who is erroneously presumed to be
 9669 covered by a registered TOE user.

9670 It should be noted that TOE users need not register; they must only be provided with a means of
 9671 doing so. However, users who choose to register must be directly sent the information (or a
 9672 notification of its availability).

9673 ISO/IEC 15408-3 ALC_FLR.3.11C: *The flaw remediation guidance shall identify the specific points of*
 9674 *contact for all reports and enquiries about security issues involving the TOE.*

9675 **13.6.3.3.14 Work unit ALC_FLR.3-14**

9676 The evaluator *shall examine* the flaw remediation guidance to determine that it identifies specific
 9677 points of contact for user reports and enquiries about security issues involving the TOE.

9678 The guidance includes a means whereby registered TOE users can interact with the developer to
 9679 report discovered security flaws in the TOE or to make enquiries regarding discovered security
 9680 flaws in the TOE.

9681 **13.7 Life-cycle definition (ALC_LCD)**

9682 **13.7.1 Evaluation of sub-activity (ALC_LCD.1)**

9683 **13.7.1.1 Objectives**

9684 The objective of this sub-activity is to determine whether the developer has used a documented
9685 model of the TOE life-cycle.

9686 **13.7.1.2 Input**

9687 The evaluation evidence for this sub-activity is:

- 9688 a) the ST;
- 9689 b) the life-cycle definition documentation.

9690 **13.7.1.3 Action ALC_LCD.1.1E**

9691 ISO/IEC 15408-3 ALC_LCD.1.1C: *The life-cycle definition documentation shall describe the model*
9692 *used to develop and maintain the TOE.*

9693 **13.7.1.3.1 Work unit ALC_LCD.1-1**

9694 The evaluator **shall examine** the documented description of the life-cycle model used to determine
9695 that it covers the development and maintenance process.

9696 The description of the life-cycle model should include:

- 9697 a) information on the life-cycle phases of the TOE and the boundaries between the
9698 subsequent phases;
- 9699 b) information on the procedures, tools and techniques used by the developer (e.g. for
9700 design, coding, testing, bug-fixing);
- 9701 c) overall management structure governing the application of the procedures (e.g. an
9702 identification and description of the individual responsibilities for each of the procedures
9703 required by the development and maintenance process covered by the life-cycle model);
- 9704 d) information on which parts of the TOE are delivered by subcontractors, if subcontractors
9705 are involved.

9706 Evaluation of sub-activity (ALC_LCD.1) does not require the model used to conform to any standard
9707 life-cycle model.

9708 ISO/IEC 15408-3 ALC_LCD.1.2C: *The life-cycle model shall provide for the necessary control over the*
9709 *development and maintenance of the TOE.*

9710 **13.7.1.3.2 Work unit ALC_LCD.1-2**

9711 The evaluator **shall examine** the life-cycle model to determine that use of the procedures, tools
9712 and techniques described by the life-cycle model will make the necessary positive contribution to
9713 the development and maintenance of the TOE.

9714 The information provided in the life-cycle model gives the evaluator assurance that the
9715 development and maintenance procedures adopted would minimise the likelihood of security
9716 flaws. For example, if the life-cycle model described the review process, but did not make provision
9717 for recording changes to components, then the evaluator may be less confident that errors will not

9718 be introduced into the TOE. The evaluator may gain further assurance by comparing the
 9719 description of the model against an understanding of the development process gleaned from
 9720 performing other evaluator actions relating to the TOE development (e.g. those covered under the
 9721 CM capabilities (ALC_CMC)). Identified deficiencies in the life-cycle model will be of concern if they
 9722 might reasonably be expected to give rise to the introduction of flaws into the TOE, either
 9723 accidentally or deliberately.

9724 ISO/IEC 15408 does not mandate any particular development approach, and each should be judged
 9725 on merit. For example, spiral, rapid-prototyping and waterfall approaches to design can all be used
 9726 to produce a quality TOE if applied in a controlled environment.

9727 **13.7.2 Evaluation of sub-activity (ALC_LCD.2)**

9728 **13.7.2.1 Objectives**

9729 The objective of this sub-activity is to determine whether the developer has used a documented
 9730 and measurable model of the TOE life-cycle.

9731 **13.7.2.2 Input**

9732 The evaluation evidence for this sub-activity is:

- 9733 a) the ST;
- 9734 b) the life-cycle definition documentation;
- 9735 c) information about the standard used;
- 9736 d) the life-cycle output documentation.

9737 **13.7.2.3 Action ALC_LCD.2.1E**

9738 ISO/IEC 15408-3 ALC_LCD.2.1C: *The life-cycle definition documentation shall describe the model*
 9739 *used to develop and maintain the TOE, including the details of its arithmetic parameters and/or*
 9740 *metrics used to measure the quality of the TOE and/or its development.*

9741 **13.7.2.3.1 Work unit ALC_LCD.2-1**

9742 The evaluator ***shall examine*** the documented description of the life-cycle model used to determine
 9743 that it covers the development and maintenance process, including the details of its arithmetic
 9744 parameters and/or metrics used to measure the TOE development.

9745 The description of the life-cycle model includes:

- 9746 a) information on the life-cycle phases of the TOE and the boundaries between the
 9747 subsequent phases;
- 9748 b) information on the procedures, tools and techniques used by the developer (e.g. for
 9749 design, coding, testing, bug-fixing);
- 9750 c) overall management structure governing the application of the procedures (e.g. an
 9751 identification and description of the individual responsibilities for each of the procedures
 9752 required by the development and maintenance process covered by the life-cycle model);
- 9753 d) information on which parts of the TOE are delivered by subcontractors, if subcontractors
 9754 are involved;

9755 e) information on the parameters/metrics that are used to measure the TOE development.
9756 Metrics standards typically include guides for measuring and producing reliable products
9757 and cover the aspects reliability, quality, performance, complexity and cost. For the
9758 evaluation all those metrics are of relevance, which are used to increase quality by
9759 decreasing the probability of faults and thereby in turn increase assurance in the security
9760 of the TOE.

9761 ISO/IEC 15408-3 ALC_LCD.2.2C: *The life-cycle model shall provide for the necessary control over the*
9762 *development and maintenance of the TOE.*

9763 **13.7.2.3.2 Work unit ALC_LCD.2-2**

9764 The evaluator *shall examine* the life-cycle model to determine that use of the procedures, tools
9765 and techniques described by the life-cycle model will make the necessary positive contribution to
9766 the development and maintenance of the TOE.

9767 The information provided in the life-cycle model gives the evaluator assurance that the
9768 development and maintenance procedures adopted would minimise the likelihood of security
9769 flaws. For example, if the life-cycle model described the review process, but did not make provision
9770 for recording changes to components, then the evaluator may be less confident that errors will not
9771 be introduced into the TOE. The evaluator may gain further assurance by comparing the
9772 description of the model against an understanding of the development process gleaned from
9773 performing other evaluator actions relating to the TOE development (e.g. those covered under the
9774 CM capabilities (ALC_CMC)). Identified deficiencies in the life-cycle model will be of concern if they
9775 might reasonably be expected to give rise to the introduction of flaws into the TOE, either
9776 accidentally or deliberately.

9777 ISO/IEC 15408 does not mandate any particular development approach, and each should be judged
9778 on merit. For example, spiral, rapid-prototyping and waterfall approaches to design can all be used
9779 to produce a quality TOE if applied in a controlled environment.

9780 For the metrics/measurements used in the life-cycle model, evidence has to be provided that
9781 shows how those metrics/measurements usefully contribute to the minimisation of the likelihood
9782 of flaws. This can be viewed as the overall goal for measurement in an ALC context. As a
9783 consequence the metrics/measurements have to be selected based on their capability to achieve
9784 that overall goal or contribute to that. In the first place a metric/measure is suitable with respect to
9785 ALC if a correlation between the metric/measure and the number of flaws can be stated with a
9786 certain degree of reliability. But also a metric/measure useful for management purposes as for
9787 planning and monitoring the TOE development are helpful since badly managed projects are
9788 endangered to produce bad quality and to introduce flaws.

9789 It may be possible to use metrics for quality improvement, for which this use is not obvious. For
9790 example a metric to estimate the expected cost of a product development may help quality, if the
9791 developer can show that this is used to provide an adequate budget for development projects and
9792 that this helps to avoid quality problems arising from resource shortages.

9793 It is not required that every single step in the life cycle of the TOE is measurable. However the
9794 evaluator should see from the description of the measures and procedures that the metrics are
9795 appropriate to control the overall quality of the TOE and to minimise possible security flaws by this.

9796 ISO/IEC 15408-3 ALC_LCD.2.3C: *The life-cycle output documentation shall provide the results of the*
9797 *measurements of the TOE development using the measurable life-cycle model.*

9798 **13.7.2.3.3 Work unit ALC_LCD.2-3**

9799 The evaluator *shall examine* the life-cycle output documentation to determine that it provides the
9800 results of the measurements of the TOE development using the measurable life-cycle model.

9801 The results of the measurements and the life-cycle progress of the TOE should be in accordance
9802 with the life-cycle model.

9803 The output documentation not only includes numeric values of the metrics but also documents
9804 actions taken as a result of the measurements and in accordance with the model. For example there
9805 may be a requirement that a certain design phase needs to be repeated, if some error rates
9806 measured during testing are outside of a defined threshold. In this case the documentation should
9807 show that such action was taken, if indeed the thresholds were not met.

9808 If the evaluation is conducted in parallel with the development of the TOE it may be possible that
9809 quality measurements have not been used in the past. In this case the evaluator should use the
9810 documentation of the planned procedures in order to gain confidence that corrective actions are
9811 defined if results of quality measurements deviate from some threshold.

9812 **13.8 TOE Development Artifacts (ALC_TDA)**

9813 **13.8.1 Evaluation of sub-activity (ALC_TDA.1)**

9814 **Editors' Notes**

9815 **Suggestions for text would be welcomed in response to CD2 review**

9816

9817 **13.9 Tools and techniques (ALC_TAT)**

9818 **13.9.1 Evaluation of sub-activity (ALC_TAT.1)**

9819 **13.9.1.1 Objectives**

9820 The objective of this sub-activity is to determine whether the developer has used well-defined
9821 development tools (e.g. programming languages or computer-aided design (CAD) systems) that
9822 yield consistent and predictable results.

9823 **13.9.1.2 Input**

9824 The evaluation evidence for this sub-activity is:

- 9825 a) the development tool documentation;
- 9826 b) the subset of the implementation representation.

9827 **13.9.1.3 Application notes**

9828 This work may be performed in parallel with the evaluation activities under Implementation
9829 representation (ADV_IMP), specifically with regard to determining the use of features in the tools
9830 that will affect the object code (e.g. compilation options).

9831 **13.9.1.4 Action ALC_TAT.1.1E**

9832 ISO/IEC 15408-3 ALC_TAT.1.1C: *Each development tool used for implementation shall be well-*
9833 *defined.*

9834 **13.9.1.4.1 Work unit ALC_TAT.1-1**

9835 The evaluator ***shall examine*** the development tool documentation provided to determine that
9836 each development tools is well-defined.

9837 For example, a well-defined language, compiler or CAD system may be considered to be one that
9838 conforms to a recognised standard, such as the ISO standards. A well-defined language is one that
9839 has a clear and complete description of its syntax, and a detailed description of the semantics of
9840 each construct.

9841 ISO/IEC 15408-3 ALC_TAT.1.2C: *The documentation of each development tool shall unambiguously*
9842 *define the meaning of all statements as well as all conventions and directives used in the*
9843 *implementation.*

9844 **13.9.1.4.2 Work unit ALC_TAT.1-2**

9845 The evaluator *shall examine* the documentation of each development tool to determine that it
9846 unambiguously defines the meaning of all statements as well as all conventions and directives used
9847 in the implementation.

9848 The development tool documentation (e.g. programming language specifications and user
9849 manuals) should cover all statements used in the implementation representation of the TOE, and
9850 for each such statement should provide a clear and unambiguous definition of the purpose and
9851 effect of that statement. This work may be performed in parallel with the evaluator's examination
9852 of the implementation representation performed during the ADV_IMP sub-activity. The key test the
9853 evaluator should apply is whether or not the documentation is sufficiently clear for the evaluator to
9854 be able to understand the implementation representation. The documentation should not assume
9855 (for example) that the reader is an expert in the programming language used.

9856 Reference to the use of a documented standard is an acceptable approach to meet this requirement,
9857 provided that the standard is available to the evaluator. Any differences from the standard should
9858 be documented.

9859 The critical test is whether the evaluator can understand the TOE source code when performing
9860 source code analysis covered in the ADV_IMP sub-activity. However, the following checklist can
9861 additionally be used in searching for problem areas:

9862 a) In the language definition, phrases such as “the effect of this construct is undefined” and
9863 terms such as “implementation dependent” or “erroneous” may indicate ill-defined areas.

9864 b) Aliasing (allowing the same piece of memory to be referenced in different ways) is a
9865 common source of ambiguity problems.

9866 c) Exception handling (e.g. what happens after memory exhaustion or stack overflow) is
9867 often poorly defined.

9868 Most languages in common use, however well designed, will have some problematic constructs. If
9869 the implementation language is mostly well defined, but some problematic constructs exist, then
9870 an inconclusive verdict should be assigned, pending examination of the source code.

9871 The evaluator should verify, during the examination of source code, that any use of the problematic
9872 constructs does not introduce vulnerabilities. The evaluator should also ensure that constructs
9873 precluded by the documented standard are not used.

9874 The development tool documentation should define all conventions and directives used in the
9875 implementation.

9876 ISO/IEC 15408-3 ALC_TAT.1.3C: *The documentation of each development tool shall unambiguously*
9877 *define the meaning of all implementation-dependent options.*

9878 **13.9.1.4.3 Work unit ALC_TAT.1-3**

9879 The evaluator *shall examine* the development tool documentation to determine that it
9880 unambiguously defines the meaning of all implementation-dependent options.

9881 The documentation of software development tools should include definitions of implementation-
9882 dependent options that may affect the meaning of the executable code, and those that are different
9883 from the standard language as documented. Where source code is provided to the evaluator,
9884 information should also be provided on compilation and linking options used.

9885 The documentation for hardware design and development tools should describe the use of all
9886 options that affect the output from the tools (e.g. detailed hardware specifications, or actual
9887 hardware).

9888 **13.9.2 Evaluation of sub-activity (ALC_TAT.2)**9889 **13.9.2.1 Objectives**

9890 The objective of this sub-activity is to determine whether the developer has used well-defined
9891 development tools (e.g. programming languages or computer-aided design (CAD) systems) that
9892 yield consistent and predictable results, and whether implementation standards have been applied.

9893 **13.9.2.2 Input**

9894 The evaluation evidence for this sub-activity is:

- 9895 a) the development tool documentation;
- 9896 b) the implementation standards description;
- 9897 c) the provided implementation representation of the TSF.

9898 **13.9.2.3 Application notes**

9899 This work may be performed in parallel with the evaluation activities under ADV_IMP, specifically
9900 with regard to determining the use of features in the tools that will affect the object code (e.g.
9901 compilation options).

9902 **13.9.2.4 Action ALC_TAT.2.1E**

9903 ISO/IEC 15408-3 ALC_TAT.2.1C: *Each development tool used for implementation shall be well-*
9904 *defined.*

9905 **13.9.2.4.1 Work unit ALC_TAT.2-1**

9906 The evaluator *shall examine* the development tool documentation provided to determine that
9907 each development tool is well-defined.

9908 For example, a well-defined language, compiler or CAD system may be considered to be one that
9909 conforms to a recognised standard, such as the ISO standards. A well-defined language is one that
9910 has a clear and complete description of its syntax, and a detailed description of the semantics of
9911 each construct.

9912 ISO/IEC 15408-3 ALC_TAT.2.2C: *The documentation of each development tool shall unambiguously*
9913 *define the meaning of all statements as well as all conventions and directives used in the*
9914 *implementation.*

9915 **13.9.2.4.2 Work unit ALC_TAT.2-2**

9916 The evaluator *shall examine* the documentation of each development tool to determine that it
9917 unambiguously defines the meaning of all statements as well as all conventions and directives used
9918 in the implementation.

9919 The development tool documentation (e.g. programming language specifications and user
9920 manuals) should cover all statements used in the implementation representation of the TOE, and
9921 for each such statement should provide a clear and unambiguous definition of the purpose and
9922 effect of that statement. This work may be performed in parallel with the evaluator's examination
9923 of the implementation representation performed during the ADV_IMP sub-activity. The key test the
9924 evaluator should apply is whether or not the documentation is sufficiently clear for the evaluator to
9925 be able to understand the implementation representation. The documentation should not assume
9926 (for example) that the reader is an expert in the programming language used.

9927 Reference to the use of a documented standard is an acceptable approach to meet this requirement,
9928 provided that the standard is available to the evaluator. Any differences from the standard should
9929 be documented.

9930 The critical test is whether the evaluator can understand the TOE source code when performing
9931 source code analysis covered in the ADV_IMP sub-activity. However, the following checklist can
9932 additionally be used in searching for problem areas:

9933 a) In the language definition, phrases such as “the effect of this construct is undefined” and
9934 terms such as “implementation dependent” or “erroneous” may indicate ill-defined areas.

9935 b) Aliasing (allowing the same piece of memory to be referenced in different ways) is a
9936 common source of ambiguity problems.

9937 c) Exception handling (e.g. what happens after memory exhaustion or stack overflow) is
9938 often poorly defined.

9939 Most languages in common use, however well designed, will have some problematic constructs. If
9940 the implementation language is mostly well defined, but some problematic constructs exist, then
9941 an inconclusive verdict should be assigned, pending examination of the source code.

9942 The evaluator should verify, during the examination of source code, that any use of the problematic
9943 constructs does not introduce vulnerabilities. The evaluator should also ensure that constructs
9944 precluded by the documented standard are not used.

9945 The development tool documentation should define all conventions and directives used in the
9946 implementation.

9947 ISO/IEC 15408-3 ALC_TAT.2.3C: *The documentation of each development tool shall unambiguously*
9948 *define the meaning of all implementation-dependent options.*

9949 **13.9.2.4.3 Work unit ALC_TAT.2-3**

9950 The evaluator *shall examine* the development tool documentation to determine that it
9951 unambiguously defines the meaning of all implementation-dependent options.

9952 The documentation of software development tools should include definitions of implementation-
9953 dependent options that may affect the meaning of the executable code, and those that are different
9954 from the standard language as documented. Where source code is provided to the evaluator,
9955 information should also be provided on compilation and linking options used.

9956 The documentation for hardware design and development tools should describe the use of all
9957 options that affect the output from the tools (e.g. detailed hardware specifications, or actual
9958 hardware).

9959 **13.9.2.5 Action ALC_TAT.2.2E**

9960 **13.9.2.5.1 Work unit ALC_TAT.2-4**

9961 The evaluator *shall examine* aspects of the implementation process to determine that documented
9962 implementation standards have been applied.

9963 This work unit requires the evaluator to analyse the provided implementation representation of
9964 the TOE to determine whether the documented implementation standards have been applied.

9965 The evaluator should verify that constructs excluded by the documented standard are not used.

9966 Additionally, the evaluator should verify the developer's procedures which ensure the application
9967 of the defined standards within the design and implementation process of the TOE. Therefore,
9968 documentary evidence should be supplemented by visiting the development environment. A visit
9969 to the development environment will allow the evaluator to:

9970 a) observe the application of defined standards;

9971 b) examine documentary evidence of application of procedures describing the use of defined
9972 standards;

9973 c) interview development staff to check awareness of the application of defined standards
9974 and procedures.

9975 A development site visit is a useful means of gaining confidence in the procedures being used. Any
9976 decision not to make such a visit should be determined in consultation with the evaluation
9977 authority.

9978 The evaluator compares the provided implementation representation with the description of the
9979 applied implementation standards and verifies their use.

9980 At this level it is not required that the complete provided implementation representation of the
9981 TSF is based on implementation standards, but only those parts that are developed by the TOE
9982 developer himself. The evaluator may consult the configuration list required by the CM scope
9983 (ALC_CMS) to get the information which parts are developed by the TOE developer, and which by
9984 third party developers.

9985 If the referenced implementation standards are not applied for at least parts of the provided
9986 implementation representation, the evaluator action related to this work unit is assigned a fail
9987 verdict.

9988 Note that parts of the TOE which are not TSF relevant do not need to be examined.

9989 This work unit may be performed in conjunction with the evaluation activities under ADV_IMP.

9990 **13.9.3 Evaluation of sub-activity (ALC_TAT.3)**

9991 **13.9.3.1 Objectives**

9992 The objective of this sub-activity is to determine whether the developer and their subcontractors
9993 have used well-defined development tools (e.g. programming languages or computer-aided design
9994 (CAD) systems) that yield consistent and predictable results, and whether implementation
9995 standards have been applied.

9996 **13.9.3.2 Input**

9997 The evaluation evidence for this sub-activity is:

9998 a) the development tool documentation;

9999 b) the implementation standards description;

10000 c) the provided implementation representation of the TSF.

10001 **13.9.3.3 Application notes**

10002 This work may be performed in parallel with the evaluation activities under ADV_IMP, specifically
10003 with regard to determining the use of features in the tools that will affect the object code (e.g.
10004 compilation options).

10005 **13.9.3.4 Action ALC_TAT.3.1E**

10006 ISO/IEC 15408-3 ALC_TAT.3.1C: *Each development tool used for implementation shall be well-*
10007 *defined.*

10008 **13.9.3.4.1 Work unit ALC_TAT.3-1**

10009 The evaluator *shall examine* the development tool documentation provided to determine that
10010 each development tool is well-defined.

10011 For example, a well-defined language, compiler or CAD system may be considered to be one that
10012 conforms to a recognised standard, such as the ISO standards. A well-defined language is one that
10013 has a clear and complete description of its syntax, and a detailed description of the semantics of
10014 each construct.

10015 At this level, the documentation of development tools used by third party contributors to the TOE
10016 has to be included in the evaluator's examination.

10017 ISO/IEC 15408-3 ALC_TAT.3.2C: *The documentation of each development tool shall unambiguously*
10018 *define the meaning of all statements as well as all conventions and directives used in the*
10019 *implementation.*

10020 **13.9.3.4.2 Work unit ALC_TAT.3-2**

10021 The evaluator *shall examine* the documentation of each development tool to determine that it
10022 unambiguously defines the meaning of all statements as well as all conventions and directives used
10023 in the implementation.

10024 The development tool documentation (e.g. programming language specifications and user
10025 manuals) should cover all statements used in the implementation representation of the TOE, and
10026 for each such statement should provide a clear and unambiguous definition of the purpose and
10027 effect of that statement. This work may be performed in parallel with the evaluator's examination
10028 of the implementation representation performed during the ADV_IMP sub-activity. The key test the
10029 evaluator should apply is whether or not the documentation is sufficiently clear for the evaluator to
10030 be able to understand the implementation representation. The documentation should not assume
10031 (for example) that the reader is an expert in the programming language used.

10032 Reference to the use of a documented standard is an acceptable approach to meet this requirement,
10033 provided that the standard is available to the evaluator. Any differences from the standard should
10034 be documented.

- 10035 The critical test is whether the evaluator can understand the TOE source code when performing
10036 source code analysis covered in the ADV_IMP sub-activity. However, the following checklist can
10037 additionally be used in searching for problem areas:
- 10038 a) In the language definition, phrases such as “the effect of this construct is undefined” and
10039 terms such as “implementation dependent” or “erroneous” may indicate ill-defined areas.
- 10040 b) Aliasing (allowing the same piece of memory to be referenced in different ways) is a
10041 common source of ambiguity problems.
- 10042 c) Exception handling (e.g. what happens after memory exhaustion or stack overflow) is
10043 often poorly defined.
- 10044 Most languages in common use, however well designed, will have some problematic constructs. If
10045 the implementation language is mostly well defined, but some problematic constructs exist, then
10046 an inconclusive verdict should be assigned, pending examination of the source code.
- 10047 The evaluator should verify, during the examination of source code, that any use of the problematic
10048 constructs does not introduce vulnerabilities. The evaluator should also ensure that constructs
10049 precluded by the documented standard are not used.
- 10050 The development tool documentation should define all conventions and directives used in the
10051 implementation.
- 10052 At this level, the documentation of development tools used by third party contributors to the TOE
10053 has to be included in the evaluator's examination.
- 10054 ISO/IEC 15408-3 ALC_TAT.3.3C: *The documentation of each development tool shall unambiguously*
10055 *define the meaning of all implementation-dependent options.*
- 10056 **13.9.3.4.3 Work unit ALC_TAT.3-3**
- 10057 The evaluator ***shall examine*** the development tool documentation to determine that it
10058 unambiguously defines the meaning of all implementation-dependent options.
- 10059 The documentation of software development tools should include definitions of implementation-
10060 dependent options that may affect the meaning of the executable code, and those that are different
10061 from the standard language as documented. Where source code is provided to the evaluator,
10062 information should also be provided on compilation and linking options used.
- 10063 The documentation for hardware design and development tools should describe the use of all
10064 options that affect the output from the tools (e.g. detailed hardware specifications, or actual
10065 hardware).
- 10066 At this level, the documentation of development tools used by third party contributors to the TOE
10067 has to be included in the evaluator's examination.
- 10068 **13.9.3.5 Action ALC_TAT.3.2E**
- 10069 **13.9.3.5.1 Work unit ALC_TAT.3-4**
- 10070 The evaluator ***shall examine*** aspects of the implementation process to determine that documented
10071 implementation standards have been applied.
- 10072 This work unit requires the evaluator to analyse the provided implementation representation of
10073 the TOE to determine whether the documented implementation standards have been applied.
- 10074 The evaluator should verify that constructs excluded by the documented standard are not used.

10075 Additionally, the evaluator should verify the developer's procedures which ensure the application
 10076 of the defined standards within the design and implementation process of the TOE. Therefore,
 10077 documentary evidence should be supplemented by visiting the development environment. A visit
 10078 to the development environment will allow the evaluator to:

10079 a) observe the application of defined standards;

10080 b) examine documentary evidence of application of procedures describing the use of defined
 10081 standards;

10082 c) interview development staff to check awareness of the application of defined standards
 10083 and procedures.

10084 A development site visit is a useful means of gaining confidence in the procedures being used. Any
 10085 decision not to make such a visit should be determined in consultation with the evaluation
 10086 authority.

10087 The evaluator compares the provided implementation representation with the description of the
 10088 applied implementation standards and verifies their use.

10089 At this level it is required that the complete provided implementation representation of the TSF is
 10090 based on implementation standards, including third party contributions. This may require the
 10091 evaluator to visit the sites of contributors. The evaluator may consult the configuration list
 10092 required by the CM scope (ALC_CMS) to see who has developed which part of the TOE.

10093 Note that parts of the TOE which are not TSF relevant do not need to be examined.

10094 This work unit may be performed in conjunction with the evaluation activities under ADV_IMP.

10095 **13.10 Integration of composition parts and consistency check of delivery**
 10096 **procedures (ALC_COMP)**

10097 The composite-specific work units defined here are intended to be integrated as refinements to the
 10098 evaluation activities of the ALC class listed in the following table. The other activities of ALC class
 10099 do not require composite-specific work units.

CC assurance family	Evaluation activity	Evaluation work unit	Composite-specific work unit
ALC_CMS	ALC_CMS.1.2C	ALC_CMS.1-2	ALC_COMP.1-1
AGD_PRE	AGD_PRE.1.1C	AGD_PRE.1-1	ALC_COMP.1-2
ALC_CMC	ALC_CMC.4.8C	ALC_CMC.4-10	ALC_CMC.4-10

10100 NB: If the level of the assurance requirement chosen is higher than those identified in this table, the
 10101 composite-specific work unit is also applicable.

10102 **13.10.1 Evaluation of sub-activity (ALC_COMP.1)**

10103 **13.10.1.1 Objectives**

10104 The aims of this activity are to determine whether – the correct version of the application is
 10105 installed onto/into the correct version of the underlying platform, and

10106 – the preparative guidance procedures of Platform and Application Developers are compatible with
 10107 the acceptance procedure of the Composite Product Integrator.

10108 **13.10.1.2 Action ALC_COMP.1.1E**

10109 The evaluator *shall examine* the evidence that the evaluated version of the application has been
10110 installed onto / embedded into the correct, certified version of the underlying platform.

10111 The AGD_PRE documentation of the platform provided by the platform developer contains
10112 requirements for the secure acceptance of the platform and security measures to which the
10113 application developer or product composite integrator has to adhere. The application developer
10114 has to provide evidence that (if applicable), these requirements are followed up and the required
10115 security measures are implemented.

10116 The special composite evaluator activity is to check the evidence of the version correctness for
10117 both parts of the composite product and that the secure acceptance and installation of the platform
10118 has been performed.

10119 For the underlying platform, the evaluator shall determine that the actual identification of the
10120 platform is commensurate with the respective data in the platform certificate as part of following
10121 up on the procedures as specified in the AGD_PRE of the platform.

10122 For the application, the relevant task is trivial due to the fact that the Composite Product Evaluator
10123 has to perform this task in the context of the assurance family ALC_CMS.

10124 Components identification evidence can be supplied in two different ways: technical and
10125 organisational. A technical evidence of version correctness is being generated by the composite
10126 product itself: the platform and the application return – in each case – strings containing
10127 unambiguous version numbers as answers to the respective commands. E.g. it can be the return
10128 string of a command or the hard copy of the Windows Information (like ‘About’); in case of smart
10129 cards it can be an appropriate ATR.

10130 A technical evidence of version correctness for hardware can also be supplied, if applicable, by
10131 reading off the unambiguous inscription on its surface. Note that there are no physical indication
10132 existing on most smart cards microcontrollers.

10133 Technical evidence is recommended to be provided.

10134 An organisational evidence of version correctness is being generated by the Composite Product
10135 Integrator on the basis of his configuration lists containing unambiguous version information of
10136 the platform and the application having been composed into the final composite product.

10137 For example, in case of smart cards it can be an acknowledgement statement (e.g. configuration
10138 list) of the underlying platform manufacturer to the application software manufacturer containing
10139 the evidence for the versions of the platform, any embedded software and its pre-personalisation
10140 parameters¹⁸. ⁵⁵ Organisational evidence is always possible and, hence, shall be provided. ⁵⁶ The
10141 result of this work unit shall be integrated to the result of ALC_CMS1.1C/ ALC_CMS.1-2 (or the
10142 equivalent higher components if a higher assurance level is selected).

10143 ALC_COMP.1.2C

10144 **Editors' Notes**

10145 **Suggestions for text improvements would be welcomed in response to CD2 review**

10146 **14 Class ATE: Tests**10147 **14.1 Introduction**

10148 The goal of this activity is to determine whether the TOE behaves as described in the ST and as
10149 specified in the evaluation evidence (described in the ADV class). This determination is achieved

10150 through some combination of the developer's own functional testing of the TSF (Functional tests
10151 (ATE_FUN)) and independent testing the TSF by the evaluator (Independent testing (ATE_IND)). At
10152 the lowest level of assurance, there is no requirement for developer involvement, so the only
10153 testing is conducted by the evaluator, using the limited available information about the TOE.
10154 Additional assurance is gained as the developer becomes increasingly involved both in testing and
10155 in providing additional information about the TOE, and as the evaluator increases the independent
10156 testing activities.

10157 **14.2 Application notes**

10158 Testing of the TSF is conducted by the evaluator and, in most cases, by the developer. The
10159 evaluator's testing efforts consist not only of creating and running original tests, but also of
10160 assessing the adequacy of the developer's tests and re-running a subset of them.

10161 The evaluator analyses the developer's tests to determine the extent to which they are sufficient to
10162 demonstrate that TSFI (see Functional specification (ADV_FSP)) perform as specified, and to
10163 understand the developer's approach to testing. Similarly, the evaluator analyses the developer's
10164 tests to determine the extent to which they are sufficient to demonstrate the internal behaviour
10165 and properties of the TSF.

10166 The evaluator also executes a subset of the developer's tests as documented to gain confidence in
10167 the developer's test results: the evaluator will use the results of this analysis as an input to
10168 independently testing a subset of the TSF. With respect to this subset, the evaluator takes a testing
10169 approach that is different from that of the developer, particularly if the developer's tests have
10170 shortcomings.

10171 To determine the adequacy of developer's test documentation or to create new tests, the evaluator
10172 needs to understand the desired expected behaviour of the TSF, both internally and as seen at the
10173 TSFI, in the context of the SFRs it is to satisfy. The evaluator may choose to divide the TSF and TSFI
10174 into subsets according to functional areas of the ST (audit subsystem, audit-related TSFI,
10175 authentication module, authentication-related TSFI, etc.) if they were not already divided in the ST,
10176 and focus on one subset of the TSF and TSFI at a time, examining the ST requirement and the
10177 relevant parts of the development and guidance documentation to gain an understanding of the
10178 way the TOE is expected to behave. This reliance upon the development documentation
10179 underscores the need for the dependencies on ADV by Coverage (ATE_COV) and Depth (ATE_DPT).

10180 ISO/IEC 15408 has separated coverage and depth from functional tests to increase the flexibility
10181 when applying the components of the families. However, the requirements of the families are
10182 intended to be applied together to confirm that the TSF operates according to its specification. This
10183 tight coupling of families has led to some duplication of evaluator work units across sub-activities.
10184 These application notes are used to minimise duplication of text between sub-activities.

10185 **14.2.1 Understanding the expected behaviour of the TOE**

10186 Before the adequacy of test documentation can be accurately evaluated, or before new tests can be
10187 created, the evaluator has to understand the desired expected behaviour of a security function in
10188 the context of the requirements it is to satisfy.

10189 As mentioned earlier, the evaluator may choose to subset the TSF and TSFI according to SFRs
10190 (audit, authentication, etc.) in the ST and focus on one subset at a time. The evaluator examines
10191 each ST requirement and the relevant parts of the functional specification and guidance
10192 documentation to gain an understanding of the way the related TSFI is expected to behave.
10193 Similarly, the evaluator examines the relevant parts of the TOE design and security architecture
10194 documentation to gain an understanding of the way the related modules or subsystems of the TSF
10195 are expected to behave.

10196 With an understanding of the expected behaviour, the evaluator examines the test plan to gain an
10197 understanding of the testing approach. In most cases, the testing approach will entail a TSFI being

10198 stimulated and its responses observed. Externally-visible functionality can be tested directly;
 10199 however, in cases where functionality is not visible external to the TOE (for example, testing the
 10200 residual information protection functionality), other means will need to be employed.

10201 **14.2.2 Testing vs. alternate approaches to verify the expected behaviour of functionality**

10202 In cases where it is impractical or inadequate to test specific functionality (where it provides no
 10203 externally-visible TSFI), the test plan should identify the alternate approach to verify expected
 10204 behaviour. It is the evaluator's responsibility to determine the suitability of the alternate approach.
 10205 However, the following should be considered when assessing the suitability of alternate
 10206 approaches:

10207 a) an analysis of the implementation representation to determine that the required
 10208 behaviour should be exhibited by the TOE is an acceptable alternate approach. This could
 10209 mean a code inspection for a software TOE or perhaps a chip mask inspection for a
 10210 hardware TOE.

10211 b) it is acceptable to use evidence of developer integration or module testing, even if the
 10212 claimed assurance requirements do not include availability of lower level descriptions of
 10213 the TOE modules (e.g. Evaluation of sub-activity (ADV_TDS.3)) or implementation
 10214 (Implementation representation (ADV_IMP)). If evidence of developer integration or
 10215 module testing is used in verifying the expected behaviour of a security functionality,
 10216 care should be given to confirm that the testing evidence reflects the current
 10217 implementation of the TOE. If the subsystems or modules have been changed since
 10218 testing occurred, evidence that the changes were tracked and addressed by analysis or
 10219 further testing will usually be required.

10220 It should be emphasised that supplementing the testing effort with alternate approaches should
 10221 only be undertaken when both the developer and evaluator determine that there exists no other
 10222 practical means to test the expected behaviour.

10223 **14.2.3 Verifying the adequacy of tests**

10224 Test pre-requisites are necessary to establish the required initial conditions for the test. They may
 10225 be expressed in terms of parameters that must be set or in terms of test ordering in cases where
 10226 the completion of one test establishes the necessary pre-requisites for another test. The evaluator
 10227 must determine that the pre-requisites are complete and appropriate in that they will not bias the
 10228 observed test results towards the expected test results.

10229 The test steps and expected results specify the actions and parameters to be applied to the TSFI as
 10230 well as how the expected results should be verified and what they are. The evaluator must
 10231 determine that the test steps and expected results are consistent with the descriptions of the TSFI
 10232 in the functional specification. This means that each characteristic of the TSFI behaviour explicitly
 10233 described in the functional specification should have tests and expected results to verify that
 10234 behaviour.

10235 The overall aim of this testing activity is to determine that each subsystem, module, and TSFI has
 10236 been sufficiently tested against the behavioural claims in the functional specification, TOE design,
 10237 and architecture description. At the higher assurance levels, testing also includes bounds testing
 10238 and negative testing. The test procedures will provide insight as to how the TSFIs, modules, and
 10239 subsystems have been exercised by the developer during testing. The evaluator uses this
 10240 information when developing additional tests to independently test the TSF.

10242 **14.3 Coverage (ATE_COV)**

10243 **14.3.1 Evaluation of sub-activity (ATE_COV.1)**

10244 **14.3.1.1 Objectives**

10245 The objective of this sub-activity is to determine whether the developer has tested the TSFIs, and
10246 that the developer's test coverage evidence shows correspondence between the tests identified in
10247 the test documentation and the TSFIs described in the functional specification.

10248 **14.3.1.2 Input**

10249 The evaluation evidence for this sub-activity is:

- 10250 a) the ST;
- 10251 b) the functional specification;
- 10252 c) the test documentation;
- 10253 d) the test coverage evidence.

10254 **14.3.1.3 Application notes**

10255 The coverage analysis provided by the developer is required to show the correspondence between
10256 the tests provided as evaluation evidence and the functional specification. However, the coverage
10257 analysis need not demonstrate that all TSFI have been tested, or that all externally-visible
10258 interfaces to the TOE have been tested. Such shortcomings are considered by the evaluator during
10259 the independent testing (Evaluation of sub-activity (ATE_IND.2)) sub-activity.

10260 **14.3.1.4 Action ATE_COV.1.1E**

10261 ISO/IEC 15408-3 ATE_COV.1.1C: *The evidence of the test coverage shall show the correspondence*
10262 *between the tests in the test documentation and the TSFIs in the functional specification.*

10263 **14.3.1.4.1 Work unit ATE_COV.1-1**

10264 The evaluator **shall examine** the test coverage evidence to determine that the correspondence
10265 between the tests identified in the test documentation and the TSFIs described in the functional
10266 specification is accurate.

10267 Correspondence may take the form of a table or matrix. The coverage evidence required for this
10268 component will reveal the extent of coverage, rather than to show complete coverage. In cases
10269 where coverage is shown to be poor the evaluator should increase the level of independent testing
10270 to compensate.

10271 **14.3.2 Evaluation of sub-activity (ATE_COV.2)**

10272 **14.3.2.1 Objectives**

10273 The objective of this sub-activity is to determine whether the developer has tested all of the TSFIs,
10274 and that the developer's test coverage evidence shows correspondence between the tests
10275 identified in the test documentation and the TSFIs described in the functional specification.

10276 **14.3.2.2 Input**

- 10277 a) the ST;

- 10278 b) the functional specification;
- 10279 c) the test documentation;
- 10280 d) the test coverage analysis.
- 10281 **14.3.2.3 Action ATE_COV.2.1E**
- 10282 ISO/IEC 15408-3 ATE_COV.2.1C: *The analysis of the test coverage shall demonstrate the*
 10283 *correspondence between the tests in the test documentation and the TSFIs in the functional*
 10284 *specification.*
- 10285 **14.3.2.3.1 Work unit ATE_COV.2-1**
- 10286 The evaluator ***shall examine*** the test coverage analysis to determine that the correspondence
 10287 between the tests in the test documentation and the interfaces in the functional specification is
 10288 accurate.
- 10289 A simple cross-table may be sufficient to show test correspondence. The identification of the tests
 10290 and the interfaces presented in the test coverage analysis has to be unambiguous.
- 10291 The evaluator is reminded that this does not imply that all tests in the test documentation must
 10292 map to interfaces in the functional specification.
- 10293 **14.3.2.3.2 Work unit ATE_COV.2-2**
- 10294 The evaluator ***shall examine*** the test plan to determine that the testing approach for each interface
 10295 demonstrates the expected behaviour of that interface.
- 10296 Guidance on this work unit can be found in:
- 10297 a) 14.2.1, Understanding the expected behaviour of the TOE
- 10298 b) 14.2.2, Testing vs. alternate approaches to verify the expected behaviour of functionality
- 10299 **14.3.2.3.3 Work unit ATE_COV.2-3**
- 10300 The evaluator ***shall examine*** the test procedures to determine that the test prerequisites, test
 10301 steps and expected result(s) adequately test each interface.
- 10302 Guidance on this work units, as it pertains to the functional specification, can be found in:
- 10303 a) 14.2.3, Verifying the adequacy of tests
- 10304 ISO/IEC 15408-3 ATE_COV.2.2C: *The analysis of the test coverage shall demonstrate that all TSFIs in*
 10305 *the functional specification have been tested.*
- 10306 **14.3.2.3.4 Work unit ATE_COV.2-4**
- 10307 The evaluator ***shall examine*** the test coverage analysis to determine that the correspondence
 10308 between the interfaces in the functional specification and the tests in the test documentation is
 10309 complete.
- 10310 All TSFIs that are described in the functional specification have to be present in the test coverage
 10311 analysis and mapped to tests in order for completeness to be claimed, although exhaustive
 10312 specification testing of interfaces is not required. Incomplete coverage would be evident if an
 10313 interface was identified in the functional specification and no test was mapped to it.

10314 The evaluator is reminded that this does not imply that all tests in the test documentation must
10315 map to interfaces in the functional specification.

10316 **14.3.3 Evaluation of sub-activity (ATE_COV.3)**

10317 **14.3.3.1 Objectives**

10318 The objective of this sub-activity is to determine whether the developer has tested all of the TSFIs
10319 exhaustively, and that the developer's test coverage evidence shows correspondence between the
10320 tests identified in the test documentation and the TSFIs described in the functional specification.

10321 A particular objective of this component is to confirm that all parameters of all of the TSFIs have
10322 been tested.

10323 **14.3.3.2 Input**

10324 The evaluation evidence for this sub-activity is:

- 10325 a) the ST;
- 10326 b) the functional specification;
- 10327 c) the test documentation;
- 10328 d) the test coverage analysis.

10329 **14.3.3.3 Action ATE_COV.3.1E**

10330 ISO/IEC 15408-3 ATE_COV.3.1C: *The analysis of the test coverage shall demonstrate the*
10331 *correspondence between the tests in the test documentation and the TSFIs in the functional*
10332 *specification.*

10333 **14.3.3.3.1 Work unit ATE_COV.3-1**

10334 The evaluator **shall examine** the test coverage analysis to determine that the correspondence
10335 between the tests in the test documentation and the interfaces in the functional specification is
10336 accurate.

10337 A simple cross-table may be sufficient to show test correspondence. The identification of the tests
10338 and the interfaces presented in the test coverage analysis has to be unambiguous.

10339 The evaluator is reminded that this does not imply that all tests in the test documentation must
10340 map to interfaces in the functional specification.

10341 **14.3.3.3.2 Work unit ATE_COV.3-2**

10342 The evaluator **shall examine** the test plan to determine that the testing approach for each interface
10343 demonstrates the expected behaviour of that interface.

10344 Guidance on this work unit can be found in:

- 10345 a) 15.2.1 Understanding the expected behaviour of the TOE
- 10346 b) 15.2.2 [Testing vs. alternate approaches to verify the expected behaviour of
10347 functionality

10348 **14.3.3.3.3 Work unit ATE_COV.3-3**

10349 The evaluator *shall examine* the test procedures to determine that the test prerequisites, test
10350 steps and expected result(s) adequately test each interface.

10351 Guidance on this work units, as it pertains to the functional specification, can be found in:

10352 a) 15.2.3 Verifying the adequacy of tests

10353 ISO/IEC 15408-3 ATE_COV.3.2C *The analysis of the test coverage shall demonstrate that all TSFIs in*
10354 *the functional specification have been completely tested.*

10355 **14.3.3.3.4 Work unit ATE_COV.3-4**

10356 The evaluator *shall examine* the test coverage analysis to determine that the correspondence
10357 between the interfaces in the functional specification and the tests in the test documentation is
10358 complete.

10359 All TSFIs that are described in the functional specification have to be present in the test coverage
10360 analysis and mapped to tests in order for completeness to be claimed. Exhaustive specification
10361 testing of interfaces is required for this mapping. Incomplete coverage would be evident if an
10362 interface was identified in the functional specification and no test was mapped to it.

10363 The evaluator is reminded that this does not imply that all tests in the test documentation must
10364 map to interfaces in the functional specification.

10365 **14.3.3.3.5 Work unit ATE_COV.3-5**

10366 The evaluator *shall examine* the test coverage analysis to determine that the correspondence
10367 between the interfaces in the functional specification and the tests in the test documentation shows
10368 that all TSFIs were tested completely.

10369 This means that the evaluator examines whether all aspects of purpose, method of use, parameters,
10370 parameter descriptions, actions and error messages for all TSFIs present in the functional
10371 specification are covered by the tests. Note that the level of detail present in the functional
10372 specification depends on the component of ADV_FSP chosen in the ST of the TOE.

10373 The evaluator may conclude that the higher level descriptions in the functional specification, like
10374 purpose or method of use, are implicitly covered, if coverage of lower level descriptions like
10375 parameters, parameter descriptions, actions and error messages are covered. Therefore in general
10376 it will only be necessary to confirm coverage on these lower levels.

10377 The evaluator is reminded that (for example) coverage of all parameters does not necessarily mean
10378 coverage of every possible value a parameter may allow. However every value for which a distinct
10379 qualitative behaviour of the TOE is expected, needs to be covered.

10380 As an example: If one of the parameters of a function call is a two byte value, which specifies the
10381 length of further parameters, only some typical values need to be tested. However the evaluator
10382 will make sure that some specific cases (like the value zero or the maximal value) will be covered.

10383 If the evaluator sees that a potential attacker might be able to invoke a TSFI with inconsistent
10384 parameter values (e. g. if one parameter specifies the length of a second parameter and it is
10385 possible to make the second parameter actually longer than the chosen value for the first
10386 parameter suggests) and this case is not covered by the developer's testing, the evaluator may
10387 decide either to test this during their activities in AVA_VAN or to require the developer to provide
10388 coverage also for this case.

10389 Similar considerations as for parameters hold for error messages specified in the functional
10390 specification: Each error message, which belongs to a qualitatively distinct error case, needs to be
10391 covered by testing. Note, that there may be exceptions, for example error messages for errors,
10392 which cannot be provoked during testing. For such error messages other ways of coverage need to
10393 be found as discussed in 15.2.2, "Testing vs. alternate approaches to verify the expected behaviour
10394 of functionality".

10395 Note that also the developer is allowed to use such alternative approaches to testing (e. g. checking
10396 something in the source code) in the coverage table. Of course the evaluator has to examine in this
10397 case, if this use of an alternative approach is acceptable (usually only in cases where testing is
10398 practically impossible).

10399 **14.4 Depth (ATE_DPT)**

10400 **14.4.1 Evaluation of sub-activity (ATE_DPT.1)**

10401 **14.4.1.1 Objectives**

10402 The objective of this sub-activity is to determine whether the developer has tested the TSF
10403 subsystems against the TOE design and the security architecture description.

10404 **14.4.1.2 Input**

- 10405 a) the ST;
- 10406 b) the functional specification;
- 10407 c) the TOE design;
- 10408 d) the security architecture description;
- 10409 e) the test documentation;
- 10410 f) the depth of testing analysis.

10411 **14.4.1.3 Action ATE_DPT.1.1E**

10412 ISO/IEC 15408-3 ATE_DPT.1.1C: *The analysis of the depth of testing shall demonstrate the*
10413 *correspondence between the tests in the test documentation and the TSF subsystems in the TOE*
10414 *design.*

10415 **14.4.1.3.1 Work unit ATE_DPT.1-1**

10416 The evaluator **shall examine** the depth of testing analysis to determine that the descriptions of the
10417 behaviour of TSF subsystems and of their interactions is included within the test documentation.

10418 This work unit verifies the content of the correspondence between the tests and the descriptions in
10419 the TOE design. In cases where the description of the TSF's architectural soundness (in Security
10420 Architecture (ADV_ARC)) cites specific mechanisms, this work unit also verifies the
10421 correspondence between the tests and the descriptions of the behaviour of such mechanisms.

10422 A simple cross-table may be sufficient to show test correspondence. The identification of the tests
10423 and the behaviour/interaction presented in the depth-of coverage analysis has to be unambiguous.

10424 When Evaluation of sub-activity (ATE_DPT.1) is combined with a component of TOE design
10425 (ADV_TDS), which includes descriptions at the module level (e.g. Evaluation of sub-activity
10426 (ADV_TDS.3)), the level of detail needed to map the test cases to the behaviour of the subsystems
10427 may require information from the module description to be used. This is because Evaluation of

10428 sub-activity (ADV_TDS.3) allows the description of details to be shifted from the subsystem level to
10429 the module level, or even to omit the subsystems altogether.

10430 In any case, the required level of detail in the provided reference to the tested behaviour can be
10431 defined as “the level of detail required for the description of subsystem behaviour as defined by
10432 Evaluation of sub-activity (ADV_TDS.2) (in particular work unit ADV_TDS.2-4)”. It states that a
10433 detailed description of the behaviour typically discusses how the functionality is provided, in terms
10434 of what key data and data structures represent; what control relationships exist within a subsystem
10435 and how these elements work together to provide the SFR-enforcing behaviour.

10436 The evaluator is reminded that not all tests in the test documentation must map to a subsystem
10437 behaviour or interaction description.

10438 **14.4.1.3.2 Work unit ATE_DPT.1-2**

10439 The evaluator *shall examine* the test plan, test prerequisites, test steps and expected result(s) to
10440 determine that the testing approach for the behaviour description demonstrates the behaviour of
10441 that subsystem as described in the TOE design.

10442 Guidance on this work unit can be found in:

10443 a) 14.2.1, Understanding the expected behaviour of the TOE

10444 b) 14.2.2, Testing vs. alternate approaches to verify the expected behaviour of functionality

10445 When Evaluation of sub-activity (ATE_DPT.1) is combined with a component of TOE design
10446 (ADV_TDS), which includes descriptions at the module level (e.g. Evaluation of sub-activity
10447 (ADV_TDS.3)), the level of detail needed to map the test cases to the behaviour of the subsystems
10448 may require information from the module description to be used. This is because Evaluation of
10449 sub-activity (ADV_TDS.3) allows the description of details to be shifted from the subsystem level to
10450 the module level, or even to omit the subsystems altogether.

10451 In any case, the required level of detail in the provided reference to the tested behaviour can be
10452 defined as “the level of detail required for the description of subsystem behaviour as defined by
10453 Evaluation of sub-activity (ADV_TDS.2) (in particular work unit ADV_TDS.2-4)”. It states that a
10454 detailed description of the behaviour typically discusses how the functionality is provided, in terms
10455 of what key data and data structures represent; what control relationships exist within a subsystem
10456 and how these elements work together to provide the SFR-enforcing behaviour.

10457 If TSF subsystem interfaces are described, the behaviour of those subsystems may be tested
10458 directly from those interfaces. Otherwise, the behaviour of those subsystems is tested from the
10459 TSFI interfaces. Or a combination of the two may be employed. Whatever strategy is used the
10460 evaluator will consider its appropriateness for adequately testing the behaviour that is described
10461 in the TOE design.

10462 **14.4.1.3.3 Work unit ATE_DPT.1-3**

10463 The evaluator *shall examine* the test plan, test prerequisites, test steps and expected result(s) to
10464 determine that the testing approach for the behaviour description demonstrates the interactions
10465 among subsystems as described in the TOE design.

10466 While the previous work unit addresses behaviour of subsystems, this work unit addresses the
10467 interactions among subsystems.

10468 Guidance on this work unit can be found in:

10469 a) 14.2.1, Understanding the expected behaviour of the TOE

- 10470 b) 14.2.2, Testing vs. alternate approaches to verify the expected behaviour of functionality
- 10471 If TSF subsystem interfaces are described, the interactions with other subsystems may be tested
10472 directly from those interfaces. Otherwise, the interactions among subsystems must be inferred
10473 from the TSFI interfaces. Whatever strategy is used the evaluator will consider its appropriateness
10474 for adequately testing the interactions among subsystems that are described in the TOE design.
- 10475 ISO/IEC 15408-3 ATE_DPT.1.2C: *The analysis of the depth of testing shall demonstrate that all TSF*
10476 *subsystems in the TOE design have been tested.*
- 10477 **14.4.1.3.4 Work unit ATE_DPT.1-4**
- 10478 The evaluator *shall examine* the test procedures to determine that all descriptions of TSF
10479 subsystem behaviour and interaction are tested.
- 10480 This work unit verifies the completeness of work unit ATE_DPT.1-1. All descriptions of TSF
10481 subsystem behaviour and of interactions among TSF subsystems that are provided in the TOE
10482 design have to be tested. Incomplete depth of testing would be evident if a description of TSF
10483 subsystem behaviour or of interactions among TSF subsystems was identified in the TOE design
10484 and no tests could be attributed to it.
- 10485 When Evaluation of sub-activity (ATE_DPT.1) is combined with a component of TOE design
10486 (ADV_TDS), which includes descriptions at the module level (e.g. Evaluation of sub-activity
10487 (ADV_TDS.3)), the level of detail needed to map the test cases to the behaviour of the subsystems
10488 may require information from the module description to be used. This is because Evaluation of
10489 sub-activity (ADV_TDS.3) allows the description of details to be shifted from the subsystem level to
10490 the module level, or even to omit the subsystems altogether.
- 10491 In any case, the required level of detail in the provided reference to the tested behaviour can be
10492 defined as “the level of detail required for the description of subsystem behaviour as defined by
10493 Evaluation of sub-activity (ADV_TDS.2) (in particular work unit ADV_TDS.2-4)”. It states that a
10494 detailed description of the behaviour typically discusses how the functionality is provided, in terms
10495 of what key data and data structures represent; what control relationships exist within a subsystem
10496 and how these elements work together to provide the SFR-enforcing behaviour.
- 10497 The evaluator is reminded that this does not imply that all tests in the test documentation must
10498 map to the subsystem behaviour or interaction description in the TOE design.
- 10499 **14.4.2 Evaluation of sub-activity (ATE_DPT.2)**
- 10500 **14.4.2.1 Objectives**
- 10501 The objective of this sub-activity is to determine whether the developer has tested all the TSF
10502 subsystems and SFR-enforcing modules against the TOE design and the security architecture
10503 description.
- 10504 **14.4.2.2 Input**
- 10505 a) the ST;
- 10506 b) the functional specification;
- 10507 c) the TOE design;
- 10508 d) the security architecture description;
- 10509 e) the test documentation;

- 10510 f) the depth of testing analysis.
- 10511 **14.4.2.3 Action ATE_DPT.2.1E**
- 10512 ISO/IEC 15408-3 ATE_DPT.2.1C: *The analysis of the depth of testing shall demonstrate the*
 10513 *correspondence between the tests in the test documentation and the TSF subsystems and SFR-*
 10514 *enforcing modules in the TOE design.*
- 10515 **14.4.2.3.1 Work unit ATE_DPT.2-1**
- 10516 The evaluator **shall examine** the depth of testing analysis to determine that descriptions of the
 10517 behaviour of TSF subsystems and of their interactions are included within the test documentation.
- 10518 This work unit verifies the content of the correspondence between the tests and the descriptions in
 10519 the TOE design. In cases where the description of the TSF's architectural soundness (in Security
 10520 Architecture (ADV_ARC)) cites specific mechanisms, this work unit also verifies the
 10521 correspondence between the tests and the descriptions of the behaviour of such mechanisms.
- 10522 A simple cross-table may be sufficient to show test correspondence. The identification of the tests
 10523 and the behaviour/interaction presented in the depth-of coverage analysis has to be unambiguous.
- 10524 The evaluator is reminded that not all tests in the test documentation must map to a subsystem
 10525 behaviour or interaction description.
- 10526 **14.4.2.3.2 Work unit ATE_DPT.2-2**
- 10527 The evaluator **shall examine** the test plan, test prerequisites, test steps and expected result(s) to
 10528 determine that the testing approach for the behaviour description demonstrates the behaviour of
 10529 that subsystem as described in the TOE design.
- 10530 Guidance on this work unit can be found in:
- 10531 a) 14.2.1, Understanding the expected behaviour of the TOE
- 10532 b) 14.2.2, Testing vs. alternate approaches to verify the expected behaviour of functionality
- 10533 If TSF subsystem interfaces are described, the behaviour of those subsystems may be tested
 10534 directly from those interfaces. Otherwise, the behaviour of those subsystems is tested from the
 10535 TSFI interfaces. Or a combination of the two may be employed. Whatever strategy is used the
 10536 evaluator will consider its appropriateness for adequately testing the behaviour that is described
 10537 in the TOE design.
- 10538 **14.4.2.3.3 Work unit ATE_DPT.2-3**
- 10539 The evaluator **shall examine** the test plan, test prerequisites, test steps and expected result(s) to
 10540 determine that the testing approach for the behaviour description demonstrates the interactions
 10541 among subsystems as described in the TOE design.
- 10542 While the previous work unit addresses behaviour of subsystems, this work unit addresses the
 10543 interactions among subsystems.
- 10544 Guidance on this work unit can be found in:
- 10545 a) 14.2.1, Understanding the expected behaviour of the TOE
- 10546 b) 14.2.2, Testing vs. alternate approaches to verify the expected behaviour of functionality

10547 If TSF subsystem interfaces are described, the interactions with other subsystems may be tested
10548 directly from those interfaces. Otherwise, the interactions among subsystems must be inferred
10549 from the TSFI interfaces. Whatever strategy is used the evaluator will consider its appropriateness
10550 for adequately testing the interactions among subsystems that are described in the TOE design.

10551 **14.4.2.3.4 Work unit ATE_DPT.2-4**

10552 The evaluator *shall examine* the depth of testing analysis to determine that the interfaces of SFR-
10553 enforcing modules are included within the test documentation.

10554 This work unit verifies the content of the correspondence between the tests and the descriptions in
10555 the TOE design. In cases where the description of the TSF's architectural soundness (in Security
10556 Architecture (ADV_ARC)) cites specific mechanisms at the modular level, this work unit also
10557 verifies the correspondence between the tests and the descriptions of the behaviour of such
10558 mechanisms.

10559 A simple cross-table may be sufficient to show test correspondence. The identification of the tests
10560 and the SFR-enforcing modules presented in the depth-of coverage analysis has to be unambiguous.

10561 The evaluator is reminded that not all tests in the test documentation must map to the interfaces of
10562 SFR-enforcing modules.

10563 **14.4.2.3.5 Work unit ATE_DPT.2-5**

10564 The evaluator *shall examine* the test plan, test prerequisites, test steps and expected result(s) to
10565 determine that the testing approach for each SFR-enforcing module interface demonstrates the
10566 expected behaviour of that interface.

10567 While work unit ATE_DPT.2-2 addresses expected behaviour of subsystems, this work unit
10568 addresses expected behaviour of the SFR-enforcing module interfaces that are covered by
10569 ATE_DPT.2-4.

10570 Guidance on this work unit can be found in:

10571 a) 14.2.1, Understanding the expected behaviour of the TOE

10572 b) 14.2.2, Testing vs. alternate approaches to verify the expected behaviour of functionality

10573 Testing of an interface may be performed directly at that interface, or at the external interfaces, or
10574 a combination of both. Whatever strategy is used the evaluator will consider its appropriateness
10575 for adequately testing the interfaces. Specifically the evaluator determines whether testing at the
10576 internal interfaces is necessary or whether these internal interfaces can be adequately tested
10577 (albeit implicitly) by exercising the external interfaces. This determination is left to the evaluator,
10578 as is its justification.

10579 ISO/IEC 15408-3 ATE_DPT.2.2C: *The analysis of the depth of testing shall demonstrate that all TSF*
10580 *subsystems in the TOE design have been tested.*

10581 **14.4.2.3.6 Work unit ATE_DPT.2-6**

10582 The evaluator *shall examine* the test procedures to determine that all descriptions of TSF
10583 subsystem behaviour and interaction are tested.

10584 This work unit verifies the completeness of work unit ATE_DPT.2-1. All descriptions of TSF
10585 subsystem behaviour and of interactions among TSF subsystems that are provided in the TOE
10586 design have to be tested. Incomplete depth of testing would be evident if a description of TSF
10587 subsystem behaviour or of interactions among TSF subsystems was identified in the TOE design
10588 and no tests could be attributed to it.

- 10589 The evaluator is reminded that this does not imply that all tests in the test documentation must
10590 map to the subsystem behaviour or interaction description in the TOE design.
- 10591 ISO/IEC 15408-3 ATE_DPT.2.3C: *The analysis of the depth of testing shall demonstrate that the SFR-*
10592 *enforcing modules in the TOE design have been tested.*
- 10593 **14.4.2.3.7 Work unit ATE_DPT.2-7**
- 10594 The evaluator **shall examine** the test procedures to determine that all interfaces of SFR-enforcing
10595 modules are tested.
- 10596 This work unit verifies the completeness of work unit ATE_DPT.2-4. All interfaces of SFR-enforcing
10597 modules that are provided in the TOE design have to be tested. Incomplete depth of testing would
10598 be evident if any interface of any SFR-enforcing modules was identified in the TOE design and no
10599 tests could be attributed to it.
- 10600 The evaluator is reminded that this does not imply that all tests in the test documentation must
10601 map to an interface of an SFR-enforcing module in the TOE design.
- 10602 **14.4.3 Evaluation of sub-activity (ATE_DPT.3)**
- 10603 **14.4.3.1 Objectives**
- 10604 The objective of this sub-activity is to determine whether the developer has tested the all the TSF
10605 subsystems and modules against the TOE design and the security architecture description.
- 10606 **14.4.3.2 Input**
- 10607 a) the ST;
- 10608 b) the functional specification;
- 10609 c) the TOE design;
- 10610 d) the security architecture description;
- 10611 e) the test documentation;
- 10612 f) the depth of testing analysis.
- 10613 **14.4.3.3 Action ATE_DPT.3.1E**
- 10614 ISO/IEC 15408-3 ATE_DPT.3.1C: *The analysis of the depth of testing shall demonstrate the*
10615 *correspondence between the tests in the test documentation and the TSF subsystems and modules in*
10616 *the TOE design.*
- 10617 **14.4.3.3.1 Work unit ATE_DPT.3-1**
- 10618 The evaluator **shall examine** the depth of testing analysis to determine that descriptions of the
10619 behaviour of TSF subsystems and of their interactions are included within the test documentation.
- 10620 This work unit verifies the content of the correspondence between the tests and the descriptions in
10621 the TOE design. A simple cross-table may be sufficient to show test correspondence. The
10622 identification of the tests and the behaviour/interaction presented in the depth-of coverage
10623 analysis has to be unambiguous.
- 10624 The evaluator is reminded that not all tests in the test documentation must map to a subsystem
10625 behaviour or interaction description.

10626 **14.4.3.3.2 Work unit ATE_DPT.3-2**

10627 The evaluator *shall examine* the test plan, test prerequisites, test steps and expected result(s) to
10628 determine that the testing approach for the behaviour description demonstrates the behaviour of
10629 that subsystem as described in the TOE design.

10630 Guidance on this work unit can be found in:

10631 a) 14.2.1, Understanding the expected behaviour of the TOE

10632 b) 14.2.2, Testing vs. alternate approaches to verify the expected behaviour of functionality

10633 If TSF subsystem interfaces are provided, the behaviour of those subsystems may be performed
10634 directly from those interfaces. Otherwise, the behaviour of those subsystems is tested from the
10635 TSFI interfaces. Or a combination of the two may be employed. Whatever strategy is used the
10636 evaluator will consider its appropriateness for adequately testing the behaviour that is described
10637 in the TOE design.

10638 **14.4.3.3.3 Work unit ATE_DPT.3-3**

10639 The evaluator *shall examine* the test plan, test prerequisites, test steps and expected result(s) to
10640 determine that the testing approach for the behaviour description demonstrates the interactions
10641 among subsystems as described in the TOE design.

10642 Guidance on this work unit can be found in:

10643 a) 14.2.1, Understanding the expected behaviour of the TOE

10644 b) 14.2.2, Testing vs. alternate approaches to verify the expected behaviour of functionality

10645 While the previous work unit addresses behaviour of subsystems, this work unit addresses the
10646 interactions among subsystems.

10647 If TSF subsystem interfaces are provided, the interactions with other subsystems may be
10648 performed directly from those interfaces. Otherwise, the interactions among subsystems must be
10649 inferred from the TSFI interfaces. Whatever strategy is used the evaluator will consider its
10650 appropriateness for adequately testing the interactions among subsystems that are described in
10651 the TOE design.

10652 **14.4.3.3.4 Work unit ATE_DPT.3-4**

10653 The evaluator *shall examine* the depth of testing analysis to determine that the interfaces of TSF
10654 modules are included within the test documentation.

10655 This work unit verifies the content of the correspondence between the tests and the descriptions in
10656 the TOE design. A simple cross-table may be sufficient to show test correspondence. The
10657 identification of the tests and the behaviour/interaction presented in the depth-of coverage
10658 analysis has to be unambiguous.

10659 The evaluator is reminded that not all tests in the test documentation must map to a subsystem
10660 behaviour or interaction description.

10661 **14.4.3.3.5 Work unit ATE_DPT.3-5**

10662 The evaluator *shall examine* the test plan, test prerequisites, test steps and expected result(s) to
10663 determine that the testing approach for each TSF module interface demonstrates the expected
10664 behaviour of that interface.

- 10665 Guidance on this work unit can be found in:
- 10666 a) 14.2.1, Understanding the expected behaviour of the TOE
- 10667 b) 14.2.2, Testing vs. alternate approaches to verify the expected behaviour of functionality
- 10668 Testing of an interface may be performed directly at that interface, or at the external interfaces, or
 10669 a combination of both. Whatever strategy is used the evaluator will consider its appropriateness
 10670 for adequately testing the interfaces. Specifically the evaluator determines whether testing at the
 10671 internal interfaces is necessary or whether these internal interfaces can be adequately tested
 10672 (albeit implicitly) by exercising the external interfaces. This determination is left to the evaluator,
 10673 as is its justification.
- 10674 ISO/IEC 15408-3 ATE_DPT.3.2C: *The analysis of the depth of testing shall demonstrate that all TSF*
 10675 *subsystems in the TOE design have been tested.*
- 10676 **14.4.3.3.6 Work unit ATE_DPT.3-6**
- 10677 The evaluator ***shall examine*** the test procedures to determine that all descriptions of TSF
 10678 subsystem behaviour and interaction are tested.
- 10679 This work unit verifies the completeness of work unit ATE_DPT.3-1. All descriptions of TSF
 10680 subsystem behaviour and of interactions among TSF subsystems that are provided in the TOE
 10681 design have to be tested. Incomplete depth of testing would be evident if a description of TSF
 10682 subsystem behaviour or of interactions among TSF subsystems was identified in the TOE design
 10683 and no tests could be attributed to it.
- 10684 The evaluator is reminded that this does not imply that all tests in the test documentation must
 10685 map to the subsystem behaviour or interaction description in the TOE design.
- 10686 ISO/IEC 15408-3 ATE_DPT.3.3C: *The analysis of the depth of testing shall demonstrate that all TSF*
 10687 *modules in the TOE design have been tested.*
- 10688 **14.4.3.3.7 Work unit ATE_DPT.3-7**
- 10689 The evaluator ***shall examine*** the test procedures to determine that all interfaces of all TSF modules
 10690 are tested.
- 10691 This work unit verifies the completeness of work unit ATE_DPT.3-4. All interfaces of TSF modules
 10692 that are provided in the TOE design have to be tested. Incomplete depth of testing would be
 10693 evident if any interface of any TSF module was identified in the TOE design and no tests could be
 10694 attributed to it.
- 10695 The evaluator is reminded that this does not imply that all tests in the test documentation must
 10696 map to an interface of a TSF module in the TOE design.
- 10697 **14.4.4 Evaluation of sub-activity (ATE_DPT.4)**
- 10698 There is no general guidance; the scheme should be consulted for guidance on this sub-activity.
- 10699 **14.5 Functional tests (ATE_FUN)**
- 10700 **14.5.1 Evaluation of sub-activity (ATE_FUN.1)**
- 10701 **14.5.1.1 Objectives**
- 10702 The objective of this sub-activity is to determine whether the developer correctly performed and
 10703 documented the tests in the test documentation.

10704 **14.5.1.2 Input**

10705 The evaluation evidence for this sub-activity is:

- 10706 a) the ST;
- 10707 b) the functional specification;
- 10708 c) the test documentation.

10709 **14.5.1.3 Application notes**

10710 The extent to which the test documentation is required to cover the TSF is dependent upon the
10711 coverage assurance component.

10712 For the developer tests provided, the evaluator determines whether the tests are repeatable, and
10713 the extent to which the developer's tests can be used for the evaluator's independent testing effort.
10714 Any TSFI for which the developer's test results indicate that it might not perform as specified
10715 should be tested independently by the evaluator to determine whether or not it does.

10716 **14.5.1.4 Action ATE_FUN.1.1E**

10717 ISO/IEC 15408-3 ATE_FUN.1.1C: *The test documentation shall consist of test plans, expected test*
10718 *results and actual test results.*

10719 **14.5.1.4.1 Work unit ATE_FUN.1-1**

10720 The evaluator **shall check** that the test documentation includes test plans, expected test results and
10721 actual test results.

10722 The evaluator checks that test plans, expected tests results and actual test results are included in
10723 the test documentation.

10724 ISO/IEC 15408-3 ATE_FUN.1.2C: *The test plans shall identify the tests to be performed and describe*
10725 *the scenarios for performing each test. These scenarios shall include any ordering dependencies on the*
10726 *results of other tests.*

10727 **14.5.1.4.2 Work unit ATE_FUN.1-2**

10728 The evaluator **shall examine** the test plan to determine that it describes the scenarios for
10729 performing each test.

10730 The evaluator determines that the test plan provides information about the test configuration
10731 being used: both on the configuration of the TOE and on any test equipment being used. This
10732 information should be detailed enough to ensure that the test configuration is reproducible.

10733 The evaluator also determines that the test plan provides information about how to execute the
10734 test: any necessary automated set-up procedures (and whether they require privilege to run),
10735 inputs to be applied, how these inputs are applied, how output is obtained, any automated clean-up
10736 procedures (and whether they require privilege to run), etc. This information should be detailed
10737 enough to ensure that the test is reproducible.

10738 The evaluator may wish to employ a sampling strategy when performing this work unit.

10739 **14.5.1.4.3 Work unit ATE_FUN.1-3**

10740 The evaluator **shall examine** the test plan to determine that the TOE test configuration is
10741 consistent with the ST.

- 10742 The TOE referred to in the developer's test plan should have the same unique reference as
10743 established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST introduction.
- 10744 It is possible for the ST to specify more than one configuration for evaluation. The evaluator
10745 verifies that all test configurations identified in the developer test documentation are consistent
10746 with the ST. For example, the ST might define configuration options that must be set, which could
10747 have an impact upon what constitutes the TOE by including or excluding additional portions. The
10748 evaluator verifies that all such variations of the TOE are considered.
- 10749 The evaluator should consider the security objectives for the operational environment described in
10750 the ST that may apply to the test environment. There may be some objectives for the operational
10751 environment that do not apply to the test environment. For example, an objective about user
10752 clearances may not apply; however, an objective about a single point of connection to a network
10753 would apply.
- 10754 The evaluator may wish to employ a sampling strategy when performing this work unit.
- 10755 If this work unit is applied to a component TOE that might be used/integrated in a composed TOE
10756 (see Class ACO: Composition), the following will apply. In the instances that the component TOE
10757 under evaluation depends on other components in the operational environment to support their
10758 operation, the developer may wish to consider using the other component(s) that will be used in
10759 the composed TOE to fulfil the requirements of the operational environment as one of the test
10760 configurations. This will reduce the amount an additional testing that will be required for the
10761 composed TOE evaluation.
- 10762 **14.5.1.4.4 Work unit ATE_FUN.1-4**
- 10763 The evaluator *shall examine* the test plans to determine that sufficient instructions are provided
10764 for any ordering dependencies.
- 10765 Some steps may have to be performed to establish initial conditions. For example, user accounts
10766 need to be added before they can be deleted. An example of ordering dependencies on the results
10767 of other tests is the need to perform actions in a test that will result in the generation of audit
10768 records, before performing a test to consider the searching and sorting of those audit records.
10769 Another example of an ordering dependency would be where one test case generates a file of data
10770 to be used as input for another test case.
- 10771 The evaluator may wish to employ a sampling strategy when performing this work unit.
- 10772 ISO/IEC 15408-3 ATE_FUN.1.3C: *The expected test results shall show the anticipated outputs from a*
10773 *successful execution of the tests.*
- 10774 **14.5.1.4.5 Work unit ATE_FUN.1-5**
- 10775 The evaluator *shall examine* the test documentation to determine that all expected tests results
10776 are included.
- 10777 The expected test results are needed to determine whether or not a test has been successfully
10778 performed. Expected test results are sufficient if they are unambiguous and consistent with
10779 expected behaviour given the testing approach.
- 10780 The evaluator may wish to employ a sampling strategy when performing this work unit.
- 10781 ISO/IEC 15408-3 ATE_FUN.1.4C: *The actual test results shall be consistent with the expected test*
10782 *results.*

10783 **14.5.1.4.6 Work unit ATE_FUN.1-6**

10784 The evaluator **shall check** that the actual test results in the test documentation are consistent with
10785 the expected test results in the test documentation.

10786 A comparison of the actual and expected test results provided by the developer will reveal any
10787 inconsistencies between the results. It may be that a direct comparison of actual results cannot be
10788 made until some data reduction or synthesis has been first performed. In such cases, the
10789 developer's test documentation should describe the process to reduce or synthesise the actual data.

10790 For example, the developer may need to test the contents of a message buffer after a network
10791 connection has occurred to determine the contents of the buffer. The message buffer will contain a
10792 binary number. This binary number would have to be converted to another form of data
10793 representation in order to make the test more meaningful. The conversion of this binary
10794 representation of data into a higher-level representation will have to be described by the developer
10795 in enough detail to allow an evaluator to perform the conversion process (i.e. synchronous or
10796 asynchronous transmission, number of stop bits, parity, etc.).

10797 It should be noted that the description of the process used to reduce or synthesise the actual data is
10798 used by the evaluator not to actually perform the necessary modification but to assess whether this
10799 process is correct. It is up to the developer to transform the expected test results into a format that
10800 allows an easy comparison with the actual test results.

10801 The evaluator may wish to employ a sampling strategy when performing this work unit.

10802 **14.5.1.4.7 Work unit ATE_FUN.1-7**

10803 The evaluator **shall report** the developer testing effort, outlining the testing approach,
10804 configuration, depth and results.

10805 The developer testing information recorded in the ETR allows the evaluator to convey the overall
10806 testing approach and effort expended on the testing of the TOE by the developer. The intent of
10807 providing this information is to give a meaningful overview of the developer testing effort. It is not
10808 intended that the information regarding developer testing in the ETR be an exact reproduction of
10809 specific test steps or results of individual tests. The intention is to provide enough detail to allow
10810 other evaluators and evaluation authorities to gain some insight about the developer's testing
10811 approach, amount of testing performed, TOE test configurations, and the overall results of the
10812 developer testing.

10813 Information that would typically be found in the ETR subclause regarding the developer testing
10814 effort is:

10815 a) TOE test configurations. The particular configurations of the TOE that were tested,
10816 including whether any privileged code was required to set up the test or clean up
10817 afterwards;

10818 b) testing approach. An account of the overall developer testing strategy employed;

10819 c) testing results. A description of the overall developer testing results.

10820 This list is by no means exhaustive and is only intended to provide some context as to the type of
10821 information that should be present in the ETR concerning the developer testing effort.

10822 **14.5.2 Evaluation of sub-activity (ATE_FUN.2)**10823 **14.5.2.1 Objectives**

10824 The objective of this sub-activity is to determine whether the developer correctly performed and
 10825 documented the tests in the test documentation and to ensure that testing is structured such as to
 10826 avoid circular arguments about the correctness of the interfaces being tested.

10827 **14.5.2.2 Input**

10828 The evaluation evidence for this sub-activity is:

- 10829 a) the ST;
- 10830 b) the functional specification;
- 10831 c) the test documentation.

10832 **14.5.2.3 Application notes**

10833 Although the test procedures may state pre-requisite initial test conditions in terms of ordering of
 10834 tests, they may not provide a rationale for the ordering. An analysis of test ordering, which
 10835 provides this rationale, is an important factor in determining the adequacy of testing, as there is a
 10836 possibility of faults being concealed by the ordering of tests.

10837 **14.5.2.4 Action ATE_FUN.2.1E**

10838 ISO/IEC 15408-3 ATE_FUN.2.1C *The test documentation shall consist of test plans, expected test*
 10839 *results and actual test results.*

10840 **14.5.2.4.1 Work unit ATE_FUN.2-1**

10841 The evaluator **shall check** that the test documentation includes test plans, expected test results and
 10842 actual test results.

10843 The evaluator checks that test plans, expected tests results and actual test results are included in
 10844 the test documentation.

10845 ISO/IEC 15408-3 ATE_FUN.2.2C *The test plans shall identify the tests to be performed and describe*
 10846 *the scenarios for performing each test. These scenarios shall include any ordering dependencies on the*
 10847 *results of other tests.*

10848 **14.5.2.4.2 Work unit ATE_FUN.2-2**

10849 The evaluator **shall examine** the test plan to determine that it describes the scenarios for
 10850 performing each test.

10851 The evaluator determines that the test plan provides information about the test configuration
 10852 being used: both on the configuration of the TOE and on any test equipment being used. This
 10853 information should be detailed enough to ensure that the test configuration is reproducible.

10854 The evaluator also determines that the test plan provides information about how to execute the
 10855 test: any necessary automated set-up procedures (and whether they require privilege to run),
 10856 inputs to be applied, how these inputs are applied, how output is obtained, any automated clean-up
 10857 procedures (and whether they require privilege to run), etc. This information should be detailed
 10858 enough to ensure that the test is reproducible.

10859 The evaluator may wish to employ a sampling strategy when performing this work unit.

10860 **14.5.2.4.3 Work unit ATE_FUN.2-3**

10861 The evaluator *shall examine* the test plan to determine that the TOE test configuration is
10862 consistent with the ST.

10863 The TOE referred to in the developer's test plan should have the same unique reference as
10864 established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST introduction.

10865 It is possible for the ST to specify more than one configuration for evaluation. The evaluator
10866 verifies that all test configurations identified in the developer test documentation are consistent
10867 with the ST. For example, the ST might define configuration options that must be set, which could
10868 have an impact upon what constitutes the TOE by including or excluding additional portions. The
10869 evaluator verifies that all such variations of the TOE are considered.

10870 The evaluator should consider the security objectives for the operational environment described in
10871 the ST that may apply to the test environment. There may be some objectives for the operational
10872 environment that do not apply to the test environment. For example, an objective about user
10873 clearances may not apply; however, an objective about a single point of connection to a network
10874 would apply.

10875 The evaluator may wish to employ a sampling strategy when performing this work unit.

10876 If this work unit is applied to a component TOE that might be used/integrated in a composed TOE
10877 (see Class ACO: Composition), the following will apply. In the instances that the component TOE
10878 under evaluation depends on other components in the operational environment to support their
10879 operation, the developer may wish to consider using the other component(s) that will be used in
10880 the composed TOE to fulfil the requirements of the operational environment as one of the test
10881 configurations. This will reduce the amount an additional testing that will be required for the
10882 composed TOE evaluation.

10883 **14.5.2.4.4 Work unit ATE_FUN.2-4**

10884 The evaluator *shall examine* the test plans to determine that sufficient instructions are provided
10885 for any ordering dependencies.

10886 Some steps may have to be performed to establish initial conditions. For example, user accounts
10887 need to be added before they can be deleted. An example of ordering dependencies on the results
10888 of other tests is the need to perform actions in a test that will result in the generation of audit
10889 records, before performing a test to consider the searching and sorting of those audit records.
10890 Another example of an ordering dependency would be where one test case generates a file of data
10891 to be used as input for another test case.

10892 The evaluator may wish to employ a sampling strategy when performing this work unit.

10893 ATE_FUN.2.3C *The expected test results shall show the anticipated outputs from a successful*
10894 *execution of the tests.*

10895 **14.5.2.4.5 Work unit ATE_FUN.2-5**

10896 The evaluator *shall examine* the test documentation to determine that all expected tests results
10897 are included.

10898 The expected test results are needed to determine whether or not a test has been successfully
10899 performed. Expected test results are sufficient if they are unambiguous and consistent with
10900 expected behaviour given the testing approach.

10901 The evaluator may wish to employ a sampling strategy when performing this work unit.

10902 **ATE_FUN.2.4C** *The actual test results shall be consistent with the expected test results.*

10903 **14.5.2.4.6 Work unit ATE_FUN.2-6**

10904 The evaluator **shall check** that the actual test results in the test documentation are consistent with
10905 the expected test results in the test documentation.

10906 A comparison of the actual and expected test results provided by the developer will reveal any
10907 inconsistencies between the results. It may be that a direct comparison of actual results cannot be
10908 made until some data reduction or synthesis has been first performed. In such cases, the
10909 developer's test documentation should describe the process to reduce or synthesise the actual data.

10910 For example, the developer may need to test the contents of a message buffer after a network
10911 connection has occurred to determine the contents of the buffer. The message buffer will contain a
10912 binary number. This binary number would have to be converted to another form of data
10913 representation in order to make the test more meaningful. The conversion of this binary
10914 representation of data into a higher-level representation will have to be described by the developer
10915 in enough detail to allow an evaluator to perform the conversion process (i.e. synchronous or
10916 asynchronous transmission, number of stop bits, parity, etc.).

10917 It should be noted that the description of the process used to reduce or synthesise the actual data is
10918 used by the evaluator not to actually perform the necessary modification but to assess whether this
10919 process is correct. It is up to the developer to transform the expected test results into a format that
10920 allows an easy comparison with the actual test results.

10921 The evaluator may wish to employ a sampling strategy when performing this work unit.

10922 **14.5.2.4.7 Work unit ATE_FUN.2-7**

10923 The evaluator **shall report** the developer testing effort, outlining the testing approach,
10924 configuration, depth and results.

10925 The developer testing information recorded in the ETR allows the evaluator to convey the overall
10926 testing approach and effort expended on the testing of the TOE by the developer. The intent of
10927 providing this information is to give a meaningful overview of the developer testing effort. It is not
10928 intended that the information regarding developer testing in the ETR be an exact reproduction of
10929 specific test steps or results of individual tests. The intention is to provide enough detail to allow
10930 other evaluators and evaluation authorities to gain some insight about the developer's testing
10931 approach, amount of testing performed, TOE test configurations, and the overall results of the
10932 developer testing.

10933 Information that would typically be found in the ETR section regarding the developer testing effort
10934 is:

10935 a) TOE test configurations. The particular configurations of the TOE that were
10936 tested, including whether any privileged code was required to set up the test or
10937 clean up afterwards;

10938 b) testing approach. An account of the overall developer testing strategy
10939 employed;

10940 c) testing results. A description of the overall developer testing results.

10941 This list is by no means exhaustive and is only intended to provide some context as to the type of
10942 information that should be present in the ETR concerning the developer testing effort.

10943 **ATE_FUN.2.5C** *The test documentation shall include an analysis of the test procedure ordering
10944 dependencies.*

10945 **14.5.2.4.8 Work unit ATE_FUN.2-8**

10946 The evaluator *shall examine* the analysis of the test procedure ordering dependencies to
10947 determine that a sufficient justification for the chosen ordering of test cases is given.

10948 Usually the evaluator will generate a table of all cases, where the test documentation requires a
10949 certain ordering of the tests and will then examine if sufficient justification is given in any case,
10950 why testing in this ordering is adequate and sufficient.

10951 As an example we assume that the TSF provide a random number generator, which needs to be
10952 initialised (for example with an adequate seed) before random numbers of a specified quality can
10953 be generated. In this case the evaluator will consider the following question:

10954 Does the test documentation only describe an ordering of tests, where the initialisation is done
10955 before calling the function to generate a random number?

10956 In this case the justification needs to show, why the developer expects, that in the intended
10957 environment of the TOE the random number function will not be called without initialisation of the
10958 random number generator.

10959 If for example the user guidance documentation includes a clear instruction that the random
10960 number generator needs to be initialised adequately before being called, this may be considered
10961 adequate as a justification. (note that the question if it can be plausibly assumed that users will
10962 follow such instruction is covered by the evaluation activities for the classes ASE and AGD and
10963 needs not to be re-examined here.)

10964 If, on the other hand, the TOE provides an authentication protocol, which implicitly uses random
10965 numbers provided by the random number generator, and an attacker can therefore "call" the
10966 random number generator implicitly by simply trying to authenticate himself, and if neither the
10967 TOE nor the environment prevent an attacker from doing this even before the random number
10968 generator is initialised, a test case needs to show, what happens in such situation.

10969 If, for example, instead of returning a "bad" random number, the random number function would
10970 return an error, when called without proper initialisation, it would be much better to include a test
10971 showing this secure behaviour instead of trying to justify why the functions are only tested in the
10972 usual order.

10973 Note: Of course even without ATE_FUN.2 an evaluator would be expected to look for potential
10974 vulnerabilities like the one described above. However, ATE_FUN.2.5C adds assurance by requiring
10975 the developer to give a systematic justification, why their chosen order of test cases doesn't hide
10976 such potential failures of security functions.

10977 **14.6 Independent testing (ATE_IND)**

10978 **14.6.1 Evaluation of sub-activity (ATE_IND.1)**

10979 **14.6.1.1 Objectives**

10980 The goal of this activity is to determine, by independently testing a subset of the TSFI, whether the
10981 TOE behaves as specified in the functional specification and guidance documentation.

10982 **14.6.1.2 Input**

10983 The evaluation evidence for this sub-activity is:

10984 a) the ST;

10985 b) the functional specification;

- 10986 c) the operational user guidance;
- 10987 d) the preparative user guidance;
- 10988 e) the TOE suitable for testing.
- 10989 **14.6.1.3 Action ATE_IND.1.1E**
- 10990 ISO/IEC 15408-3 ATE_IND.1.1C: *The TOE shall be suitable for testing.*
- 10991 **14.6.1.3.1 Work unit ATE_IND.1-1**
- 10992 The evaluator **shall examine** the TOE to determine that the test configuration is consistent with
10993 the configuration under evaluation as specified in the ST.
- 10994 The TOE provided by the developer should have the same unique reference as established by the
10995 CM capabilities (ALC_CMC) sub-activities and identified in the ST introduction.
- 10996 It is possible for the ST to specify more than one configuration for evaluation. The TOE may
10997 comprise a number of distinct hardware and software entities that need to be tested in accordance
10998 with the ST. The evaluator verifies that all test configurations are consistent with the ST.
- 10999 The evaluator should consider the security objectives for the operational environment described in
11000 the ST that may apply to the test environment and ensure they are met in the testing environment.
11001 There may be some objectives for the operational environment that do not apply to the test
11002 environment. For example, an objective about user clearances may not apply; however, an
11003 objective about a single point of connection to a network would apply.
- 11004 If any test resources are used (e.g. meters, analysers) it will be the evaluator's responsibility to
11005 ensure that these resources are calibrated correctly.
- 11006 **14.6.1.3.2 Work unit ATE_IND.1-2**
- 11007 The evaluator **shall examine** the TOE to determine that it has been installed properly and is in a
11008 known state.
- 11009 It is possible for the evaluator to determine the state of the TOE in a number of ways. For example,
11010 previous successful completion of the Evaluation of sub-activity (AGD_PRE.1) will satisfy this work
11011 unit if the evaluator still has confidence that the TOE being used for testing was installed properly
11012 and is in a known state. If this is not the case, then the evaluator should follow the developer's
11013 procedures to install and start up the TOE, using the supplied guidance only.
- 11014 If the evaluator has to perform the installation procedures because the TOE is in an unknown state,
11015 this work unit when successfully completed could satisfy work unit AGD_PRE.1-3.
- 11016 **14.6.1.4 Action ATE_IND.1.2E**
- 11017 **14.6.1.4.1 Work unit ATE_IND.1-3**
- 11018 The evaluator **shall devise** a test subset.
- 11019 The evaluator selects a test subset and testing strategy that is appropriate for the TOE. One
11020 extreme testing strategy would be to have the test subset contain as many interfaces as possible
11021 tested with little rigour. Another testing strategy would be to have the test subset contain a few
11022 interfaces based on their perceived relevance and rigorously test these interfaces.

11023 Typically the testing approach taken by the evaluator should fall somewhere between these two
11024 extremes. The evaluator should exercise most of the interfaces using at least one test, but testing
11025 need not demonstrate exhaustive specification testing.

11026 The evaluator, when selecting the subset of the interfaces to be tested, should consider the
11027 following factors:

11028 a) The number of interfaces from which to draw upon for the test subset. Where the TSF
11029 includes only a small number of relatively simple interfaces, it may be practical to
11030 rigorously test all of the interfaces. In other cases this may not be cost-effective, and
11031 sampling is required.

11032 b) Maintaining a balance of evaluation activities. The evaluator effort expended on the test
11033 activity should be commensurate with that expended on any other evaluation activity.

11034 The evaluator selects the interfaces to compose the subset. This selection will depend on a number
11035 of factors, and consideration of these factors may also influence the choice of test subset size:

11036 a) Significance of interfaces. Those interfaces more significant than others should be
11037 included in the test subset. One major factor of “significance” is the security-relevance
11038 (SFR-enforcing interfaces would be more significant than SFR-supporting interfaces,
11039 which are more significant than SFR-non-interfering interfaces; see ISO/IEC 15408-3
11040 Subclause Functional specification (ADV_FSP)). The other major factor of “significance” is
11041 the number of SFRs mapping to this interface (as determined when identifying the
11042 correspondence between levels of abstraction in ADV).

11043 b) Complexity of the interface. Complex interfaces may require complex tests that impose
11044 onerous requirements on the developer or evaluator, which may not be conducive to
11045 cost-effective evaluations. Conversely, they are a likely area to find errors and are good
11046 candidates for the subset. The evaluator will need to strike a balance between these
11047 considerations.

11048 c) Implicit testing. Testing some interfaces may often implicitly test other interfaces, and
11049 their inclusion in the subset may maximise the number of interfaces tested (albeit
11050 implicitly). Certain interfaces will typically be used to provide a variety of security
11051 functionality, and will tend to be the target of an effective testing approach.

11052 d) Types of interfaces (e.g. programmatic, command-line, protocol). The evaluator should
11053 consider including tests for all different types of interfaces that the TOE supports.

11054 e) Interfaces that give rise to features that are innovative or unusual. Where the TOE
11055 contains innovative or unusual features, which may feature strongly in marketing
11056 literature and guidance documents, the corresponding interfaces should be strong
11057 candidates for testing.

11058 This guidance articulates factors to consider during the selection process of an appropriate test
11059 subset, but these are by no means exhaustive.

11060 **14.6.1.4.2 Work unit ATE_IND.1-4**

11061 The evaluator **shall produce** test documentation for the test subset that is sufficiently detailed to
11062 enable the tests to be reproducible.

11063 With an understanding of the expected behaviour of the TSF, from the ST and the functional
11064 specification, the evaluator has to determine the most feasible way to test the interface. Specifically
11065 the evaluator considers:

- 11066 a) the approach that will be used, for instance, whether an external interface will be tested,
11067 or an internal interface using a test harness, or will an alternate test approach be
11068 employed (e.g. in exceptional circumstances, a code inspection, if the implementation
11069 representation is available);
- 11070 b) the interface(s) that will be used to test and observe responses;
- 11071 c) the initial conditions that will need to exist for the test (i.e. any particular objects or
11072 subjects that will need to exist and security attributes they will need to have);
- 11073 d) special test equipment that will be required to either stimulate an interface (e.g. packet
11074 generators) or make observations of an interface (e.g. network analysers).
- 11075 The evaluator may find it practical to test each interface using a series of test cases, where each test
11076 case will test a very specific aspect of expected behaviour.
- 11077 The evaluator's test documentation should specify the derivation of each test, tracing it back to the
11078 relevant interface(s).
- 11079 **14.6.1.4.3 Work unit ATE_IND.1-5**
- 11080 The evaluator **shall conduct** testing.
- 11081 The evaluator uses the test documentation developed as a basis for executing tests on the TOE. The
11082 test documentation is used as a basis for testing but this does not preclude the evaluator from
11083 performing additional ad hoc tests. The evaluator may devise new tests based on behaviour of the
11084 TOE discovered during testing. These new tests are recorded in the test documentation.
- 11085 **14.6.1.4.4 Work unit ATE_IND.1-6**
- 11086 The evaluator **shall record** the following information about the tests that compose the test subset:
- 11087 a) identification of the interface behaviour to be tested;
- 11088 b) instructions to connect and setup all required test equipment as required to conduct the
11089 test;
- 11090 c) instructions to establish all prerequisite test conditions;
- 11091 d) instructions to stimulate the interface;
- 11092 e) instructions for observing the behaviour of the interface;
- 11093 f) descriptions of all expected results and the necessary analysis to be performed on the
11094 observed behaviour for comparison against expected results;
- 11095 g) instructions to conclude the test and establish the necessary post-test state for the TOE;
- 11096 h) actual test results.
- 11097 The level of detail should be such that another evaluator could repeat the tests and obtain an
11098 equivalent result. While some specific details of the test results may be different (e.g. time and date
11099 fields in an audit record) the overall result should be identical.
- 11100 There may be instances when it is unnecessary to provide all the information presented in this
11101 work unit (e.g. the actual test results of a test may not require any analysis before a comparison
11102 between the expected results can be made). The determination to omit this information is left to
11103 the evaluator, as is the justification.

11104 **14.6.1.4.5 Work unit ATE_IND.1-7**

11105 The evaluator *shall check* that all actual test results are consistent with the expected test results.

11106 Any differences in the actual and expected test results may indicate that the TOE does not perform
11107 as specified or that the evaluator test documentation may be incorrect. Unexpected actual results
11108 may require corrective maintenance to the TOE or test documentation and perhaps require re-
11109 running of impacted tests and modifying the test sample size and composition. This determination
11110 is left to the evaluator, as is its justification.

11111 **14.6.1.4.6 Work unit ATE_IND.1-8**

11112 The evaluator *shall report* in the ETR the evaluator testing effort, outlining the testing approach,
11113 configuration, depth and results.

11114 The evaluator testing information reported in the ETR allows the evaluator to convey the overall
11115 testing approach and effort expended on the testing activity during the evaluation. The intent of
11116 providing this information is to give a meaningful overview of the testing effort. It is not intended
11117 that the information regarding testing in the ETR be an exact reproduction of specific test
11118 instructions or results of individual tests. The intention is to provide enough detail to allow other
11119 evaluators and evaluation authorities to gain some insight about the testing approach chosen,
11120 amount of testing performed, TOE test configurations, and the overall results of the testing activity.

11121 Information that would typically be found in the ETR subclause regarding the evaluator testing
11122 effort is:

11123 a) TOE test configurations. The particular configurations of the TOE that were tested;

11124 b) subset size chosen. The amount of interfaces that were tested during the evaluation and a
11125 justification for the size;

11126 c) selection criteria for the interfaces that compose the subset. Brief statements about the
11127 factors considered when selecting interfaces for inclusion in the subset;

11128 d) interfaces tested. A brief listing of the interfaces that merited inclusion in the subset;

11129 e) verdict for the activity. The overall judgement on the results of testing during the
11130 evaluation.

11131 This list is by no means exhaustive and is only intended to provide some context as to the type of
11132 information that should be present in the ETR concerning the testing the evaluator performed
11133 during the evaluation.

11134 **14.6.2 Evaluation of sub-activity (ATE_IND.2)**

11135 **14.6.2.1 Objectives**

11136 The goal of this activity is to determine, by independently testing a subset of the TSF, whether the
11137 TOE behaves as specified in the design documentation, and to gain confidence in the developer's
11138 test results by performing a sample of the developer's tests.

11139 **14.6.2.2 Input**

11140 The evaluation evidence for this sub-activity is:

11141 a) the ST;

11142 b) the functional specification;

- 11143 c) the TOE design description;
- 11144 d) the operational user guidance;
- 11145 e) the preparative user guidance;
- 11146 f) the configuration management documentation;
- 11147 g) the test documentation;
- 11148 h) the TOE suitable for testing.
- 11149 **14.6.2.3 Action ATE_IND.2.1E**
- 11150 ISO/IEC 15408-3 ATE_IND.2.1C: *The TOE shall be suitable for testing.*
- 11151 **14.6.2.3.1 Work unit ATE_IND.2-1**
- 11152 The evaluator ***shall examine*** the TOE to determine that the test configuration is consistent with
11153 the configuration under evaluation as specified in the ST.
- 11154 The TOE provided by the developer and identified in the test plan should have the same unique
11155 reference as established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST
11156 introduction.
- 11157 It is possible for the ST to specify more than one configuration for evaluation. The TOE may
11158 comprise a number of distinct hardware and software entities that need to be tested in accordance
11159 with the ST. The evaluator verifies that all test configurations are consistent with the ST.
- 11160 The evaluator should consider the security objectives for the operational environment described in
11161 the ST that may apply to the test environment and ensure they are met in the testing environment.
11162 There may be some objectives for the operational environment that do not apply to the test
11163 environment. For example, an objective about user clearances may not apply; however, an
11164 objective about a single point of connection to a network would apply.
- 11165 If any test resources are used (e.g. meters, analysers) it will be the evaluator's responsibility to
11166 ensure that these resources are calibrated correctly.
- 11167 **14.6.2.3.2 Work unit ATE_IND.2-2**
- 11168 The evaluator ***shall examine*** the TOE to determine that it has been installed properly and is in a
11169 known state.
- 11170 It is possible for the evaluator to determine the state of the TOE in a number of ways. For example,
11171 previous successful completion of the Evaluation of sub-activity (AGD_PRE.1) sub-activity will
11172 satisfy this work unit if the evaluator still has confidence that the TOE being used for testing was
11173 installed properly and is in a known state. If this is not the case, then the evaluator should follow
11174 the developer's procedures to install and start up the TOE, using the supplied guidance only.
- 11175 If the evaluator has to perform the installation procedures because the TOE is in an unknown state,
11176 this work unit when successfully completed could satisfy work unit AGD_PRE.1-3.
- 11177 ISO/IEC 15408-3 ATE_IND.2.2C: *The developer shall provide an equivalent set of resources to those
11178 that were used in the developer's functional testing of the TSF.*

11179 **14.6.2.3.3 Work unit ATE_IND.2-3**

11180 The evaluator *shall examine* the set of resources provided by the developer to determine that they
11181 are equivalent to the set of resources used by the developer to functionally test the TSF.

11182 The set of resource used by the developer is documented in the developer test plan, as considered
11183 in the Functional tests (ATE_FUN) family. The resource set may include laboratory access and
11184 special test equipment, among others. Resources that are not identical to those used by the
11185 developer need to be equivalent in terms of any impact they may have on test results.

11186 **14.6.2.4 Action ATE_IND.2.2E**

11187 **14.6.2.4.1 Work unit ATE_IND.2-4**

11188 The evaluator *shall conduct* testing using a sample of tests found in the developer test plan and
11189 procedures.

11190 The overall aim of this work unit is to perform a sufficient number of the developer tests to confirm
11191 the validity of the developer's test results. The evaluator has to decide on the size of the sample,
11192 and the developer tests that will compose the sample (see A.2).

11193 All the developer tests can be traced back to specific interfaces. Therefore, the factors to consider
11194 in the selection of the tests to compose the sample are similar to those listed for subset selection in
11195 work-unit ATE_IND.2-6. Additionally, the evaluator may wish to employ a random sampling
11196 method to select developer tests to include in the sample.

11197 **14.6.2.4.2 Work unit ATE_IND.2-5**

11198 The evaluator *shall check* that all the actual test results are consistent with the expected test
11199 results.

11200 Inconsistencies between the developer's expected test results and actual test results will compel
11201 the evaluator to resolve the discrepancies. Inconsistencies encountered by the evaluator could be
11202 resolved by a valid explanation and resolution of the inconsistencies by the developer.

11203 If a satisfactory explanation or resolution can not be reached, the evaluator's confidence in the
11204 developer's test results may be lessened and it may be necessary for the evaluator to increase the
11205 sample size to the extent that the subset identified in work unit ATE_IND.2-4 is adequately tested:
11206 deficiencies with the developer's tests need to result in either corrective action to the TOE by the
11207 developer (e.g., if the inconsistency is caused by incorrect behaviour) or to the developer's tests
11208 (e.g., if the inconsistency is caused by an incorrect test), or in the production of new tests by the
11209 evaluator.

11210 **14.6.2.5 Action ATE_IND.2.3E**

11211 **14.6.2.5.1 Work unit ATE_IND.2-6**

11212 The evaluator *shall devise* a test subset.

11213 The evaluator selects a test subset and testing strategy that is appropriate for the TOE. One
11214 extreme testing strategy would be to have the test subset contain as many interfaces as possible
11215 tested with little rigour. Another testing strategy would be to have the test subset contain a few
11216 interfaces based on their perceived relevance and rigorously test these interfaces.

11217 Typically the testing approach taken by the evaluator should fall somewhere between these two
11218 extremes. The evaluator should exercise most of the interfaces using at least one test, but testing
11219 need not demonstrate exhaustive specification testing.

- 11220 The evaluator, when selecting the subset of the interfaces to be tested, should consider the
11221 following factors:
- 11222 a) The developer test evidence. The developer test evidence consists of: the test
11223 documentation, the available test coverage analysis, and the available depth of testing
11224 analysis. The developer test evidence will provide insight as to how the TSF has been
11225 exercised by the developer during testing. The evaluator applies this information when
11226 developing new tests to independently test the TOE. Specifically the evaluator should
11227 consider:
- 11228 1) augmentation of developer testing for interfaces. The evaluator may wish to perform
11229 more of the same type of tests by varying parameters to more rigorously test the interface.
- 11230 2) supplementation of developer testing strategy for interfaces. The evaluator may wish to
11231 vary the testing approach of a specific interface by testing it using another test strategy.
- 11232 b) The number of interfaces from which to draw upon for the test subset. Where the TSF
11233 includes only a small number of relatively simple interfaces, it may be practical to
11234 rigorously test all of them. In other cases this may not be cost-effective, and sampling is
11235 required.
- 11236 c) Maintaining a balance of evaluation activities. The evaluator effort expended on the test
11237 activity should be commensurate with that expended on any other evaluation activity.
- 11238 The evaluator selects the interfaces to compose the subset. This selection will depend on a number
11239 of factors, and consideration of these factors may also influence the choice of test subset size:
- 11240 a) Rigour of developer testing of the interfaces. Those interfaces that the evaluator
11241 determines require additional testing should be included in the test subset.
- 11242 b) Developer test results. If the results of developer tests cause the evaluator to doubt that
11243 an interface is not properly implemented, then the evaluator should include such
11244 interfaces in the test subset.
- 11245 c) Significance of interfaces. Those interfaces more significant than others should be
11246 included in the test subset. One major factor of "significance" is the security-relevance
11247 (SFR-enforcing interfaces would be more significant than SFR-supporting interfaces,
11248 which are more significant than SFR-non-interfering interfaces; see ISO/IEC 15408-3
11249 Subclause ADV_FSP). The other major factor of "significance" is the number of SFRs
11250 mapping to this interface (as determined when identifying the correspondence between
11251 levels of abstraction in ADV).
- 11252 d) Complexity of interfaces. Interfaces that require complex implementation may require
11253 complex tests that impose onerous requirements on the developer or evaluator, which
11254 may not be conducive to cost-effective evaluations. Conversely, they are a likely area to
11255 find errors and are good candidates for the subset. The evaluator will need to strike a
11256 balance between these considerations.
- 11257 e) Implicit testing. Testing some interfaces may often implicitly test other interfaces, and
11258 their inclusion in the subset may maximise the number of interfaces tested (albeit
11259 implicitly). Certain interfaces will typically be used to provide a variety of security
11260 functionality, and will tend to be the target of an effective testing approach.
- 11261 f) Types of interfaces (e.g. programmatic, command-line, protocol). The evaluator should
11262 consider including tests for all different types of interfaces that the TOE supports.
- 11263 g) Interfaces that give rise to features that are innovative or unusual. Where the TOE
11264 contains innovative or unusual features, which may feature strongly in marketing

- 11265 literature and guidance documents, the corresponding interfaces should be strong
11266 candidates for testing.
- 11267 This guidance articulates factors to consider during the selection process of an appropriate test
11268 subset, but these are by no means exhaustive.
- 11269 **14.6.2.5.2 Work unit ATE_IND.2-7**
- 11270 The evaluator *shall produce* test documentation for the test subset that is sufficiently detailed to
11271 enable the tests to be reproducible.
- 11272 With an understanding of the expected behaviour of the TSF, from the ST, the functional
11273 specification, and the TOE design description, the evaluator has to determine the most feasible way
11274 to test the interface. Specifically the evaluator considers:
- 11275 a) the approach that will be used, for instance, whether an external interface will be tested,
11276 or an internal interface using a test harness, or will an alternate test approach be
11277 employed (e.g. in exceptional circumstances, a code inspection);
- 11278 b) the interface(s) that will be used to test and observe responses;
- 11279 c) the initial conditions that will need to exist for the test (i.e. any particular objects or
11280 subjects that will need to exist and security attributes they will need to have);
- 11281 d) special test equipment that will be required to either stimulate an interface (e.g. packet
11282 generators) or make observations of an interface (e.g. network analysers).
- 11283 The evaluator may find it practical to test each interface using a series of test cases, where each test
11284 case will test a very specific aspect of expected behaviour of that interface.
- 11285 The evaluator's test documentation should specify the derivation of each test, tracing it back to the
11286 relevant interface(s).
- 11287 **14.6.2.5.3 Work unit ATE_IND.2-8**
- 11288 The evaluator *shall conduct* testing.
- 11289 The evaluator uses the test documentation developed as a basis for executing tests on the TOE. The
11290 test documentation is used as a basis for testing but this does not preclude the evaluator from
11291 performing additional ad hoc tests. The evaluator may devise new tests based on behaviour of the
11292 TOE discovered during testing. These new tests are recorded in the test documentation.
- 11293 **14.6.2.5.4 Work unit ATE_IND.2-9**
- 11294 The evaluator *shall record* the following information about the tests that compose the test subset:
- 11295 a) identification of the interface behaviour to be tested;
- 11296 b) instructions to connect and setup all required test equipment as required to conduct the
11297 test;
- 11298 c) instructions to establish all prerequisite test conditions;
- 11299 d) instructions to stimulate the interface;
- 11300 e) instructions for observing the interface;

- 11301 f) descriptions of all expected results and the necessary analysis to be performed on the
11302 observed behaviour for comparison against expected results;
- 11303 g) instructions to conclude the test and establish the necessary post-test state for the TOE;
- 11304 h) actual test results.
- 11305 The level of detail should be such that another evaluator could repeat the tests and obtain an
11306 equivalent result. While some specific details of the test results may be different (e.g. time and date
11307 fields in an audit record) the overall result should be identical.
- 11308 There may be instances when it is unnecessary to provide all the information presented in this
11309 work unit (e.g. the actual test results of a test may not require any analysis before a comparison
11310 between the expected results can be made). The determination to omit this information is left to
11311 the evaluator, as is the justification.
- 11312 **14.6.2.5.5 Work unit ATE_IND.2-10**
- 11313 The evaluator *shall check* that all actual test results are consistent with the expected test results.
- 11314 Any differences in the actual and expected test results may indicate that the TOE does not perform
11315 as specified or that the evaluator test documentation may be incorrect. Unexpected actual results
11316 may require corrective maintenance to the TOE or test documentation and perhaps require re-
11317 running of impacted tests and modifying the test sample size and composition. This determination
11318 is left to the evaluator, as is its justification.
- 11319 **14.6.2.5.6 Work unit ATE_IND.2-11**
- 11320 The evaluator *shall report* in the ETR the evaluator testing effort, outlining the testing approach,
11321 configuration, depth and results.
- 11322 The evaluator testing information reported in the ETR allows the evaluator to convey the overall
11323 testing approach and effort expended on the testing activity during the evaluation. The intent of
11324 providing this information is to give a meaningful overview of the testing effort. It is not intended
11325 that the information regarding testing in the ETR be an exact reproduction of specific test
11326 instructions or results of individual tests. The intention is to provide enough detail to allow other
11327 evaluators and evaluation authorities to gain some insight about the testing approach chosen,
11328 amount of evaluator testing performed, amount of developer tests performed, TOE test
11329 configurations, and the overall results of the testing activity.
- 11330 Information that would typically be found in the ETR subclause regarding the evaluator testing
11331 effort is:
- 11332 a) TOE test configurations. The particular configurations of the TOE that were tested.
- 11333 b) subset size chosen. The amount of interfaces that were tested during the evaluation and a
11334 justification for the size.
- 11335 c) selection criteria for the interfaces that compose the subset. Brief statements about the
11336 factors considered when selecting interfaces for inclusion in the subset.
- 11337 d) Interfaces tested. A brief listing of the interfaces that merited inclusion in the subset.
- 11338 e) developer tests performed. The amount of developer tests performed and a brief
11339 description of the criteria used to select the tests.
- 11340 f) verdict for the activity. The overall judgement on the results of testing during the
11341 evaluation.

11342 This list is by no means exhaustive and is only intended to provide some context as to the type of
 11343 information that should be present in the ETR concerning the testing the evaluator performed
 11344 during the evaluation.

11345 **14.6.3 Evaluation of sub-activity (ATE_IND.3)**

11346 There is no general guidance; the scheme should be consulted for guidance on this sub-activity.

11347 **14.7 Composite functional testing (ATE_COMP)**

11348 The composite-specific work units defined here are intended to be integrated as refinements to the
 11349 evaluation activities of the ATE class listed in the following table. The other activities of ATE class
 11350 do not require composite-specific work units.

CC assurance family	Evaluation activity	Evaluation work unit	Composite-specific work unit
ATE_COV	ATE_COV.1.1C	ATE_COV.1-1	ATE_COMP.1-1
ATE_FUN	ATE_FUN.1.2C	ATE_FUN.1-3	ATE_COMP.1-1

11351 NB: If the level of the assurance requirement chosen is higher than those identified in this table, the
 11352 composite-specific work unit is also applicable.

11353 **14.7.1 Evaluation of sub-activity (ATE_COMP.1)**

11354 **14.7.1.1 Objectives**

11355 The aim of this activity is to determine whether composite product as a whole exhibits the
 11356 properties necessary to satisfy the functional requirements of its Security Target.

11357 **14.7.1.2 Application notes**

11358 A composite product can be tested separately and integrated. Separate testing means that the
 11359 platform and the application are being tested independent of each other. A lot of tests of the
 11360 platform may have been performed within the scope of its accomplished evaluation. The
 11361 application may be tested on a simulator or an emulator, which represent a virtual machine.
 11362 Integration testing means that the composite product is being tested as it is: the application is
 11363 running on the platform.

11364 Behaviour of implementation of some SFRs can depend on properties of the underlying platform as
 11365 well as of the application (e.g. correctness of the measures of the composite product to withstand a
 11366 side channel attack or correctness of the implementation of tamper resistance against physical
 11367 attacks). In such a case the SFR implementation shall be tested on the final composite product, but
 11368 not on a simulator or an emulator.

11369 This activity focuses exclusively on testing of the composite product as a whole and represents
 11370 merely partial efforts within the general test approach being covered by the assurance ATE. These
 11371 integration tests shall be specified and performed, whereby the approach of the standard
 11372 assurance families of the class ATE shall be applied.

11373 A correct behaviour of the Platform-TSF being relevant for the Composite-ST (corresponding to the
 11374 group RP_SFR-SERV and RP-SFR-MECH in the work unit ADV_COMP.1-1 above),and- absence of
 11375 exploitable vulnerabilities (sufficient effectiveness) in the context of the Platform-ST are confirmed
 11376 by the valid Platform Certificate, cf. chapter 6 above.

- 11377 **14.7.1.3 Action ATE_COMP.1.1E**
- 11378 The evaluator shall confirm that the information provided meets all requirements for content and
11379 presentation of evidence.
- 11380 **14.7.1.3.1 Work unit ATE_COMP.1-1**
- 11381 The evaluator shall examine that the developer performed the integration tests for all SFRs having
11382 to be tested on the composite product as a whole.
- 11383 In order to perform this work unit the evaluator shall analyse, for each SFR, whether it directly
11384 depends on security properties of the platform and of the application. Then the evaluator shall
11385 verify that the integration tests performed by the developer cover at least all such SFRs.
- 11386 If the assurance package chosen does not contain the families ATE_FUN and ATE_COV (e.g. EAL1),
11387 this work unit is not applicable.
- 11388 The result of this work unit shall be integrated to the result of ATE_COV.1-1C/ ATE_COV.1-1 and
11389 ATE_FUN.1.2C/ ATE_FUN.1-3 (or the equivalent higher components if a higher assurance level is
11390 selected).
- 11391 **15 Class AVA: Vulnerability assessment**
- 11392 **15.1 Introduction**
- 11393 The purpose of the vulnerability assessment activity is to determine the exploitability of flaws or
11394 weaknesses in the TOE in the operational environment. This determination is based upon analysis
11395 of the evaluation evidence and a search of publicly available material by the evaluator and is
11396 supported by evaluator penetration testing.
- 11397 Vulnerability analysis (AVA_VAN)
- 11398 **15.1.1 Evaluation of sub-activity (AVA_VAN.1)**
- 11399 **15.1.1.1 Objectives**
- 11400 The objective of this sub-activity is to determine whether the TOE, in its operational environment,
11401 has easily identifiable exploitable vulnerabilities.
- 11402 **15.1.1.2 Input**
- 11403 The evaluation evidence for this sub-activity is:
- 11404 a) the ST;
- 11405 b) the guidance documentation;
- 11406 c) the TOE suitable for testing;
- 11407 d) information publicly available to support the identification of potential vulnerabilities.
- 11408 Other input for this sub-activity is:
- 11409 a) current information regarding potential vulnerabilities (e.g. from an evaluation authority).

11410 **15.1.1.3 Application notes**

11411 The evaluator should consider performing additional tests as a result of potential vulnerabilities
11412 encountered during the conduct of other parts of the evaluation.

11413 The use of the term guidance in this sub-activity refers to the operational guidance and the
11414 preparative guidance.

11415 Potential vulnerabilities may be in information that is publicly available, or not, and may require
11416 skill to exploit, or not. These two aspects are related, but are distinct. It should not be assumed that,
11417 simply because a potential vulnerability is identifiable from information that is publicly available, it
11418 can be easily exploited.

11419 **15.1.1.4 Action AVA_VAN.1.1E**

11420 ISO/IEC 15408-3 AVA_VAN.1.1C: *The TOE shall be suitable for testing.*

11421 **15.1.1.4.1 Work unit AVA_VAN.1-1**

11422 The evaluator *shall examine* the TOE to determine that the test configuration is consistent with
11423 the configuration under evaluation as specified in the ST.

11424 The TOE provided by the developer and identified in the test plan should have the same unique
11425 reference as established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST
11426 introduction.

11427 It is possible for the ST to specify more than one configuration for evaluation. The TOE may
11428 comprise a number of distinct hardware and software entities that need to be tested in accordance
11429 with the ST. The evaluator verifies that all test configurations are consistent with the ST.

11430 The evaluator should consider the security objectives for the operational environment described in
11431 the ST that may apply to the test environment and ensure they are met in the testing environment.
11432 There may be some objectives for the operational environment that do not apply to the test
11433 environment. For example, an objective about user clearances may not apply; however, an
11434 objective about a single point of connection to a network would apply.

11435 If any test resources are used (e.g. meters, analysers) it will be the evaluator's responsibility to
11436 ensure that these resources are calibrated correctly.

11437 **15.1.1.4.2 Work unit AVA_VAN.1-2**

11438 The evaluator *shall examine* the TOE to determine that it has been installed properly and is in a
11439 known state

11440 It is possible for the evaluator to determine the state of the TOE in a number of ways. For example,
11441 previous successful completion of the Evaluation of sub-activity (AGD_PRE.1) sub-activity will
11442 satisfy this work unit if the evaluator still has confidence that the TOE being used for testing was
11443 installed properly and is in a known state. If this is not the case, then the evaluator should follow
11444 the developer's procedures to install and start up the TOE, using the supplied guidance only.

11445 If the evaluator has to perform the installation procedures because the TOE is in an unknown state,
11446 this work unit when successfully completed could satisfy work unit AGD_PRE.1-3.

11447 **15.1.1.5 Action AVA_VAN.1.2E**11448 **15.1.1.5.1 Work unit AVA_VAN.1-3**

11449 The evaluator *shall examine* sources of information publicly available to identify potential
11450 vulnerabilities in the TOE.

11451 The evaluator examines the sources of information publicly available to support the identification
11452 of possible potential vulnerabilities in the TOE. There are many sources of publicly available
11453 information, which should be considered, e.g. mailing lists and security forums on the world wide
11454 web that report known vulnerabilities in specified technologies.

11455 The evaluator should not constrain their consideration of publicly available information to the
11456 above, but should consider any other relevant information available.

11457 While examining the evidence provided the evaluator will use the information in the public domain
11458 to further search for potential vulnerabilities. Where the evaluators have identified areas of
11459 concern, the evaluator should consider information publicly available that relate to those areas of
11460 concern.

11461 The availability of information that may be readily available to an attacker that helps to identify
11462 and facilitate attacks effectively operates to substantially enhance the attack potential of a given
11463 attacker. The accessibility of vulnerability information and sophisticated attack tools on the
11464 Internet makes it more likely that this information will be used in attempts to identify potential
11465 vulnerabilities in the TOE and exploit them. Modern search tools make such information easily
11466 available to the evaluator, and the determination of resistance to published potential
11467 vulnerabilities and well known generic attacks can be achieved in a cost-effective manner.

11468 The search of the information publicly available should be focused on those sources that refer
11469 specifically to the product from which the TOE is derived. The extensiveness of this search should
11470 consider the following factors: TOE type, evaluator experience in this TOE type, expected attack
11471 potential and the level of ADV evidence available.

11472 The identification process is iterative, where the identification of one potential vulnerability may
11473 lead to identifying another area of concern that requires further investigation.

11474 The evaluator will report what actions were taken to identify potential vulnerabilities in the
11475 information publicly available. However, in this type of search, the evaluator may not be able to
11476 describe the steps in identifying potential vulnerabilities before the outset of the examination, as
11477 the approach may evolve as a result of findings during the search.

11478 The evaluator will report the evidence examined in completing the search for potential
11479 vulnerabilities.

11480 **15.1.1.5.2 Work unit AVA_VAN.1-4**

11481 The evaluator *shall record* in the ETR the identified potential vulnerabilities that are candidates
11482 for testing and applicable to the TOE in its operational environment.

11483 It may be identified that no further consideration of the potential vulnerability is required if for
11484 example the evaluator identifies that measures in the operational environment, either IT or non-IT,
11485 prevent exploitation of the potential vulnerability in that operational environment. For instance,
11486 restricting physical access to the TOE to authorised users only may effectively render a potential
11487 vulnerability to tampering unexploitable.

11488 The evaluator records any reasons for exclusion of potential vulnerabilities from further
11489 consideration if the evaluator determines that the potential vulnerability is not applicable in the

11490 operational environment. Otherwise the evaluator records the potential vulnerability for further
11491 consideration.

11492 A list of potential vulnerabilities applicable to the TOE in its operational environment, which can be
11493 used as an input into penetration testing activities, shall be reported in the ETR by the evaluators.

11494 **15.1.1.6 Action AVA_VAN.1.3E**

11495 **15.1.1.6.1 Work unit AVA_VAN.1-5**

11496 The evaluator *shall devise* penetration tests, based on the independent search for potential
11497 vulnerabilities.

11498 The evaluator prepares for penetration testing as necessary to determine the susceptibility of the
11499 TOE, in its operational environment, to the potential vulnerabilities identified during the search of
11500 the sources of information publicly available. Any current information provided to the evaluator by
11501 a third party (e.g. evaluation authority) regarding known potential vulnerabilities will be
11502 considered by the evaluator, together with any encountered potential vulnerabilities resulting
11503 from the performance of other evaluation activities.

11504 The evaluator will probably find it practical to carry out penetration test using a series of test cases,
11505 where each test case will test for a specific potential vulnerability.

11506 The evaluator is not expected to test for potential vulnerabilities (including those in the public
11507 domain) beyond those which required a Basic attack potential. In some cases, however, it will be
11508 necessary to carry out a test before the exploitability can be determined. Where, as a result of
11509 evaluation expertise, the evaluator discovers a potential vulnerability that is beyond Basic attack
11510 potential, this is reported in the ETR as a residual vulnerability.

11511 **15.1.1.6.2 Work unit AVA_VAN.1-6**

11512 The evaluator *shall produce* penetration test documentation for the tests based on the list of
11513 potential vulnerabilities in sufficient detail to enable the tests to be repeatable. The test
11514 documentation shall include:

11515 a) identification of the potential vulnerability the TOE is being tested for;

11516 b) instructions to connect and setup all required test equipment as required to conduct the
11517 penetration test;

11518 c) instructions to establish all penetration test prerequisite initial conditions;

11519 d) instructions to stimulate the TSF;

11520 e) instructions for observing the behaviour of the TSF;

11521 f) descriptions of all expected results and the necessary analysis to be performed on the
11522 observed behaviour for comparison against expected results;

11523 g) instructions to conclude the test and establish the necessary post-test state for the TOE.

11524 The evaluator prepares for penetration testing based on the list of potential vulnerabilities
11525 identified during the search of the public domain.

11526 The evaluator is not expected to determine the exploitability for potential vulnerabilities beyond
11527 those for which a Basic attack potential is required to effect an attack. However, as a result of
11528 evaluation expertise, the evaluator may discover a potential vulnerability that is exploitable only

11529 by an attacker with greater than Basic attack potential. Such vulnerabilities are to be reported in
11530 the ETR as residual vulnerabilities.

11531 With an understanding of the potential vulnerability, the evaluator determines the most feasible
11532 way to test for the TOE's susceptibility. Specifically the evaluator considers:

11533 a) the TSFI or other TOE interface that will be used to stimulate the TSF and observe
11534 responses;

11535 b) initial conditions that will need to exist for the test (i.e. any particular objects or subjects
11536 that will need to exist and security attributes they will need to have);

11537 c) special test equipment that will be required to either stimulate a TSFI or make
11538 observations of a TSFI (although it is unlikely that specialist equipment would be
11539 required to exploit a potential vulnerability assuming a Basic attack potential);

11540 d) whether theoretical analysis should replace physical testing, particularly relevant where
11541 the results of an initial test can be extrapolated to demonstrate that repeated attempts of
11542 an attack are likely to succeed after a given number of attempts.

11543 The evaluator will probably find it practical to carry out penetration testing using a series of test
11544 cases, where each test case will test for a specific potential vulnerability.

11545 The intent of specifying this level of detail in the test documentation is to allow another evaluator
11546 to repeat the tests and obtain an equivalent result.

11547 **15.1.1.6.3 Work unit AVA_VAN.1-7**

11548 The evaluator *shall conduct* penetration testing.

11549 The evaluator uses the penetration test documentation resulting from work unit AVA_VAN.1-5 as a
11550 basis for executing penetration tests on the TOE, but this does not preclude the evaluator from
11551 performing additional ad hoc penetration tests. If required, the evaluator may devise ad hoc tests
11552 as a result of information learnt during penetration testing that, if performed by the evaluator, are
11553 to be recorded in the penetration test documentation. Such tests may be required to follow up
11554 unexpected results or observations, or to investigate potential vulnerabilities suggested to the
11555 evaluator during the pre-planned testing.

11556 The evaluator is not expected to test for potential vulnerabilities (including those in the public
11557 domain) beyond those which required a Basic attack potential. In some cases, however, it will be
11558 necessary to carry out a test before the exploitability can be determined. Where, as a result of
11559 evaluation expertise, the evaluator discovers a potential vulnerability that is beyond Basic attack
11560 potential, this is reported in the ETR as a residual vulnerability.

11561 **15.1.1.6.4 Work unit AVA_VAN.1-8**

11562 The evaluator *shall record* the actual results of the penetration tests.

11563 While some specific details of the actual test results may be different from those expected (e.g. time
11564 and date fields in an audit record) the overall result should be identical. Any unexpected test
11565 results should be investigated. The impact on the evaluation should be stated and justified.

11566 **15.1.1.6.5 Work unit AVA_VAN.1-9**

11567 The evaluator *shall report* in the ETR the evaluator penetration testing effort, outlining the testing
11568 approach, configuration, depth and results.

11569 The penetration testing information reported in the ETR allows the evaluator to convey the overall
11570 penetration testing approach and effort expended on this sub-activity. The intent of providing this
11571 information is to give a meaningful overview of the evaluator's penetration testing effort. It is not
11572 intended that the information regarding penetration testing in the ETR be an exact reproduction of
11573 specific test steps or results of individual penetration tests. The intention is to provide enough
11574 detail to allow other evaluators and evaluation authorities to gain some insight about the
11575 penetration testing approach chosen, amount of penetration testing performed, TOE test
11576 configurations, and the overall results of the penetration testing activity.

11577 Information that would typically be found in the ETR subclause regarding evaluator penetration
11578 testing efforts is:

11579 a) TOE test configurations. The particular configurations of the TOE that were penetration
11580 tested;

11581 b) TSFI penetration tested. A brief listing of the TSFI and other TOE interfaces that were the
11582 focus of the penetration testing;

11583 c) verdict for the sub-activity. The overall judgement on the results of penetration testing.

11584 This list is by no means exhaustive and is only intended to provide some context as to the type of
11585 information that should be present in the ETR concerning the penetration testing the evaluator
11586 performed during the evaluation.

11587 **15.1.1.6.6 Work unit AVA_VAN.1-10**

11588 The evaluator *shall examine* the results of all penetration testing to determine that the TOE, in its
11589 operational environment, is resistant to an attacker possessing a Basic attack potential.

11590 If the results reveal that the TOE, in its operational environment, has vulnerabilities exploitable by
11591 an attacker possessing less than Enhanced-Basic attack potential, then this evaluator action fails.

11592 The guidance in B.4 should be used to determine the attack potential required to exploit a
11593 particular vulnerability and whether it can therefore be exploited in the intended environment. It
11594 may not be necessary for the attack potential to be calculated in every instance, only if there is
11595 some doubt as to whether or not the vulnerability can be exploited by an attacker possessing an
11596 attack potential less than Enhanced-Basic.

11597 **15.1.1.6.7 Work unit AVA_VAN.1-11**

11598 The evaluator *shall report* in the ETR all exploitable vulnerabilities and residual vulnerabilities,
11599 detailing for each:

11600 a) its source (e.g. evaluation methodology activity being undertaken when it was conceived,
11601 known to the evaluator, read in a publication);

11602 b) the SFR(s) not met;

11603 c) a description;

11604 d) whether it is exploitable in its operational environment or not (i.e. exploitable or
11605 residual).

11606 e) the amount of time, level of expertise, level of knowledge of the TOE, level of opportunity
11607 and the equipment required to perform the identified vulnerabilities, and the
11608 corresponding values using the tables B.2 and B.3 of Annex B.4.

11609 **15.1.2 Evaluation of sub-activity (AVA_VAN.2)**11610 **15.1.2.1 Objectives**

11611 The objective of this sub-activity is to determine whether the TOE, in its operational environment,
11612 has vulnerabilities exploitable by attackers possessing Basic attack potential.

11613 **15.1.2.2 Input**

11614 The evaluation evidence for this sub-activity is:

11615 a) the ST;

11616 b) the functional specification;

11617 c) the TOE design;

11618 d) the security architecture description;

11619 e) the guidance documentation;

11620 f) the TOE suitable for testing;

11621 g) information publicly available to support the identification of possible potential
11622 vulnerabilities.

11623 The remaining implicit evaluation evidence for this sub-activity depends on the components that
11624 have been included in the assurance package. The evidence provided for each component is to be
11625 used as input in this sub-activity.

11626 Other input for this sub-activity is:

11627 a) current information regarding public domain potential vulnerabilities and attacks (e.g.
11628 from an evaluation authority).

11629 **15.1.2.3 Application notes**

11630 The evaluator should consider performing additional tests as a result of potential vulnerabilities
11631 encountered during other parts of the evaluation.

11632 **15.1.2.4 Action AVA_VAN.2.1E**

11633 ISO/IEC 15408-3 AVA_VAN.2.1C: *The TOE shall be suitable for testing.*

11634 **15.1.2.4.1 Work unit AVA_VAN.2-1**

11635 The evaluator **shall examine** the TOE to determine that the test configuration is consistent with
11636 the configuration under evaluation as specified in the ST.

11637 The TOE provided by the developer and identified in the test plan should have the same unique
11638 reference as established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST
11639 introduction.

11640 It is possible for the ST to specify more than one configuration for evaluation. The TOE may
11641 comprise a number of distinct hardware and software entities that need to be tested in accordance
11642 with the ST. The evaluator verifies that all test configurations are consistent with the ST.

11643 The evaluator should consider the security objectives for the operational environment described in
11644 the ST that may apply to the test environment and ensure they are met in the testing environment.
11645 There may be some objectives for the operational environment that do not apply to the test
11646 environment. For example, an objective about user clearances may not apply; however, an
11647 objective about a single point of connection to a network would apply.

11648 If any test resources are used (e.g. meters, analysers) it will be the evaluator's responsibility to
11649 ensure that these resources are calibrated correctly.

11650 **15.1.2.4.2 Work unit AVA_VAN.2-2**

11651 The evaluator *shall examine* the TOE to determine that it has been installed properly and is in a
11652 known state

11653 It is possible for the evaluator to determine the state of the TOE in a number of ways. For example,
11654 previous successful completion of the Evaluation of sub-activity (AGD_PRE.1) sub-activity will
11655 satisfy this work unit if the evaluator still has confidence that the TOE being used for testing was
11656 installed properly and is in a known state. If this is not the case, then the evaluator should follow
11657 the developer's procedures to install and start up the TOE, using the supplied guidance only.

11658 If the evaluator has to perform the installation procedures because the TOE is in an unknown state,
11659 this work unit when successfully completed could satisfy work unit AGD_PRE.1-3.

11660 **15.1.2.5 Action AVA_VAN.2.2E**

11661 **15.1.2.5.1 Work unit AVA_VAN.2-3**

11662 The evaluator *shall examine* sources of information publicly available to identify potential
11663 vulnerabilities in the TOE.

11664 The evaluator examines the sources of information publicly available to support the identification
11665 of possible potential vulnerabilities in the TOE. There are many sources of publicly available
11666 information which the evaluator should consider using items such as those available on the world
11667 wide web, including:

11668 a) specialist publications (magazines, books);

11669 b) research papers.

11670 The evaluator should not constrain their consideration of publicly available information to the
11671 above, but should consider any other relevant information available.

11672 While examining the evidence provided the evaluator will use the information in the public domain
11673 to further search for potential vulnerabilities. Where the evaluators have identified areas of
11674 concern, the evaluator should consider information publicly available that relate to those areas of
11675 concern.

11676 The availability of information that may be readily available to an attacker that helps to identify
11677 and facilitate attacks may substantially enhance the attack potential of a given attacker. The
11678 accessibility of vulnerability information and sophisticated attack tools on the Internet makes it
11679 more likely that this information will be used in attempts to identify potential vulnerabilities in the
11680 TOE and exploit them. Modern search tools make such information easily available to the evaluator,
11681 and the determination of resistance to published potential vulnerabilities and well known generic
11682 attacks can be achieved in a cost-effective manner.

11683 The search of the information publicly available should be focused on those sources that refer
11684 specifically to the product from which the TOE is derived. The extensiveness of this search should

- 11685 consider the following factors: TOE type, evaluator experience in this TOE type, expected attack
11686 potential and the level of ADV evidence available.
- 11687 The identification process is iterative, where the identification of one potential vulnerability may
11688 lead to identifying another area of concern that requires further investigation.
- 11689 The evaluator will report what actions were taken to identify potential vulnerabilities in the
11690 evidence. However, in this type of search, the evaluator may not be able to describe the steps in
11691 identifying potential vulnerabilities before the outset of the examination, as the approach may
11692 evolve as a result of findings during the search.
- 11693 The evaluator will report the evidence examined in completing the search for potential
11694 vulnerabilities. This selection of evidence may be derived from those areas of concern identified by
11695 the evaluator, linked to the evidence the attacker is assumed to be able to obtain, or according to
11696 another rationale provided by the evaluator.
- 11697 **15.1.2.6 Action AVA_VAN.2.3E**
- 11698 **15.1.2.6.1 Work unit AVA_VAN.2-4**
- 11699 The evaluator *shall conduct* a search of ST, guidance documentation, functional specification, TOE
11700 design and security architecture description evidence to identify possible potential vulnerabilities
11701 in the TOE.
- 11702 A search of the evidence should be completed whereby specifications and documentation for the
11703 TOE are analysed and then potential vulnerabilities in the TOE are hypothesised, or speculated.
11704 The list of hypothesised potential vulnerabilities is then prioritised on the basis of the estimated
11705 probability that a potential vulnerability exists and, assuming an exploitable vulnerability does
11706 exist the attack potential required to exploit it, and on the extent of control or compromise it would
11707 provide. The prioritised list of potential vulnerabilities is used to direct penetration testing against
11708 the TOE.
- 11709 The security architecture description provides the developer vulnerability analysis, as it
11710 documents how the TSF protects itself from interference from untrusted subjects and prevents the
11711 bypass of security enforcement functionality. Therefore, the evaluator should use this description
11712 of the protection of the TSF as a basis for the search for possible ways to undermine the TSF.
- 11713 Subject to the SFRs the TOE is to meet in the operational environment, the evaluator's independent
11714 vulnerability analysis should consider generic potential vulnerabilities under each of the following
11715 headings:
- 11716 a) generic potential vulnerabilities relevant for the type of TOE being evaluated, as may be
11717 supplied by the evaluation authority;
- 11718 b) bypassing;
- 11719 c) tampering;
- 11720 d) direct attacks;
- 11721 e) monitoring;
- 11722 f) misuse.
- 11723 Items b) - f) are explained in greater detail in Annex B.

11724 The security architecture description should be considered in light of each of the above generic
11725 potential vulnerabilities. Each potential vulnerability should be considered to search for possible
11726 ways in which to defeat the TSF protection and undermine the TSF.

11727 **15.1.2.6.2 Work unit AVA_VAN.2-5**

11728 The evaluator *shall record* in the ETR the identified potential vulnerabilities that are candidates
11729 for testing and applicable to the TOE in its operational environment.

11730 It may be identified that no further consideration of the potential vulnerability is required if for
11731 example the evaluator identifies that measures in the operational environment, either IT or non-IT,
11732 prevent exploitation of the potential vulnerability in that operational environment. For instance,
11733 restricting physical access to the TOE to authorised users only may effectively render a potential
11734 vulnerability to tampering unexploitable.

11735 The evaluator records any reasons for exclusion of potential vulnerabilities from further
11736 consideration if the evaluator determines that the potential vulnerability is not applicable in the
11737 operational environment. Otherwise the evaluator records the potential vulnerability for further
11738 consideration.

11739 A list of potential vulnerabilities applicable to the TOE in its operational environment, which can be
11740 used as an input into penetration testing activities, shall be reported in the ETR by the evaluators.

11741 **15.1.2.7 Action AVA_VAN.2.4E**

11742 **15.1.2.7.1 Work unit AVA_VAN.2-6**

11743 The evaluator *shall devise* penetration tests, based on the independent search for potential
11744 vulnerabilities.

11745 The evaluator prepares for penetration testing as necessary to determine the susceptibility of the
11746 TOE, in its operational environment, to the potential vulnerabilities identified during the search of
11747 the sources of information publicly available. Any current information provided to the evaluator by
11748 a third party (e.g. evaluation authority) regarding known potential vulnerabilities will be
11749 considered by the evaluator, together with any encountered potential vulnerabilities resulting
11750 from the performance of other evaluation activities.

11751 The evaluator is reminded that, as for considering the security architecture description in the
11752 search for vulnerabilities (as detailed in AVA_VAN.2-4), testing should be performed to confirm the
11753 architectural properties. This is likely to require negative tests attempting to disprove the
11754 properties of the security architecture. In developing the strategy for penetration testing, the
11755 evaluator will ensure that each of the major characteristics of the security architecture description
11756 are tested, either in functional testing (as considered in 14) or evaluator penetration testing.

11757 The evaluator will probably find it practical to carry out penetration test using a series of test cases,
11758 where each test case will test for a specific potential vulnerability.

11759 The evaluator is not expected to test for potential vulnerabilities (including those in the public
11760 domain) beyond those which required a Basic attack potential. In some cases, however, it will be
11761 necessary to carry out a test before the exploitability can be determined. Where, as a result of
11762 evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond Basic
11763 attack potential, this is reported in the ETR as a residual vulnerability.

11764 Guidance on determining the necessary attack potential to exploit a potential vulnerability can be
11765 found in Annex B.4.

11766 Potential vulnerabilities hypothesised as exploitable only by attackers possessing Enhanced-Basic,
11767 Moderate or High attack potential do not result in a failure of this evaluator action. Where analysis

11768 supports the hypothesis, these need not be considered further as an input to penetration testing.
11769 However, such vulnerabilities are reported in the ETR as residual vulnerabilities.

11770 Potential vulnerabilities hypothesised as exploitable by an attacker possessing a Basic attack
11771 potential and resulting in a violation of the security objectives should be the highest priority
11772 potential vulnerabilities comprising the list used to direct penetration testing against the TOE.

11773 **15.1.2.7.2 Work unit AVA_VAN.2-7**

11774 The evaluator **shall produce** penetration test documentation for the tests based on the list of
11775 potential vulnerabilities in sufficient detail to enable the tests to be repeatable. The test
11776 documentation shall include:

- 11777 a) identification of the potential vulnerability the TOE is being tested for;
- 11778 b) instructions to connect and setup all required test equipment as required to conduct the
11779 penetration test;
- 11780 c) instructions to establish all penetration test prerequisite initial conditions;
- 11781 d) instructions to stimulate the TSF;
- 11782 e) instructions for observing the behaviour of the TSF;
- 11783 f) descriptions of all expected results and the necessary analysis to be performed on the
11784 observed behaviour for comparison against expected results;
- 11785 g) instructions to conclude the test and establish the necessary post-test state for the TOE.

11786 The evaluator prepares for penetration testing based on the list of potential vulnerabilities
11787 identified during the search of the public domain and the analysis of the evaluation evidence.

11788 The evaluator is not expected to determine the exploitability for potential vulnerabilities beyond
11789 those for which a Basic attack potential is required to effect an attack. However, as a result of
11790 evaluation expertise, the evaluator may discover a potential vulnerability that is exploitable only
11791 by an attacker with greater than Basic attack potential. Such vulnerabilities are to be reported in
11792 the ETR as residual vulnerabilities.

11793 With an understanding of the potential vulnerability, the evaluator determines the most feasible
11794 way to test for the TOE's susceptibility. Specifically the evaluator considers:

- 11795 a) the TSFI or other TOE interface that will be used to stimulate the TSF and observe
11796 responses (It is possible that the evaluator will need to use an interface to the TOE other
11797 than the TSFI to demonstrate properties of the TSF such as those described in the
11798 security architecture description (as required by ADV_ARC). It should be noted, that
11799 although these TOE interfaces provide a means of testing the TSF properties, they are not
11800 the subject of the test.);
- 11801 b) initial conditions that will need to exist for the test (i.e. any particular objects or subjects
11802 that will need to exist and security attributes they will need to have);
- 11803 c) special test equipment that will be required to either stimulate a TSFI or make
11804 observations of a TSFI (although it is unlikely that specialist equipment would be
11805 required to exploit a potential vulnerability assuming a Basic attack potential);
- 11806 d) whether theoretical analysis should replace physical testing, particularly relevant where
11807 the results of an initial test can be extrapolated to demonstrate that repeated attempts of
11808 an attack are likely to succeed after a given number of attempts.

11809 The evaluator will probably find it practical to carry out penetration testing using a series of test
11810 cases, where each test case will test for a specific potential vulnerability.

11811 The intent of specifying this level of detail in the test documentation is to allow another evaluator
11812 to repeat the tests and obtain an equivalent result.

11813 **15.1.2.7.3 Work unit AVA_VAN.2-8**

11814 The evaluator *shall conduct* penetration testing.

11815 The evaluator uses the penetration test documentation resulting from work unit AVA_VAN.2-6 as a
11816 basis for executing penetration tests on the TOE, but this does not preclude the evaluator from
11817 performing additional ad hoc penetration tests. If required, the evaluator may devise ad hoc tests
11818 as a result of information learnt during penetration testing that, if performed by the evaluator, are
11819 to be recorded in the penetration test documentation. Such tests may be required to follow up
11820 unexpected results or observations, or to investigate potential vulnerabilities suggested to the
11821 evaluator during the pre-planned testing.

11822 Should penetration testing show that a hypothesised potential vulnerability does not exist, then the
11823 evaluator should determine whether or not the evaluator's own analysis was incorrect, or if
11824 evaluation deliverables are incorrect or incomplete.

11825 The evaluator is not expected to test for potential vulnerabilities (including those in the public
11826 domain) beyond those which required a Basic attack potential. In some cases, however, it will be
11827 necessary to carry out a test before the exploitability can be determined. Where, as a result of
11828 evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond basic
11829 attack potential, this is reported in the ETR as a residual vulnerability.

11830 **15.1.2.7.4 Work unit AVA_VAN.2-9**

11831 The evaluator *shall record* the actual results of the penetration tests.

11832 While some specific details of the actual test results may be different from those expected (e.g. time
11833 and date fields in an audit record) the overall result should be identical. Any unexpected test
11834 results should be investigated. The impact on the evaluation should be stated and justified.

11835 **15.1.2.7.5 Work unit AVA_VAN.2-10**

11836 The evaluator *shall report* in the ETR the evaluator penetration testing effort, outlining the testing
11837 approach, configuration, depth and results.

11838 The penetration testing information reported in the ETR allows the evaluator to convey the overall
11839 penetration testing approach and effort expended on this sub-activity. The intent of providing this
11840 information is to give a meaningful overview of the evaluator's penetration testing effort. It is not
11841 intended that the information regarding penetration testing in the ETR be an exact reproduction of
11842 specific test steps or results of individual penetration tests. The intention is to provide enough
11843 detail to allow other evaluators and evaluation authorities to gain some insight about the
11844 penetration testing approach chosen, amount of penetration testing performed, TOE test
11845 configurations, and the overall results of the penetration testing activity.

11846 Information that would typically be found in the ETR subclause regarding evaluator penetration
11847 testing efforts is:

11848 a) TOE test configurations. The particular configurations of the TOE that were penetration
11849 tested;

11850 b) TSFI penetration tested. A brief listing of the TSFI and other TOE interfaces that were the
11851 focus of the penetration testing;

- 11852 c) Verdict for the sub-activity. The overall judgement on the results of penetration testing.
- 11853 This list is by no means exhaustive and is only intended to provide some context as to the type of
11854 information that should be present in the ETR concerning the penetration testing the evaluator
11855 performed during the evaluation.
- 11856 **15.1.2.7.6 Work unit AVA_VAN.2-11**
- 11857 The evaluator *shall examine* the results of all penetration testing to determine that the TOE, in its
11858 operational environment, is resistant to an attacker possessing a Basic attack potential.
- 11859 If the results reveal that the TOE, in its operational environment, has vulnerabilities exploitable by
11860 an attacker possessing less than an Enhanced-Basic attack potential, then this evaluator action fails.
- 11861 The guidance in B.4 should be used to determine the attack potential required to exploit a
11862 particular vulnerability and whether it can therefore be exploited in the intended environment. It
11863 may not be necessary for the attack potential to be calculated in every instance, only if there is
11864 some doubt as to whether or not the vulnerability can be exploited by an attacker possessing an
11865 attack potential less than Enhanced-Basic.
- 11866 **15.1.2.7.7 Work unit AVA_VAN.2-12**
- 11867 The evaluator *shall report* in the ETR all exploitable vulnerabilities and residual vulnerabilities,
11868 detailing for each:
- 11869 a) its source (e.g. evaluation methodology activity being undertaken when it was conceived,
11870 known to the evaluator, read in a publication);
- 11871 b) the SFR(s) not met;
- 11872 c) a description;
- 11873 d) whether it is exploitable in its operational environment or not (i.e. exploitable or
11874 residual).
- 11875 e) the amount of time, level of expertise, level of knowledge of the TOE, level of opportunity
11876 and the equipment required to perform the identified vulnerabilities, and the
11877 corresponding values using the tables B.2 and B.3 of Annex B.4.
- 11878 **15.1.3 Evaluation of sub-activity (AVA_VAN.3)**
- 11879 **15.1.3.1 Objectives**
- 11880 The objective of this sub-activity is to determine whether the TOE, in its operational environment,
11881 has vulnerabilities exploitable by attackers possessing Enhanced-Basic attack potential.
- 11882 **15.1.3.2 Input**
- 11883 The evaluation evidence for this sub-activity is:
- 11884 a) the ST;
- 11885 b) the functional specification;
- 11886 c) the TOE design;
- 11887 d) the security architecture description;

- 11888 e) the implementation subset selected;
- 11889 f) the guidance documentation;
- 11890 g) the TOE suitable for testing;
- 11891 h) information publicly available to support the identification of possible potential
11892 vulnerabilities;
- 11893 i) the results of the testing of the basic design.

11894 The remaining implicit evaluation evidence for this sub-activity depends on the components that
11895 have been included in the assurance package. The evidence provided for each component is to be
11896 used as input in this sub-activity.

11897 Other input for this sub-activity is:

- 11898 a) current information regarding public domain potential vulnerabilities and attacks (e.g.
11899 from an evaluation authority).

11900 **15.1.3.3 Application notes**

11901 During the conduct of evaluation activities the evaluator may also identify areas of concern. These
11902 are specific portions of the TOE evidence that the evaluator has some reservation about, although
11903 the evidence meets the requirements for the activity with which the evidence is associated. For
11904 example, a particular interface specification looks particularly complex, and therefore may be
11905 prone to error either in the development of the TOE or in the operation of the TOE. There is no
11906 potential vulnerability apparent at this stage, further investigation is required. This is beyond the
11907 bounds of encountered, as further investigation is required.

11908 The focused approach to the identification of potential vulnerabilities is an analysis of the evidence
11909 with the aim of identifying any potential vulnerabilities evident through the contained information.
11910 It is an unstructured analysis, as the approach is not predetermined. Further guidance on focused
11911 vulnerability analysis can be found in Annex B.2.2.2.2.

11912 **15.1.3.4 Action AVA_VAN.3.1E**

11913 ISO/IEC 15408-3 AVA_VAN.3.1C: *The TOE shall be suitable for testing.*

11914 **15.1.3.4.1 Work unit AVA_VAN.3-1**

11915 The evaluator *shall examine* the TOE to determine that the test configuration is consistent with
11916 the configuration under evaluation as specified in the ST.

11917 The TOE provided by the developer and identified in the test plan should have the same unique
11918 reference as established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST
11919 introduction.

11920 It is possible for the ST to specify more than one configuration for evaluation. The TOE may
11921 comprise a number of distinct hardware and software entities that need to be tested in accordance
11922 with the ST. The evaluator verifies that all test configurations are consistent with the ST.

11923 The evaluator should consider the security objectives for the operational environment described in
11924 the ST that may apply to the test environment and ensure they are met in the testing environment.
11925 There may be some objectives for the operational environment that do not apply to the test
11926 environment. For example, an objective about user clearances may not apply; however, an
11927 objective about a single point of connection to a network would apply.

11928 If any test resources are used (e.g. meters, analysers) it will be the evaluator's responsibility to
11929 ensure that these resources are calibrated correctly.

11930 **15.1.3.4.2 Work unit AVA_VAN.3-2**

11931 The evaluator *shall examine* the TOE to determine that it has been installed properly and is in a
11932 known state

11933 It is possible for the evaluator to determine the state of the TOE in a number of ways. For example,
11934 previous successful completion of the Evaluation of sub-activity (AGD_PRE.1) sub-activity will
11935 satisfy this work unit if the evaluator still has confidence that the TOE being used for testing was
11936 installed properly and is in a known state. If this is not the case, then the evaluator should follow
11937 the developer's procedures to install and start up the TOE, using the supplied guidance only.

11938 If the evaluator has to perform the installation procedures because the TOE is in an unknown state,
11939 this work unit when successfully completed could satisfy work unit AGD_PRE.1-3.

11940 **15.1.3.5 Action AVA_VAN.3.2E**

11941 **15.1.3.5.1 Work unit AVA_VAN.3-3**

11942 The evaluator *shall examine* sources of information publicly available to identify potential
11943 vulnerabilities in the TOE.

11944 The evaluator examines the sources of information publicly available to support the identification
11945 of possible potential vulnerabilities in the TOE. There are many sources of publicly available
11946 information which the evaluator should consider using items such as those available on the world
11947 wide web, including:

11948 a) specialist publications (magazines, books);

11949 b) research papers;

11950 c) conference proceedings.

11951 The evaluator should not constrain their consideration of publicly available information to the
11952 above, but should consider any other relevant information available.

11953 While examining the evidence provided the evaluator will use the information in the public domain
11954 to further search for potential vulnerabilities. Where the evaluators have identified areas of
11955 concern, the evaluator should consider information publicly available that relate to those areas of
11956 concern.

11957 The availability of information that may be readily available to an attacker that helps to identify
11958 and facilitate attacks may substantially enhance the attack potential of a given attacker. The
11959 accessibility of vulnerability information and sophisticated attack tools on the Internet makes it
11960 more likely that this information will be used in attempts to identify potential vulnerabilities in the
11961 TOE and exploit them. Modern search tools make such information easily available to the evaluator,
11962 and the determination of resistance to published potential vulnerabilities and well known generic
11963 attacks can be achieved in a cost-effective manner.

11964 The search of the information publicly available should be focused on those sources that refer to
11965 the technologies used in the development of the product from which the TOE is derived. The
11966 extensiveness of this search should consider the following factors: TOE type, evaluator experience
11967 in this TOE type, expected attack potential and the level of ADV evidence available.

11968 The identification process is iterative, where the identification of one potential vulnerability may
11969 lead to identifying another area of concern that requires further investigation.

11970 The evaluator will report what actions were taken to identify potential vulnerabilities in the
11971 evidence. However, in this type of search, the evaluator may not be able to describe the steps in
11972 identifying potential vulnerabilities before the outset of the examination, as the approach may
11973 evolve as a result of findings during the search.

11974 The evaluator will report the evidence examined in completing the search for potential
11975 vulnerabilities. This selection of evidence may be derived from those areas of concern identified by
11976 the evaluator, linked to the evidence the attacker is assumed to be able to obtain, or according to
11977 another rationale provided by the evaluator.

11978 **15.1.3.6 Action AVA_VAN.3.3E**

11979 **15.1.3.6.1 Work unit AVA_VAN.3-4**

11980 The evaluator *shall conduct* a focused search of ST, guidance documentation, functional
11981 specification, TOE design, security architecture description and implementation representation to
11982 identify possible potential vulnerabilities in the TOE.

11983 A flaw hypothesis methodology needs to be used whereby specifications and development and
11984 guidance evidence are analysed and then potential vulnerabilities in the TOE are hypothesised, or
11985 speculated.

11986 The evaluator uses the knowledge of the TOE design and operation gained from the TOE
11987 deliverables to conduct a flaw hypothesis to identify potential flaws in the development of the TOE
11988 and potential errors in the specified method of operation of the TOE.

11989 The security architecture description provides the developer vulnerability analysis, as it
11990 documents how the TSF protects itself from interference from untrusted subjects and prevents the
11991 bypass of security enforcement functionality. Therefore, the evaluator should build upon the
11992 understanding of the TSF protection gained from the analysis of this evidence and then develop
11993 this in the knowledge gained from other development ADV evidence.

11994 The approach taken is directed by areas of concern identified during examination of the evidence
11995 during the conduct of evaluation activities and ensuring a representative sample of the
11996 development and guidance evidence provided for the evaluation is searched.

11997 For guidance on sampling see Annex A.2. This guidance should be considered when selecting the
11998 subset, giving reasons for:

11999 a) the approach used in selection;

12000 b) qualification that the evidence to be examined supports that approach.

12001 The areas of concern may relate to the sufficiency of specific protection features detailed in the
12002 security architecture description.

12003 The evidence to be considered during the vulnerability analysis may be linked to the evidence the
12004 attacker is assumed to be able to obtain. For example, the developer may protect the TOE design
12005 and implementation representations, so the only information assumed to be available to an
12006 attacker is the functional specification and guidance (publicly available). So, although the
12007 objectives for assurance in the TOE ensure the TOE design and implementation representation
12008 requirements are met, these design representations may only be searched to further investigate
12009 areas of concerns.

12010 On the other hand, if the source is publicly available it would be reasonable to assume that the
12011 attacker has access to the source and can use this in attempts to attack the TOE. Therefore, the
12012 source should be considered in the focused examination approach.

- 12013 The following indicates examples for the selection of the subset of evidence to be considered:
- 12014 a) For an evaluation where all levels of design abstraction from functional specification to
12015 implementation representation are provided, examination of information in the
12016 functional specification and the implementation representation may be selected, as the
12017 functional specification provides detail of interfaces available to an attacker, and the
12018 implementation representation incorporates the design decisions made at all other
12019 design abstractions. Therefore, the TOE design information will be considered as part of
12020 the implementation representation.
- 12021 b) Examination of a particular subset of information in each of the design representations
12022 provided for the evaluation.
- 12023 c) Coverage of particular SFRs through each of the design representations provided for the
12024 evaluation.
- 12025 d) Examination of each of the design representations provided for the evaluation,
12026 considering different SFRs within each design representations.
- 12027 e) Examination of aspects of the evidence provided for the evaluation relating to current
12028 potential vulnerability information the evaluator has received (e.g. from a scheme).
- 12029 This approach to identification of potential vulnerabilities is to take an ordered and planned
12030 approach; applying a system to the examination. The evaluator is to describe the method to be used
12031 in terms of what evidence will be considered, the information within the evidence that is to be
12032 examined, the manner in which this information is to be considered and the hypothesis that is to be
12033 created.
- 12034 The following provide some examples that a hypothesis may take:
- 12035 a) consideration of malformed input for interfaces available to an attacker at the external
12036 interfaces;
- 12037 b) examination of a key security mechanism cited in the security architecture description,
12038 such as process separation, hypothesising internal buffer overflows that may lead to
12039 degradation of separation;
- 12040 c) search to identify any objects created in the TOE implementation representation that are
12041 then not fully controlled by the TSF, and could be used by an attacker to undermine SFRs.
- 12042 For example, the evaluator may identify that interfaces are a potential area of weakness in the TOE
12043 and specify an approach to the search that “all interface specifications provided in the functional
12044 specification and TOE design will be searched to hypothesise potential vulnerabilities” and go on to
12045 explain the methods used in the hypothesis.
- 12046 The identification process is iterative, where the identification of one potential vulnerability may
12047 lead to identifying another area of concern that requires further investigation.
- 12048 The evaluator will report what actions were taken to identify potential vulnerabilities in the
12049 evidence. However, in this type of search, the evaluator may not be able to describe the steps in
12050 identifying potential vulnerabilities before the outset of the examination, as the approach may
12051 evolve as a result of findings during the search.
- 12052 The evaluator will report the evidence examine in completing the search for potential
12053 vulnerabilities. This selection of evidence may be derived from those areas of concern identified by
12054 the evaluator, linked to the evidence the attacker is assumed to be able to obtain, or according to
12055 another rationale provided by the evaluator.

12056 Subject to the SFRs the TOE is to meet in the operational environment, the evaluator's independent
12057 vulnerability analysis should consider generic potential vulnerabilities under each of the following
12058 headings:

12059 a) generic potential vulnerabilities relevant for the type of TOE being evaluated, as may be
12060 supplied by the evaluation authority;

12061 b) bypassing;

12062 c) tampering;

12063 d) direct attacks;

12064 e) monitoring;

12065 f) misuse.

12066 Items b) - f) are explained in greater detail in Annex B.

12067 The security architecture description should be considered in light of each of the above generic
12068 potential vulnerabilities. Each potential vulnerability should be considered to search for possible
12069 ways in which to defeat the TSF protection and undermine the TSF.

12070 **15.1.3.6.2 Work unit AVA_VAN.3-5**

12071 The evaluator *shall record* in the ETR the identified potential vulnerabilities that are candidates
12072 for testing and applicable to the TOE in its operational environment.

12073 It may be identified that no further consideration of the potential vulnerability is required if for
12074 example the evaluator identifies that measures in the operational environment, either IT or non-IT,
12075 prevent exploitation of the potential vulnerability in that operational environment. For instance,
12076 restricting physical access to the TOE to authorised users only may effectively render a potential
12077 vulnerability to tampering unexploitable.

12078 The evaluator records any reasons for exclusion of potential vulnerabilities from further
12079 consideration if the evaluator determines that the potential vulnerability is not applicable in the
12080 operational environment. Otherwise the evaluator records the potential vulnerability for further
12081 consideration.

12082 A list of potential vulnerabilities applicable to the TOE in its operational environment, which can be
12083 used as an input into penetration testing activities, shall be reported in the ETR by the evaluators.

12084 **15.1.3.7 Action AVA_VAN.3.4E**

12085 **15.1.3.7.1 Work unit AVA_VAN.3-6**

12086 The evaluator *shall devise* penetration tests, based on the independent search for potential
12087 vulnerabilities.

12088 The evaluator prepares for penetration testing as necessary to determine the susceptibility of the
12089 TOE, in its operational environment, to the potential vulnerabilities identified during the search of
12090 the sources of information publicly available. Any current information provided to the evaluator by
12091 a third party (e.g. evaluation authority) regarding known potential vulnerabilities will be
12092 considered by the evaluator, together with any encountered potential vulnerabilities resulting
12093 from the performance of other evaluation activities.

12094 The evaluator is reminded that, as for considering the security architecture description in the
12095 search for vulnerabilities (as detailed in AVA_VAN.3-4), testing should be performed to confirm the

- 12096 architectural properties. If requirements from ATE_DPT are included in the SARs, the developer
 12097 testing evidence will include testing performed to confirm the correct implementation of any
 12098 specific mechanisms detailed in the security architecture description. However, the developer
 12099 testing will not necessarily include testing of all aspects of the architectural properties that protect
 12100 the TSF, as much of this testing will be negative testing in nature, attempting to disprove the
 12101 properties. In developing the strategy for penetration testing, the evaluator will ensure that all
 12102 aspects of the security architecture description are tested, either in functional testing (as
 12103 considered in 14) or evaluator penetration testing.
- 12104 It will probably be practical to carry out penetration test using a series of test cases, where each
 12105 test case will test for a specific potential vulnerability.
- 12106 The evaluator is not expected to test for potential vulnerabilities (including those in the public
 12107 domain) beyond those which required an Enhanced-Basic attack potential. In some cases, however,
 12108 it will be necessary to carry out a test before the exploitability can be determined. Where, as a
 12109 result of evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond
 12110 Enhanced-Basic attack potential, this is reported in the ETR as a residual vulnerability.
- 12111 Guidance on determining the necessary attack potential to exploit a potential vulnerability can be
 12112 found in Annex B.4.
- 12113 Potential vulnerabilities hypothesised as exploitable only by attackers possessing Moderate or
 12114 High attack potential do not result in a failure of this evaluator action. Where analysis supports the
 12115 hypothesis, these need not be considered further as an input to penetration testing. However, such
 12116 vulnerabilities are reported in the ETR as residual vulnerabilities.
- 12117 Potential vulnerabilities hypothesised as exploitable by an attacker possessing a Basic or
 12118 Enhanced-Basic attack potential and resulting in a violation of the security objectives should be the
 12119 highest priority potential vulnerabilities comprising the list used to direct penetration testing
 12120 against the TOE.
- 12121 **15.1.3.7.2 Work unit AVA_VAN.3-7**
- 12122 The evaluator *shall produce* penetration test documentation for the tests based on the list of
 12123 potential vulnerabilities in sufficient detail to enable the tests to be repeatable. The test
 12124 documentation shall include:
- 12125 a) identification of the potential vulnerability the TOE is being tested for;
- 12126 b) instructions to connect and setup all required test equipment as required to conduct the
 12127 penetration test;
- 12128 c) instructions to establish all penetration test prerequisite initial conditions;
- 12129 d) instructions to stimulate the TSF;
- 12130 e) instructions for observing the behaviour of the TSF;
- 12131 f) descriptions of all expected results and the necessary analysis to be performed on the
 12132 observed behaviour for comparison against expected results;
- 12133 g) instructions to conclude the test and establish the necessary post-test state for the TOE.
- 12134 The evaluator prepares for penetration testing based on the list of potential vulnerabilities
 12135 identified during the search of the public domain and the analysis of the evaluation evidence.
- 12136 The evaluator is not expected to determine the exploitability for potential vulnerabilities beyond
 12137 those for which an Enhanced-Basic attack potential is required to effect an attack. However, as a

12138 result of evaluation expertise, the evaluator may discover a potential vulnerability that is
12139 exploitable only by an attacker with greater than Enhanced-Basic attack potential. Such
12140 vulnerabilities are to be reported in the ETR as residual vulnerabilities.

12141 With an understanding of the potential vulnerability, the evaluator determines the most feasible
12142 way to test for the TOE's susceptibility. Specifically the evaluator considers:

12143 a) the TSFI or other TOE interface that will be used to stimulate the TSF and observe
12144 responses (It is possible that the evaluator will need to use an interface to the TOE other
12145 than the TSFI to demonstrate properties of the TSF such as those described in the
12146 security architecture description (as required by ADV_ARC). It should be noted, that
12147 although these TOE interfaces provide a means of testing the TSF properties, they are not
12148 the subject of the test.);

12149 b) initial conditions that will need to exist for the test (i.e. any particular objects or subjects
12150 that will need to exist and security attributes they will need to have);

12151 c) special test equipment that will be required to either stimulate a TSFI or make
12152 observations of a TSFI (although it is unlikely that specialist equipment would be
12153 required to exploit a potential vulnerability assuming an Enhanced-Basic attack
12154 potential);

12155 d) whether theoretical analysis should replace physical testing, particularly relevant where
12156 the results of an initial test can be extrapolated to demonstrate that repeated attempts of
12157 an attack are likely to succeed after a given number of attempts.

12158 The evaluator will probably find it practical to carry out penetration testing using a series of test
12159 cases, where each test case will test for a specific potential vulnerability.

12160 The intent of specifying this level of detail in the test documentation is to allow another evaluator
12161 to repeat the tests and obtain an equivalent result.

12162 **15.1.3.7.3 Work unit AVA_VAN.3-8**

12163 The evaluator *shall conduct* penetration testing.

12164 The evaluator uses the penetration test documentation resulting from work unit AVA_VAN.3-6 as a
12165 basis for executing penetration tests on the TOE, but this does not preclude the evaluator from
12166 performing additional ad hoc penetration tests. If required, the evaluator may devise ad hoc tests
12167 as a result of information learnt during penetration testing that, if performed by the evaluator, are
12168 to be recorded in the penetration test documentation. Such tests may be required to follow up
12169 unexpected results or observations, or to investigate potential vulnerabilities suggested to the
12170 evaluator during the pre-planned testing.

12171 Should penetration testing show that a hypothesised potential vulnerability does not exist, then the
12172 evaluator should determine whether or not the evaluator's own analysis was incorrect, or if
12173 evaluation deliverables are incorrect or incomplete.

12174 The evaluator is not expected to test for potential vulnerabilities (including those in the public
12175 domain) beyond those which required an Enhanced-Basic attack potential. In some cases, however,
12176 it will be necessary to carry out a test before the exploitability can be determined. Where, as a
12177 result of evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond
12178 Enhanced-Basic attack potential, this is reported in the ETR as a residual vulnerability.

12179 **15.1.3.7.4 Work unit AVA_VAN.3-9**

12180 The evaluator *shall record* the actual results of the penetration tests.

12181 While some specific details of the actual test results may be different from those expected (e.g. time
12182 and date fields in an audit record) the overall result should be identical. Any unexpected test
12183 results should be investigated. The impact on the evaluation should be stated and justified.

12184 **15.1.3.7.5 Work unit AVA_VAN.3-10**

12185 The evaluator **shall report** in the ETR the evaluator penetration testing effort, outlining the testing
12186 approach, configuration, depth and results.

12187 The penetration testing information reported in the ETR allows the evaluator to convey the overall
12188 penetration testing approach and effort expended on this sub-activity. The intent of providing this
12189 information is to give a meaningful overview of the evaluator's penetration testing effort. It is not
12190 intended that the information regarding penetration testing in the ETR be an exact reproduction of
12191 specific test steps or results of individual penetration tests. The intention is to provide enough
12192 detail to allow other evaluators and evaluation authorities to gain some insight about the
12193 penetration testing approach chosen, amount of penetration testing performed, TOE test
12194 configurations, and the overall results of the penetration testing activity.

12195 Information that would typically be found in the ETR subclause regarding evaluator penetration
12196 testing efforts is:

12197 a) TOE test configurations. The particular configurations of the TOE that were penetration
12198 tested;

12199 b) TSFI penetration tested. A brief listing of the TSFI and other TOE interfaces that were the
12200 focus of the penetration testing;

12201 c) Verdict for the sub-activity. The overall judgement on the results of penetration testing.

12202 This list is by no means exhaustive and is only intended to provide some context as to the type of
12203 information that should be present in the ETR concerning the penetration testing the evaluator
12204 performed during the evaluation.

12205 **15.1.3.7.6 Work unit AVA_VAN.3-11**

12206 The evaluator **shall examine** the results of all penetration testing to determine that the TOE, in its
12207 operational environment, is resistant to an attacker possessing an Enhanced-Basic attack potential.

12208 If the results reveal that the TOE, in its operational environment, has vulnerabilities exploitable by
12209 an attacker possessing less than Moderate attack potential, then this evaluator action fails.

12210 The guidance in B.4 should be used to determine the attack potential required to exploit a
12211 particular vulnerability and whether it can therefore be exploited in the intended environment. It
12212 may not be necessary for the attack potential to be calculated in every instance, only if there is
12213 some doubt as to whether or not the vulnerability can be exploited by an attacker possessing an
12214 attack potential less than Moderate.

12215 **15.1.3.7.7 Work unit AVA_VAN.3-12**

12216 The evaluator **shall report** in the ETR all exploitable vulnerabilities and residual vulnerabilities,
12217 detailing for each:

12218 a) its source (e.g. evaluation methodology activity being undertaken when it was conceived,
12219 known to the evaluator, read in a publication);

12220 b) the SFR(s) not met;

12221 c) a description;

- 12222 d) whether it is exploitable in its operational environment or not (i.e. exploitable or
12223 residual).
- 12224 e) the amount of time, level of expertise, level of knowledge of the TOE, level of opportunity
12225 and the equipment required to perform the identified vulnerabilities, and the
12226 corresponding values using the tables B.2 and B.3 of Annex B.4.
- 12227 **15.1.4 Evaluation of sub-activity (AVA_VAN.4)**
- 12228 **15.1.4.1 Objectives**
- 12229 The objective of this sub-activity is to determine whether the TOE, in its operational environment,
12230 has vulnerabilities exploitable by attackers possessing Moderate attack potential.
- 12231 **15.1.4.2 Input**
- 12232 The evaluation evidence for this sub-activity is:
- 12233 a) the ST;
- 12234 b) the functional specification;
- 12235 c) the TOE design;
- 12236 d) the security architecture description;
- 12237 e) the implementation representation;
- 12238 f) the guidance documentation;
- 12239 g) the TOE suitable for testing;
- 12240 h) information publicly available to support the identification of possible potential
12241 vulnerabilities;
- 12242 i) the results of the testing of the basic design.
- 12243 The remaining implicit evaluation evidence for this sub-activity depends on the components that
12244 have been included in the assurance package. The evidence provided for each component is to be
12245 used as input in this sub-activity.
- 12246 Other input for this sub-activity is:
- 12247 a) current information regarding public domain potential vulnerabilities and attacks (e.g.
12248 from an evaluation authority).
- 12249 **15.1.4.3 Application notes**
- 12250 The methodical analysis approach takes the form of a structured examination of the evidence. This
12251 method requires the evaluator to specify the structure and form the analysis will take (i.e. the
12252 manner in which the analysis is performed is predetermined, unlike the focused analysis). The
12253 method is specified in terms of the information that will be considered and how/why it will be
12254 considered. Further guidance on methodical vulnerability analysis can be found in Annex B.2.2.2.3.
- 12255 **15.1.4.4 Action AVA_VAN.4.1E**
- 12256 ISO/IEC 15408-3 AVA_VAN.4.1C: *The TOE shall be suitable for testing.*

- 12257 **15.1.4.4.1 Work unit AVA_VAN.4-1**
- 12258 The evaluator *shall examine* the TOE to determine that the test configuration is consistent with
12259 the configuration under evaluation as specified in the ST.
- 12260 The TOE provided by the developer and identified in the test plan should have the same unique
12261 reference as established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST
12262 introduction.
- 12263 It is possible for the ST to specify more than one configuration for evaluation. The TOE may
12264 comprise a number of distinct hardware and software entities that need to be tested in accordance
12265 with the ST. The evaluator verifies that all test configurations are consistent with the ST.
- 12266 The evaluator should consider the security objectives for the operational environment described in
12267 the ST that may apply to the test environment and ensure they are met in the testing environment.
12268 There may be some objectives for the operational environment that do not apply to the test
12269 environment. For example, an objective about user clearances may not apply; however, an
12270 objective about a single point of connection to a network would apply.
- 12271 If any test resources are used (e.g. meters, analysers) it will be the evaluator's responsibility to
12272 ensure that these resources are calibrated correctly.
- 12273 **15.1.4.4.2 Work unit AVA_VAN.4-2**
- 12274 The evaluator *shall examine* the TOE to determine that it has been installed properly and is in a
12275 known state
- 12276 It is possible for the evaluator to determine the state of the TOE in a number of ways. For example,
12277 previous successful completion of the Evaluation of sub-activity (AGD_PRE.1) sub-activity will
12278 satisfy this work unit if the evaluator still has confidence that the TOE being used for testing was
12279 installed properly and is in a known state. If this is not the case, then the evaluator should follow
12280 the developer's procedures to install and start up the TOE, using the supplied guidance only.
- 12281 If the evaluator has to perform the installation procedures because the TOE is in an unknown state,
12282 this work unit when successfully completed could satisfy work unit AGD_PRE.1-3.
- 12283 **15.1.4.5 Action AVA_VAN.4.2E**
- 12284 **15.1.4.5.1 Work unit AVA_VAN.4-3**
- 12285 The evaluator *shall examine* sources of information publicly available to identify potential
12286 vulnerabilities in the TOE.
- 12287 The evaluator examines the sources of information publicly available to support the identification
12288 of possible potential vulnerabilities in the TOE. There are many sources of publicly available
12289 information which the evaluator should consider using items such as those available on the world
12290 wide web, including:
- 12291 a) specialist publications (magazines, books);
- 12292 b) research papers;
- 12293 c) conference proceedings.
- 12294 The evaluator should not constrain their consideration of publicly available information to the
12295 above, but should consider any other relevant information available.

12296 While examining the evidence provided the evaluator will use the information in the public domain
12297 to further search for potential vulnerabilities. Where the evaluators have identified areas of
12298 concern, the evaluator should consider information publicly available that relate to those areas of
12299 concern.

12300 The availability of information that may be readily available to an attacker that helps to identify
12301 and facilitate attacks may substantially enhance the attack potential of a given attacker. The
12302 accessibility of vulnerability information and sophisticated attack tools on the Internet makes it
12303 more likely that this information will be used in attempts to identify potential vulnerabilities in the
12304 TOE and exploit them. Modern search tools make such information easily available to the evaluator,
12305 and the determination of resistance to published potential vulnerabilities and well known generic
12306 attacks can be achieved in a cost-effective manner.

12307 The search of the information publicly available should be focused on those sources that refer to
12308 the technologies used in the development of the product from which the TOE is derived. The
12309 extensiveness of this search should consider the following factors: TOE type, evaluator experience
12310 in this TOE type, expected attack potential and the level of ADV evidence available.

12311 The identification process is iterative, where the identification of one potential vulnerability may
12312 lead to identifying another area of concern that requires further investigation.

12313 The evaluator will describe the approach to be taken to identify potential vulnerabilities in the
12314 publicly available material, detailing the search to be performed. This may be driven by factors
12315 such as areas of concern identified by the evaluator, linked to the evidence the attacker is assumed
12316 to be able to obtain. However, it is recognised that in this type of search the approach may further
12317 evolve as a result of findings during the search. Therefore, the evaluator will also report any
12318 actions taken in addition to those described in the approach to further investigate issues thought to
12319 lead to potential vulnerabilities, and will report the evidence examined in completing the search
12320 for potential vulnerabilities.

12321 **15.1.4.6 Action AVA_VAN.4.3E**

12322 **15.1.4.6.1 Work unit AVA_VAN.4-4**

12323 The evaluator *shall conduct* a methodical analysis of ST, guidance documentation, functional
12324 specification, TOE design, security architecture description and implementation representation to
12325 identify possible potential vulnerabilities in the TOE.

12326 Guidance on methodical vulnerability analysis is provided in Annex B.2.2.2.3.

12327 This approach to identification of potential vulnerabilities is to take an ordered and planned
12328 approach. A system is to be applied in the examination. The evaluator is to describe the method to
12329 be used in terms of the manner in which this information is to be considered and the hypothesis
12330 that is to be created.

12331 A flaw hypothesis methodology needs to be used whereby the ST, development (functional
12332 specification, TOE design and implementation representation) and guidance evidence are analysed
12333 and then vulnerabilities in the TOE are hypothesised, or speculated.

12334 The evaluator uses the knowledge of the TOE design and operation gained from the TOE
12335 deliverables to conduct a flaw hypothesis to identify potential flaws in the development of the TOE
12336 and potential errors in the specified method of operation of the TOE.

12337 The security architecture description provides the developer vulnerability analysis, as it
12338 documents how the TSF protects itself from interference from untrusted subjects and prevents the
12339 bypass of security enforcement functionality. Therefore, the evaluator should build upon the
12340 understanding of the TSF protection gained from the analysis of this evidence and then develop
12341 this in the knowledge gained from other development ADV evidence.

12342 The approach taken to the methodical search for vulnerabilities is to consider any areas of concern
 12343 identified in the results of the evaluator's assessment of the development and guidance evidence.
 12344 However, the evaluator should also consider each aspect of the security architecture analysis to
 12345 search for any ways in which the protection of the TSF can be undermined. It may be helpful to
 12346 structure the methodical analysis on the basis of the material presented in the security architecture
 12347 description, introducing concerns from other ADV evidence as appropriate. The analysis can then
 12348 be further developed to ensure all other material from the ADV evidence is considered.

12349 The following provide some examples of hypotheses that may be created when examining the
 12350 evidence:

12351 a) consideration of malformed input for interfaces available to an attacker at the external
 12352 interfaces;

12353 b) examination of a key security mechanism cited in the security architecture description,
 12354 such as process separation, hypothesising internal buffer overflows that may lead to
 12355 degradation of separation;

12356 c) search to identify any objects created in the TOE implementation representation that are
 12357 then not fully controlled by the TSF, and could be used by an attacker to undermine SFRs.

12358 For example, the evaluator may identify that interfaces are a potential area of weakness in the TOE
 12359 and specify an approach to the search that 'all interface specifications in the evidence provided will
 12360 be searched to hypothesise potential vulnerabilities' and go on to explain the methods used in the
 12361 hypothesis.

12362 In addition, areas of concern the evaluator has identified during examination of the evidence
 12363 during the conduct of evaluation activities. Areas of concern may also be identified during the
 12364 conduct of other work units associated with this component, in particular AVA_VAN.4-7,
 12365 AVA_VAN.4-5 and AVA_VAN.4-6 where the development and conduct of penetration tests may
 12366 identify further areas of concerns for investigation, or potential vulnerabilities.

12367 However, examination of only a subset of the development and guidance evidence or their contents
 12368 is not permitted in this level of rigour. The approach description should provide a demonstration
 12369 that the methodical approach used is complete, providing confidence that the approach used to
 12370 search the deliverables has considered all of the information provided in those deliverables.

12371 This approach to identification of potential vulnerabilities is to take an ordered and planned
 12372 approach; applying a system to the examination. The evaluator is to describe the method to be used
 12373 in terms of how the evidence will be considered; the manner in which this information is to be
 12374 considered and the hypothesis that is to be created. This approach should be agreed with the
 12375 evaluation authority, and the evaluation authority may provide detail of any additional approaches
 12376 the evaluator should take to the vulnerability analysis and identify any additional information that
 12377 should be considered by the evaluator.

12378 Although a system to identifying potential vulnerabilities is predefined, the identification process
 12379 may still be iterative, where the identification of one potential vulnerability may lead to identifying
 12380 another area of concern that requires further investigation.

12381 Subject to the SFRs the TOE is to meet in the operational environment, the evaluator's independent
 12382 vulnerability analysis should consider generic potential vulnerabilities under each of the following
 12383 headings:

12384 a) generic potential vulnerabilities relevant for the type of TOE being evaluated, as may be
 12385 supplied by the evaluation authority;

12386 b) bypassing;

12387 c) tampering;

12388 d) direct attacks;

12389 e) monitoring;

12390 f) misuse.

12391 Items b) - f) are explained in greater detail in Annex B.

12392 The security architecture description should be considered in light of each of the above generic
12393 potential vulnerabilities. Each potential vulnerability should be considered to search for possible
12394 ways in which to defeat the TSF protection and undermine the TSF.

12395 **15.1.4.6.2 Work unit AVA_VAN.4-5**

12396 The evaluator *shall record* in the ETR the identified potential vulnerabilities that are candidates
12397 for testing and applicable to the TOE in its operational environment.

12398 It may be identified that no further consideration of the potential vulnerability is required if for
12399 example the evaluator identifies that measures in the operational environment, either IT or non-IT,
12400 prevent exploitation of the potential vulnerability in that operational environment. For instance,
12401 restricting physical access to the TOE to authorised users only may effectively render a potential
12402 vulnerability to tampering unexploitable.

12403 The evaluator records any reasons for exclusion of potential vulnerabilities from further
12404 consideration if the evaluator determines that the potential vulnerability is not applicable in the
12405 operational environment. Otherwise the evaluator records the potential vulnerability for further
12406 consideration.

12407 A list of potential vulnerabilities applicable to the TOE in its operational environment, which can be
12408 used as an input into penetration testing activities, shall be reported in the ETR by the evaluators.

12409 **15.1.4.7 Action AVA_VAN.4.4E**

12410 **15.1.4.7.1 Work unit AVA_VAN.4-6**

12411 The evaluator *shall devise* penetration tests, based on the independent search for potential
12412 vulnerabilities.

12413 The evaluator prepares for penetration testing as necessary to determine the susceptibility of the
12414 TOE, in its operational environment, to the potential vulnerabilities identified during the search of
12415 the sources of information publicly available. Any current information provided to the evaluator by
12416 a third party (e.g. evaluation authority) regarding known potential vulnerabilities will be
12417 considered by the evaluator, together with any encountered potential vulnerabilities resulting
12418 from the performance of other evaluation activities.

12419 The evaluator is reminded that, as for considering the security architecture description in the
12420 search for vulnerabilities (as detailed in AVA_VAN.4-3), testing should be performed to confirm the
12421 architectural properties. If requirements from ATE_DPT are included in the SARs, the developer
12422 testing evidence will include testing performed to confirm the correct implementation of any
12423 specific mechanisms detailed in the security architecture description. However, the developer
12424 testing will not necessarily include testing of all aspects of the architectural properties that protect
12425 the TSF, as much of this testing will be negative testing in nature, attempting to disprove the
12426 properties. In developing the strategy for penetration testing, the evaluator will ensure that all
12427 aspects of the security architecture description are tested, either in functional testing (as
12428 considered in 14) or evaluator penetration testing.

- 12429 The evaluator will probably find it practical to carry out penetration test using a series of test cases,
12430 where each test case will test for a specific potential vulnerability.
- 12431 The evaluator is not expected to test for potential vulnerabilities (including those in the public
12432 domain) beyond those which required a Moderate attack potential. In some cases, however, it will
12433 be necessary to carry out a test before the exploitability can be determined. Where, as a result of
12434 evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond Moderate
12435 attack potential, this is reported in the ETR as a residual vulnerability.
- 12436 Guidance on determining the necessary attack potential to exploit a potential vulnerability can be
12437 found in Annex B.4.
- 12438 Potential vulnerabilities hypothesised as exploitable by an attacker possessing a Moderate (or less)
12439 attack potential and resulting in a violation of the security objectives should be the highest priority
12440 potential vulnerabilities comprising the list used to direct penetration testing against the TOE.
- 12441 **15.1.4.7.2 Work unit AVA_VAN.4-7**
- 12442 The evaluator *shall produce* penetration test documentation for the tests based on the list of
12443 potential vulnerabilities in sufficient detail to enable the tests to be repeatable. The test
12444 documentation shall include:
- 12445 a) identification of the potential vulnerability the TOE is being tested for;
- 12446 b) instructions to connect and setup all required test equipment as required to conduct the
12447 penetration test;
- 12448 c) instructions to establish all penetration test prerequisite initial conditions;
- 12449 d) instructions to stimulate the TSF;
- 12450 e) instructions for observing the behaviour of the TSF;
- 12451 f) descriptions of all expected results and the necessary analysis to be performed on the
12452 observed behaviour for comparison against expected results;
- 12453 g) instructions to conclude the test and establish the necessary post-test state for the TOE.
- 12454 The evaluator prepares for penetration testing based on the list of potential vulnerabilities
12455 identified during the search of the public domain and the analysis of the evaluation evidence.
- 12456 The evaluator is not expected to determine the exploitability for potential vulnerabilities beyond
12457 those for which a Moderate attack potential is required to effect an attack. However, as a result of
12458 evaluation expertise, the evaluator may discover a potential vulnerability that is exploitable only
12459 by an attacker with greater than Moderate attack potential. Such vulnerabilities are to be reported
12460 in the ETR as residual vulnerabilities.
- 12461 With an understanding of the potential vulnerability, the evaluator determines the most feasible
12462 way to test for the TOE's susceptibility. Specifically the evaluator considers:
- 12463 a) the TSFI or other TOE interface that will be used to stimulate the TSF and observe
12464 responses (It is possible that the evaluator will need to use an interface to the TOE other
12465 than the TSFI to demonstrate properties of the TSF such as those described in the
12466 security architecture description (as required by ADV_ARC). It should be noted, that
12467 although these TOE interfaces provide a means of testing the TSF properties, they are not
12468 the subject of the test.);

- 12469 b) initial conditions that will need to exist for the test (i.e. any particular objects or subjects
12470 that will need to exist and security attributes they will need to have);
- 12471 c) special test equipment that will be required to either stimulate a TSFI or make
12472 observations of a TSFI;
- 12473 d) whether theoretical analysis should replace physical testing, particularly relevant where
12474 the results of an initial test can be extrapolated to demonstrate that repeated attempts of
12475 an attack are likely to succeed after a given number of attempts.

12476 The evaluator will probably find it practical to carry out penetration testing using a series of test
12477 cases, where each test case will test for a specific potential vulnerability.

12478 The intent of specifying this level of detail in the test documentation is to allow another evaluator
12479 to repeat the tests and obtain an equivalent result.

12480 **15.1.4.7.3 Work unit AVA_VAN.4-8**

12481 The evaluator *shall conduct* penetration testing.

12482 The evaluator uses the penetration test documentation resulting from work unit AVA_VAN.4-6 as a
12483 basis for executing penetration tests on the TOE, but this does not preclude the evaluator from
12484 performing additional ad hoc penetration tests. If required, the evaluator may devise ad hoc tests
12485 as a result of information learnt during penetration testing that, if performed by the evaluator, are
12486 to be recorded in the penetration test documentation. Such tests may be required to follow up
12487 unexpected results or observations, or to investigate potential vulnerabilities suggested to the
12488 evaluator during the pre-planned testing.

12489 Should penetration testing show that a hypothesised potential vulnerability does not exist, then the
12490 evaluator should determine whether or not the evaluator's own analysis was incorrect, or if
12491 evaluation deliverables are incorrect or incomplete.

12492 The evaluator is not expected to test for potential vulnerabilities (including those in the public
12493 domain) beyond those which required a Moderate attack potential. In some cases, however, it will
12494 be necessary to carry out a test before the exploitability can be determined. Where, as a result of
12495 evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond Moderate
12496 attack potential, this is reported in the ETR as a residual vulnerability.

12497 **15.1.4.7.4 Work unit AVA_VAN.4-9**

12498 The evaluator *shall record* the actual results of the penetration tests.

12499 While some specific details of the actual test results may be different from those expected (e.g. time
12500 and date fields in an audit record) the overall result should be identical. Any unexpected test
12501 results should be investigated. The impact on the evaluation should be stated and justified.

12502 **15.1.4.7.5 Work unit AVA_VAN.4-10**

12503 The evaluator *shall report* in the ETR the evaluator penetration testing effort, outlining the testing
12504 approach, configuration, depth and results.

12505 The penetration testing information reported in the ETR allows the evaluator to convey the overall
12506 penetration testing approach and effort expended on this sub-activity. The intent of providing this
12507 information is to give a meaningful overview of the evaluator's penetration testing effort. It is not
12508 intended that the information regarding penetration testing in the ETR be an exact reproduction of
12509 specific test steps or results of individual penetration tests. The intention is to provide enough
12510 detail to allow other evaluators and evaluation authorities to gain some insight about the

12511 penetration testing approach chosen, amount of penetration testing performed, TOE test
12512 configurations, and the overall results of the penetration testing activity.

12513 Information that would typically be found in the ETR subclause regarding evaluator penetration
12514 testing efforts is:

12515 a) TOE test configurations. The particular configurations of the TOE that were penetration
12516 tested;

12517 b) TSFI penetration tested. A brief listing of the TSFI and other TOE interfaces that were the
12518 focus of the penetration testing;

12519 c) Verdict for the sub-activity. The overall judgement on the results of penetration testing.

12520 This list is by no means exhaustive and is only intended to provide some context as to the type of
12521 information that should be present in the ETR concerning the penetration testing the evaluator
12522 performed during the evaluation.

12523 **15.1.4.7.6 Work unit AVA_VAN.4-11**

12524 The evaluator *shall examine* the results of all penetration testing to determine that the TOE, in its
12525 operational environment, is resistant to an attacker possessing a Moderate attack potential.

12526 If the results reveal that the TOE, in its operational environment, has vulnerabilities exploitable by
12527 an attacker possessing less than a High attack potential, then this evaluator action fails.

12528 The guidance in B.4 should be used to determine the attack potential required to exploit a
12529 particular vulnerability and whether it can therefore be exploited in the intended environment. It
12530 may not be necessary for the attack potential to be calculated in every instance, only if there is
12531 some doubt as to whether or not the vulnerability can be exploited by an attacker possessing an
12532 attack potential less than High.

12533 **15.1.4.7.7 Work unit AVA_VAN.4-12**

12534 The evaluator *shall report* in the ETR all exploitable vulnerabilities and residual vulnerabilities,
12535 detailing for each:

12536 a) its source (e.g. evaluation methodology activity being undertaken when it was conceived,
12537 known to the evaluator, read in a publication);

12538 b) the SFR(s) not met;

12539 c) a description;

12540 d) whether it is exploitable in its operational environment or not (i.e. exploitable or
12541 residual).

12542 e) the amount of time, level of expertise, level of knowledge of the TOE, level of opportunity
12543 and the equipment required to perform the identified vulnerabilities, and the
12544 corresponding values using the tables B.2 and B.3 of Annex B.4.

12545 **15.1.5 Evaluation of sub-activity (AVA_VAN.5)**

12546 The work units for the evaluation of the sub-activity AVA_VAN.5 are copied from the work units of
12547 AVA_VAN.4 as far as possible except that the TOE is attacked by attackers possessing High attack
12548 potential.

12549 **15.1.5.1 Objectives**

12550 The objective of this sub-activity is to determine whether the TOE, in its operational environment,
12551 has vulnerabilities exploitable by attackers possessing **High** attack potential.

12552 **15.1.5.2 Input**

12553 The evaluation evidence for this sub-activity is:

12554 a) the ST;

12555 b) the functional specification;

12556 c) the TOE design;

12557 d) the security architecture description;

12558 e) the implementation representation;

12559 f) the guidance documentation;

12560 g) the TOE suitable for testing;

12561 h) information publicly available to support the identification of possible potential
12562 vulnerabilities;

12563 i) the results of the testing of the basic design.

12564 The remaining implicit evaluation evidence for this sub-activity depends on the components that
12565 have been included in the assurance package. The evidence provided for each component is to be
12566 used as input in this sub-activity.

12567 Other input for this sub-activity is:

12568 a) current information regarding public domain potential vulnerabilities and attacks (e.g.
12569 from an evaluation authority).

12570 **15.1.5.3 Application notes**

12571 The methodical analysis approach takes the form of a structured examination of the evidence. This
12572 method requires the evaluator to specify the structure and form the analysis will take (i.e. the
12573 manner in which the analysis is performed is predetermined, unlike the focused analysis). The
12574 method is specified in terms of the information that will be considered and how/why it will be
12575 considered. Further guidance on methodical vulnerability analysis can be found in Annex B.2.2.2.3.

12576 **15.1.5.4 Action AVA_VAN.5.1E**

12577 **AVA_VAN.5.1C**

12578 The TOE shall be suitable for testing.

12579 **15.1.5.4.1 Work unit AVA_VAN.5-1**

12580 The evaluator *shall examine* the TOE to determine that the test configuration is consistent with
12581 the configuration under evaluation as specified in the ST.

- 12582 The TOE provided by the developer and identified in the test plan should have the same unique
12583 reference as established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST
12584 introduction.
- 12585 It is possible for the ST to specify more than one configuration for evaluation. The TOE may
12586 comprise a number of distinct hardware and software entities that need to be tested in accordance
12587 with the ST. The evaluator verifies that all test configurations are consistent with the ST.
- 12588 The evaluator should consider the security objectives for the operational environment described in
12589 the ST that may apply to the test environment and ensure they are met in the testing environment.
12590 There may be some objectives for the operational environment that do not apply to the test
12591 environment. For example, an objective about user clearances may not apply; however, an
12592 objective about a single point of connection to a network would apply.
- 12593 If any test resources are used (e.g. meters, analysers) it will be the evaluator's responsibility to
12594 ensure that these resources are calibrated correctly.
- 12595 **15.1.5.4.2 Work unit AVA_VAN.5-2**
- 12596 The evaluator *shall examine* the TOE to determine that it has been installed properly and is in a
12597 known state
- 12598 It is possible for the evaluator to determine the state of the TOE in a number of ways. For example,
12599 previous successful completion of the Evaluation of sub-activity (AGD_PRE.1) sub-activity will
12600 satisfy this work unit if the evaluator still has confidence that the TOE being used for testing was
12601 installed properly and is in a known state. If this is not the case, then the evaluator should follow
12602 the developer's procedures to install and start up the TOE, using the supplied guidance only.
- 12603 If the evaluator has to perform the installation procedures because the TOE is in an unknown state,
12604 this work unit when successfully completed could satisfy work unit AGD_PRE.1-3.
- 12605 **15.1.5.5 Action AVA_VAN.5.2E**
- 12606 **15.1.5.5.1 Work unit AVA_VAN.5-3**
- 12607 The evaluator *shall examine* sources of information publicly available to identify potential
12608 vulnerabilities in the TOE.
- 12609 The evaluator examines the sources of information publicly available to support the identification
12610 of possible potential vulnerabilities in the TOE. There are many sources of publicly available
12611 information which the evaluator should consider using items such as those available on the world
12612 wide web, including:
- 12613 a) specialist publications (magazines, books);
- 12614 b) research papers;
- 12615 c) conference proceedings.
- 12616 The evaluator should not constrain their consideration of publicly available information to the
12617 above, but should consider any other relevant information available.
- 12618 While examining the evidence provided the evaluator will use the information in the public domain
12619 to further search for potential vulnerabilities. Where the evaluators have identified areas of
12620 concern, the evaluator should consider information publicly available that relate to those areas of
12621 concern.

12622 The availability of information that may be readily available to an attacker that helps to identify
12623 and facilitate attacks may substantially enhance the attack potential of a given attacker. The
12624 accessibility of vulnerability information and sophisticated attack tools on the Internet makes it
12625 more likely that this information will be used in attempts to identify potential vulnerabilities in the
12626 TOE and exploit them. Modern search tools make such information easily available to the evaluator,
12627 and the determination of resistance to published potential vulnerabilities and well known generic
12628 attacks can be achieved in a cost-effective manner.

12629 The search of the information publicly available should be focused on those sources that refer to
12630 the technologies used in the development of the product from which the TOE is derived. The
12631 extensiveness of this search should consider the following factors: TOE type, evaluator experience
12632 in this TOE type, expected attack potential and the level of ADV evidence available.

12633 The identification process is iterative, where the identification of one potential vulnerability may
12634 lead to identifying another area of concern that requires further investigation.

12635 The evaluator will describe the approach to be taken to identify potential vulnerabilities in the
12636 publicly available material, detailing the search to be performed. This may be driven by factors
12637 such as areas of concern identified by the evaluator, linked to the evidence the attacker is assumed
12638 to be able to obtain. However, it is recognised that in this type of search the approach may further
12639 evolve as a result of findings during the search. Therefore, the evaluator will also report any
12640 actions taken in addition to those described in the approach to further investigate issues thought to
12641 lead to potential vulnerabilities, and will report the evidence examined in completing the search
12642 for potential vulnerabilities.

12643 **15.1.5.6 Action AVA_VAN.5.3E**

12644 **15.1.5.6.1 Work unit AVA_VAN.5-4**

12645 The evaluator **shall conduct** a methodical analysis of ST, guidance documentation, functional
12646 specification, TOE design, security architecture description and implementation representation to
12647 identify possible potential vulnerabilities in the TOE.

12648 Guidance on methodical vulnerability analysis is provided in Annex B.2.2.2.3.

12649 This approach to identification of potential vulnerabilities is to take an ordered and planned
12650 approach. A system is to be applied in the examination. The evaluator is to describe the method to
12651 be used in terms of the manner in which this information is to be considered and the hypothesis
12652 that is to be created.

12653 A flaw hypothesis methodology should be used whereby the ST, development (functional
12654 specification, TOE design and implementation representation) and guidance evidence are analysed
12655 and then vulnerabilities in the TOE are hypothesised, or speculated.

12656 The evaluator should use the knowledge of the TOE design and operation gained from the TOE
12657 deliverables to conduct a flaw hypothesis to identify potential flaws in the development of the TOE
12658 and potential errors in the specified method of operation of the TOE.

12659 The security architecture description provides the developer vulnerability analysis, as it
12660 documents how the TSF protects itself from interference from untrusted subjects and prevents the
12661 bypass of security enforcement functionality. Therefore, the evaluator should build upon the
12662 understanding of the TSF protection gained from the analysis of this evidence and then develop
12663 this in the knowledge gained from other development (e.g. ADV) evidence.

12664 The approach taken to the methodical search for vulnerabilities is to consider any areas of concern
12665 identified in the results of the evaluator's assessment of the development and guidance evidence.
12666 However, the evaluator should also consider each aspect of the security architecture analysis to
12667 search for any ways in which the protection of the TSF can be undermined. It may be helpful to

- 12668 structure the methodical analysis on the basis of the material presented in the security architecture
 12669 description, introducing concerns from other ADV evidence as appropriate. The analysis can then
 12670 be further developed to ensure all other material from the ADV evidence is considered.
- 12671 The following provide some examples of hypotheses that may be created when examining the
 12672 evidence:
- 12673 consideration of malformed input for interfaces available to an attacker at the external interfaces;
- 12674 examination of a key security mechanism cited in the security architecture description, such as
 12675 process separation, hypothesising internal buffer overflows that may lead to degradation of
 12676 separation;
- 12677 search to identify any objects created in the TOE implementation representation that are then not
 12678 fully controlled by the TSF, and could be used by an attacker to undermine SFRs.
- 12679 For example, the evaluator may identify that interfaces are a potential area of weakness in the TOE
 12680 and specify an approach to the search that 'all interface specifications in the evidence provided will
 12681 be searched to hypothesise potential vulnerabilities' and go on to explain the methods used in the
 12682 hypothesis.
- 12683 In addition, areas of concern the evaluator has identified during examination of the evidence
 12684 during the conduct of evaluation activities. Areas of concern may also be identified during the
 12685 conduct of other work units associated with this component, in particular AVA_VAN.5-7,
 12686 AVA_VAN.5-5 and AVA_VAN.5-6) where the development and conduct of penetration tests may
 12687 identify further areas of concerns for investigation, or potential vulnerabilities.
- 12688 However, examination of only a subset of the development and guidance evidence or their contents
 12689 is not permitted in this level of rigour. The approach description should provide a demonstration
 12690 that the methodical approach used is complete, providing confidence that the approach used to
 12691 search the deliverables has considered all of the information provided in those deliverables.
- 12692 This approach to identification of potential vulnerabilities is to take an ordered and planned
 12693 approach; applying a system to the examination. The evaluator is to describe the method to be used
 12694 in terms of how the evidence will be considered; the manner in which this information is to be
 12695 considered and the hypothesis that is to be created. This approach should be agreed with the
 12696 evaluation authority, and the evaluation authority should provide detail of any additional
 12697 approaches the evaluator should take to the vulnerability analysis and identify any additional
 12698 information that should be considered by the evaluator.
- 12699 Although a system to identifying potential vulnerabilities is predefined, the identification process
 12700 may still be iterative, where the identification of one potential vulnerability may lead to identifying
 12701 another area of concern that requires further investigation.
- 12702 Subject to the SFRs the TOE is to meet in the operational environment, the evaluator's independent
 12703 vulnerability analysis should consider generic potential vulnerabilities under each of the following
 12704 headings:
- 12705 a) generic potential vulnerabilities relevant for the type of TOE being evaluated, as
 12706 may be supplied by the evaluation authority;
- 12707 b) bypassing;
- 12708 c) tampering;
- 12709 d) direct attacks;
- 12710 e) monitoring;

- 12711 f) misuse.
- 12712 Items b) - f) are explained in greater detail in Annex B.2.1.
- 12713 The security architecture description should be considered in light of each of the above generic
12714 potential vulnerabilities. Each potential vulnerability should be considered to search for possible
12715 ways in which to defeat the TSF protection and undermine the TSF.
- 12716 **15.1.5.6.2 Work unit AVA_VAN.5-5**
- 12717 The evaluator *shall record* in the ETR the identified potential vulnerabilities that are candidates
12718 for testing and applicable to the TOE in its operational environment.
- 12719 It may be identified that no further consideration of the potential vulnerability is required if for
12720 example the evaluator identifies that measures in the operational environment, either IT or non-IT,
12721 prevent exploitation of the potential vulnerability in that operational environment. For instance,
12722 restricting physical access to the TOE to authorised users only may effectively render a potential
12723 vulnerability to tampering unexploitable.
- 12724 The evaluator records any reasons for exclusion of potential vulnerabilities from further
12725 consideration if the evaluator determines that the potential vulnerability is not applicable in the
12726 operational environment. Otherwise the evaluator records the potential vulnerability for further
12727 consideration.
- 12728 A list of potential vulnerabilities applicable to the TOE in its operational environment, which can be
12729 used as an input into penetration testing activities, shall be reported in the ETR by the evaluators.
- 12730 **15.1.5.7 Action AVA_VAN.5.4E**
- 12731 **15.1.5.7.1 Work unit AVA_VAN.5-6**
- 12732 The evaluator *shall devise* penetration tests, based on the independent search for potential
12733 vulnerabilities.
- 12734 The evaluator prepares for penetration testing as necessary to determine the susceptibility of the
12735 TOE, in its operational environment, to the potential vulnerabilities identified during the search of
12736 the sources of publicly available information and the analysis of the TOE guidance and design
12737 evidence. The evaluator should have access to current information (e.g. from the evaluation
12738 authority) regarding known potential vulnerabilities that may not have been considered by the
12739 evaluator.
- 12740 The evaluator is reminded that, as for considering the security architecture description in the
12741 search for vulnerabilities (as detailed in AVA_VAN.5-3), testing should be performed to confirm the
12742 architectural properties. If requirements from ATE_DPT are included in the SARs, the developer
12743 testing evidence will include testing performed to confirm the correct implementation of any
12744 specific mechanisms detailed in the security architecture description. However, the developer
12745 testing will not necessarily include testing of all aspects of the architectural properties that protect
12746 the TSF, as much of this testing will be negative testing in nature, attempting to disprove the
12747 properties. In developing the strategy for penetration testing, the evaluator will ensure that all
12748 aspects of the security architecture description are tested, either in functional testing (as
12749 considered in 15,) or evaluator penetration testing.
- 12750 The evaluator will probably find it practical to carry out penetration test using a series of test cases,
12751 where each test case will test for a specific potential vulnerability.
- 12752 The evaluator is not expected to test for potential vulnerabilities (including those in the public
12753 domain) beyond those which required a **High** attack potential. In some cases, however, it will be
12754 necessary to carry out a test before the exploitability can be determined. Where, as a result of

- 12755 evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond **High**
12756 attack potential, this is reported in the ETR as a residual vulnerability.
- 12757 Guidance on determining the necessary attack potential to exploit a potential vulnerability can be
12758 found in Annex B.4.
- 12759 Potential vulnerabilities hypothesised as exploitable by an attacker possessing a **High** (or less)
12760 attack potential and resulting in a violation of the security objectives should be the highest priority
12761 potential vulnerabilities comprising the list used to direct penetration testing against the TOE.
- 12762 **15.1.5.7.2 Work unit AVA_VAN.5-7**
- 12763 The evaluator *shall produce* penetration test documentation for the tests based on the list of
12764 potential vulnerabilities in sufficient detail to enable the tests to be repeatable. The test
12765 documentation shall include:
- 12766 a) identification of the potential vulnerability the TOE is being tested for;
- 12767 b) instructions to connect and setup all required test equipment as required to conduct the
12768 penetration test;
- 12769 c) instructions to establish all penetration test prerequisite initial conditions;
- 12770 d) instructions to stimulate the TSF;
- 12771 e) instructions for observing the behaviour of the TSF;
- 12772 f) descriptions of all expected results and the necessary analysis to be performed on the
12773 observed behaviour for comparison against expected results;
- 12774 g) instructions to conclude the test and establish the necessary post-test state for the TOE.
- 12775 The evaluator prepares for penetration testing based on the list of potential vulnerabilities
12776 identified during the search of the public domain and the analysis of the evaluation evidence.
- 12777 The evaluator is not expected to determine the exploitability for potential vulnerabilities beyond
12778 those for which a **High** attack potential is required to effect an attack. However, as a result of
12779 evaluation expertise, the evaluator may discover a potential vulnerability that is exploitable only
12780 by an attacker with greater than **High** attack potential. Such vulnerabilities are to be reported in
12781 the ETR as residual vulnerabilities.
- 12782 With an understanding of the potential vulnerability, the evaluator determines the most feasible
12783 way to test for the TOE's susceptibility. Specifically the evaluator considers:
- 12784 the TSFI or other TOE interface that will be used to stimulate the TSF and observe responses (It is
12785 possible that the evaluator will need to use an interface to the TOE other than the TSFI to
12786 demonstrate properties of the TSF such as those described in the security architecture description
12787 (as required by ADV_ARC). It should be noted, that although these TOE interfaces provide a means
12788 of testing the TSF properties, they are not the subject of the test.);
- 12789 initial conditions that will need to exist for the test (i.e. any particular objects or subjects that will
12790 need to exist and security attributes they will need to have);
- 12791 special test equipment that will be required to either stimulate a TSFI or make observations of a
12792 TSFI;

12793 whether theoretical analysis should replace physical testing, particularly relevant where the
12794 results of an initial test can be extrapolated to demonstrate that repeated attempts of an attack are
12795 likely to succeed after a given number of attempts.

12796 The evaluator will probably find it practical to carry out penetration testing using a series of test
12797 cases, where each test case will test for a specific potential vulnerability.

12798 The intent of specifying this level of detail in the test documentation is to allow another evaluator
12799 to repeat the tests and obtain an equivalent result.

12800 **15.1.5.7.3 Work unit AVA_VAN.5-8**

12801 The evaluator *shall conduct* penetration testing.

12802 The evaluator uses the penetration test documentation resulting from work unit AVA_VAN.5-6 as a
12803 basis for executing penetration tests on the TOE, but this does not preclude the evaluator from
12804 performing additional ad hoc penetration tests. If required, the evaluator may devise ad hoc tests
12805 as a result of information learnt during penetration testing that, if performed by the evaluator, are
12806 to be recorded in the penetration test documentation. Such tests may be required to follow up
12807 unexpected results or observations, or to investigate potential vulnerabilities suggested to the
12808 evaluator during the pre-planned testing.

12809 Should penetration testing show that a hypothesised potential vulnerability does not exist, then the
12810 evaluator should determine whether or not the evaluator's own analysis was incorrect, or if
12811 evaluation deliverables are incorrect or incomplete.

12812 The evaluator is not expected to test for potential vulnerabilities (including those in the public
12813 domain) beyond those which required a **High** attack potential. In some cases, however, it will be
12814 necessary to carry out a test before the exploitability can be determined. Where, as a result of
12815 evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond **High**
12816 attack potential, this is reported in the ETR as a residual vulnerability.

12817 **15.1.5.7.4 Work unit AVA_VAN.5-9**

12818 The evaluator *shall record* the actual results of the penetration tests.

12819 While some specific details of the actual test results may be different from those expected (e.g. time
12820 and date fields in an audit record) the overall result should be identical. Any unexpected test
12821 results should be investigated. The impact on the evaluation should be stated and justified.

12822 **15.1.5.7.5 Work unit AVA_VAN.5-10**

12823 The evaluator *shall report* in the ETR the evaluator penetration testing effort, outlining the testing
12824 approach, configuration, depth and results.

12825 The penetration testing information reported in the ETR allows the evaluator to convey the overall
12826 penetration testing approach and effort expended on this sub-activity. The intent of providing this
12827 information is to give a meaningful overview of the evaluator's penetration testing effort. It is not
12828 intended that the information regarding penetration testing in the ETR be an exact reproduction of
12829 specific test steps or results of individual penetration tests. The intention is to provide enough
12830 detail to allow other evaluators and evaluation authorities to gain some insight about the
12831 penetration testing approach chosen, amount of penetration testing performed, TOE test
12832 configurations, and the overall results of the penetration testing activity.

12833 Information that would typically be found in the ETR section regarding evaluator penetration
12834 testing efforts is:

12835 TOE test configurations. The particular configurations of the TOE that were penetration tested;

12836 TSFI penetration tested. A brief listing of the TSFI and other TOE interfaces that were the focus of
 12837 the penetration testing;

12838 Verdict for the sub-activity. The overall judgement on the results of penetration testing.

12839 This list is by no means exhaustive and is only intended to provide some context as to the type of
 12840 information that should be present in the ETR concerning the penetration testing the evaluator
 12841 performed during the evaluation.

12842

12843 **15.1.5.7.6 Work unit AVA_VAN.5-11**

12844 The evaluator *shall examine* the results of all penetration testing to determine that the TOE, in its
 12845 operational environment, is resistant to an attacker possessing a **High** attack potential.

12846 If the results reveal that the TOE, in its operational environment, has vulnerabilities exploitable by
 12847 an attacker possessing an attack potential less than **or equal to** High, then this evaluator action
 12848 fails.

12849 This text was incorporated from a national scheme document (AIS34 from BSI). References within
 12850 that text to other scheme documents (such as AIS14, 19, 26) have been deleted but additional text
 12851 would be welcome where it might add to clarity

12852 The guidance in B.4 and the guidance for special technical areas that is relevant for the national
 12853 scheme should be used to determine the attack potential required to exploit a particular
 12854 vulnerability and whether it can therefore be exploited in the intended environment. It may not be
 12855 necessary for the attack potential to be calculated in every instance, only if there is some doubt as
 12856 to whether or not the vulnerability can be exploited by an attacker possessing an attack potential
 12857 less than **or equal to** High.

12858 **15.1.5.7.7 Work unit AVA_VAN.5-12**

12859 The evaluator *shall report* in the corresponding ETR-part all exploitable vulnerabilities and
 12860 residual vulnerabilities, detailing for each:

12861 a) its source (e.g. ISO/IEC 18045 activity being undertaken when it was conceived, known to
 12862 the evaluator, read in a publication);

12863 b) the SFR(s) not met;

12864 c) a description;

12865 d) whether it is exploitable in its operational environment or not (i.e. exploitable or
 12866 residual);

12867 e) the amount of time, level of expertise, level of knowledge of the TOE, level of opportunity
 12868 and the equipment required to perform the identified vulnerabilities, and the
 12869 corresponding values using the tables 3 and 4 of Annex B.4.

12870 **15.2 Composite vulnerability assessment (AVA_COMP)**

12871 The composite-specific work units defined in this chapter are intended to be integrated as
 12872 refinements to the evaluation activities of the AVA class listed in the following table. The other
 12873 activities of AVA class do not require composite-specific work units.

CC assurance family	Evaluation activity	Evaluation work unit	Composite-specific
---------------------	---------------------	----------------------	--------------------

			work unit
AVA_VAN	AVA_VAN.1.3E	AVA_VAN.1-5	AVA_COMP.1-1
	AVA_VAN.1.3E	AVA_VAN.1-6	AVA_COMP.1-2
	AVA_VAN.1.3E	AVA_VAN.1-7	AVA_COMP.1-2
	AVA_VAN.1.3E	AVA_VAN.1-8	AVA_COMP.1-2

12874 NB: If the level of the assurance requirement chosen is higher than those identified in this table, the
 12875 composite-specific work unit is also applicable.

12876 **15.2.1 Evaluation of sub-activity (AVA_COMP.1)**

12877 **15.2.1.1 Objectives**

12878 The aim of this activity is to determine the exploitability of flaws or weaknesses in the composite
 12879 TOE as a whole in the intended environment.

12880 **15.2.1.2 Application notes**

12881 This activity focuses exclusively on vulnerability assessment of the composite product as a whole
 12882 and represents merely partial efforts within the general approach being covered by the standard
 12883 assurance family of the class AVA: AVA_VAN. The results of the vulnerability assessment for the
 12884 underlying platform represented in the ETR_COMP can be reused under the following conditions:
 12885 they are up to date and all composite activities for correctness – ASE_COMP.1, ALC_COMP.1,
 12886 ADV_COMP.1 and ATE_COMP.1 – are finalised with the verdict PASS.

12887 **15.2.1.3 AVA_COMP.1.1E**

12888 The evaluator *shall conduct* penetration testing of the composite product as a whole building on
 12889 evaluator’s own vulnerability analysis, to ensure that the vulnerabilities being relevant for the
 12890 Composite-ST are not exploitable.

12891 **15.2.1.3.1 Work unit AVA_COMP.1-1**

12892 The evaluator shall examine the results of the vulnerability assessment for the underlying platform
 12893 to determine that they can be reused for the composite evaluation.

12894 The results of the vulnerability assessment for the underlying platform are usually represented in
 12895 the ETR_COMP. They can be reused if the following conditions are met: they are up to date and all
 12896 composite activities for correctness – ASE_COMP.1, ALC_COMP.1, ADV_COMP.1 and ATE_COMP.1 –
 12897 are finalised with the verdict PASS. The evaluator shall also consider the relevant determinations in
 12898 any Platform Certification Report.. It is noted that the platform itself could be a composite TOE.
 12899 This means also that the validity of each ETR for composition of the TOEs that compose the
 12900 platform TOE must be checked.

12901 When the validity of the ETRs for composition is checked, the necessity of checking the contents
 12902 depends on the application and user available TSFI. If the TSFI are available to the user or used by
 12903 the application, the content of the ETR must be checked. If not and formal platform TSFI are no
 12904 longer available as TSFI, the validity date of the ETR_COMP is sufficient.

12905 The result of this work unit shall be integrated to the result of AVA_VAN.1.3E/ AVA_VAN.1-5 (or the
 12906 equivalent higher components if a higher assurance level is selected).

12907 **15.2.1.3.2 Work unit AVA_COMP.1-2**

12908 The evaluator shall *specify, conduct and document* penetration testing of the composite product
12909 as a whole, using the standard approach of the assurance family AVA_VAN.

12910 If the correctness-related activities – ASE_COMP.1, ALC_COMP.1, ADV_COMP.1 and ATE_COMP.1 –
12911 are finalised with the verdict PASS and the certificate for the platform covers all security properties
12912 needed for the composite product, composing of the platform and the application must not create
12913 additional vulnerabilities of the platform.

12914 If the evaluator determined that composing of the platform and the application creates additional
12915 vulnerabilities of the platform²², a contradiction to the verdict PASS for the correctness activities
12916 has to be supposed or the certificate for the platform does not cover all security properties needed
12917 for the current composite product.

12918 The result of this work unit shall be integrated to the result of AVA_VAN.1.3E/ AVA_VAN.1-6,
12919 AVA_VAN.1-7, AVA_VAN.1-8 (or the equivalent higher components if a higher assurance level
12920 is selected).

12921 **16 Class ACO: Composition**12922 **16.1 Introduction**

12923 The goal of this activity is to determine whether the components can be integrated in a secure
12924 manner, as defined in the ST for the composed TOE. This is achieved through examination and
12925 testing of the interfaces between the components, supported by examination of the design of the
12926 components and the conduct of vulnerability analysis.

12927 **16.2 Application notes**

12928 The Reliance of dependent component (ACO_REL) family identifies where the dependent
12929 component is reliant upon IT in its operational environment (satisfied by a base component in the
12930 composed TOE evaluation) in order to provide its own security services. This reliance is identified
12931 in terms of the interfaces expected by the dependent component to be provided by the base
12932 component. Development evidence (ACO_DEV) then determines which interfaces of the base
12933 component were considered (as TSFI) during the component evaluation of the base component.

12934 It should be noted that Reliance of dependent component (ACO_REL) does not cover other
12935 evidence that may be needed to address the technical integration problem of composing
12936 components (e.g. descriptions of non-TSF interfaces of the operating system, rules for integration,
12937 etc.). This is outside the security assessment of the composition and is a functional composition
12938 issue.

12939 As part of Composed TOE testing (ACO_CTT) the evaluator will perform testing of the composed
12940 TOE SFRs at the composed TOE interfaces and of the interfaces of the base component relied upon
12941 by the dependent component to confirm they operate as specified. The subset selected will
12942 consider the possible effects of changes to the configuration/use of the base component as used in
12943 the composed TOE. These changes are identified from the configuration of the base component
12944 determined during the base component evaluation. The developer will provide test evidence for
12945 each of the base component interfaces (the requirements for coverage are consistent with those
12946 applied to the evaluation of the base component).

12947 Composition rationale (ACO_COR) requires the evaluator to determine whether the appropriate
12948 assurance measures have been applied to the base component, and whether the base component is
12949 being used in its evaluated configuration. This includes determination of whether all security
12950 functionality required by the dependent component was within the TSF of the base component.
12951 The Composition rationale (ACO_COR) requirement may be met through the production of

12952 evidence that each of these is demonstrated to be upheld. This evidence may be in the form of the
12953 security target and a public report of the component evaluation (e.g. certification report).

12954 If, on the other hand, one of the above have not been upheld, then it may be possible that an
12955 argument can be made as to why the assurance gained during an original evaluation is unaffected.
12956 If this is not possible then additional evaluation evidence for those aspects of the base component
12957 not covered may have to be provided. This material is then assessed in Development evidence
12958 (ACO_DEV).

12959 For example, it may be the case as described in the Interactions between entities (see Annex B.3,
12960 **Interactions between composed IT entities** in ISO/IEC 15408-3) that the dependent component
12961 requires the base component to provide more security functionality in the composed TOE than
12962 included in the base component evaluation. This would be determined during the application of the
12963 Reliance of dependent component (ACO_REL) and Development evidence (ACO_DEV) families. In
12964 this case the composition rationale evidence provided for Composition rationale (ACO_COR) would
12965 demonstrate that the assurance gained from the base component evaluation is unaffected. This
12966 may be achieved by means including:

12967 a) Performing a re-evaluation of the base component focusing on the evidence relating to
12968 the extended part of the TSF;

12969 b) Demonstrating that the extended part of the TSF cannot affect other portions of the TSF,
12970 and providing evidence that the extended part of the TSF provides the necessary security
12971 functionality.

12972 **16.3 Composition rationale (ACO_COR)**

12973 **16.3.1 Evaluation of sub-activity (ACO_COR.1)**

12974 **16.3.1.1 Input**

12975 The evaluation evidence for this sub-activity is:

12976 a) the composed ST;

12977 b) the composition rationale;

12978 c) the reliance information;

12979 d) the development information;

12980 e) unique identifier.

12981 **16.3.1.2 Action ACO_COR.1.1E**

12982 ISO/IEC 15408-3 ACO_COR.1.1C: *The composition rationale shall demonstrate that a level of*
12983 *assurance at least as high as that of the dependent component has been obtained for the support*
12984 *functionality of the base component, when the base component is configured as required to support*
12985 *the TSF of the dependent component.*

12986 **16.3.1.2.1 Work unit ACO_COR.1-1**

12987 The evaluator **shall examine** the correspondence analysis with the development information and
12988 the reliance information to identify the interfaces that are relied upon by the dependent
12989 component which are not detailed in the development information.

12990 The evaluator's goal in this work unit is two fold:

- 12991 a) to determine which interfaces relied upon by the dependent component have had the
12992 appropriate assurance measures applied.
- 12993 b) to determine that the assurance package applied to the base component during the base
12994 component evaluation contained either the same assurance requirements as those in the
12995 package applied to the dependent component during its' evaluation, or hierarchically
12996 higher assurance requirements.
- 12997 The evaluator may use the correspondence tracing in the development information developed
12998 during the Development evidence (ACO_DEV) activities (e.g. ACO_DEV.1-2, ACO_DEV.2-4,
12999 ACO_DEV.3-6) to help identify the interfaces identified in the reliance information that are not
13000 considered in the development information.
- 13001 The evaluator will record the SFR-enforcing interfaces described in the reliance information that
13002 are not included in the development information. These will provide input to ACO_COR.1-3 work
13003 unit, helping to identify the portions of the base component in which further assurance is required.
- 13004 If the both the base and dependent components were evaluated against the same assurance
13005 package, then the determination of whether the level of assurance in the portions within the base
13006 component evaluation is at least as high as that of the dependent component is trivial. If however,
13007 the assurance packages applied to the components during the component evaluations differ, the
13008 evaluator needs to determine that the assurance requirements applied to the base component are
13009 all hierarchically higher to the assurance requirements applied to the dependent component.
- 13010 **16.3.1.2.2 Work unit ACO_COR.1-2**
- 13011 The evaluator *shall examine* the composition rationale to determine, for those included base
13012 component interfaces on which the dependent TSF relies, whether the interface was considered
13013 during the evaluation of the base component.
- 13014 The ST, component public evaluation report (e.g. certification report) and guidance documents for
13015 the base component all provide information on the scope and boundary of the base component.
13016 The ST provides details of the logical scope and boundary of the composed TOE, allowing the
13017 evaluator to determine whether an interface relates to a portion of the product that was within the
13018 scope of the evaluation. The guidance documentation provides details of use of all interfaces for the
13019 composed TOE. Although the guidance documentation may include details of interfaces in the
13020 product that are not within the scope of the evaluation, any such interfaces should be identifiable,
13021 either from the scoping information in the ST or through a portion of the guidance that deals with
13022 the evaluated configuration. The public evaluation report may provide any additional constraints
13023 on the use of the composed TOE that are necessary.
- 13024 Therefore, the combination of these inputs allows the evaluator to determine whether an interface
13025 described in the composition rationale has the necessary assurance associated with it, or whether
13026 further assurance is required. The evaluator will record those interfaces of the base component for
13027 which additional assurance is required, for consideration during ACO_COR.1-3.
- 13028 **16.3.1.2.3 Work unit ACO_COR.1-3**
- 13029 The evaluator *shall examine* the composition rationale to determine that the necessary assurance
13030 measures have been applied to the base component.
- 13031 The evaluation verdicts, and resultant assurance, for the base component can be reused provided
13032 the same portions of the base component are used in the composed TOE and they are used in a
13033 consistent manner.
- 13034 In order to determine whether the necessary assurance measures have already been applied to the
13035 component, and the portions of the component for which assurance measures still need to be

13036 applied, the evaluator should use the output of the ACO_DEV.*.2E action and the work units
13037 ACO_COR.1-1 and ACO_COR.1-2:

13038 a) For those interfaces identified in the reliance information (Reliance of dependent
13039 component (ACO_REL)), but not discussed in development information (Development
13040 evidence (ACO_DEV)), additional information is required. (Identified in ACO_COR.1-1.)

13041 b) For those interfaces used inconsistently in the composed TOE from the base component
13042 (difference between the information provided in Development evidence (ACO_DEV) and
13043 Reliance of dependent component (ACO_REL) the impact of the differences in use need to
13044 be considered. (Identified in ACO_DEV.*.2E.)

13045 c) For those interfaces identified in composition rationale for which no assurance has
13046 previously been gained, additional information is required. (Identified in ACO_COR.1-2.)

13047 d) For those interfaces consistently described in the reliance information, composition
13048 rationale and the development information, no further action is required as the results
13049 from the base component evaluation can be re-used.

13050 The interfaces of the base component reported to be required by the reliance information but not
13051 included in the development information indicate the portions of the base component where
13052 further assurance is required. The interfaces identify the entry points into the base component.

13053 For those interfaces included in both the development information and reliance information, the
13054 evaluator is to determine whether the interfaces are being used in the composed TOE in a manner
13055 that is consistent with the base component evaluation. The method of use of the interface will be
13056 considered during the Development evidence (ACO_DEV) activities to determine that the use of the
13057 interface is consistent in both the base component and the composed TOE. The remaining
13058 consideration is the determination of whether the configurations of the base component and the
13059 composed TOE are consistent. To determine this, the evaluator will consider the guidance
13060 documentation of each to ensure they are consistent (see further guidance below regarding
13061 consistent guidance documentation). Any deviation in the documentation will be further analysed
13062 by the evaluation to determine the possible effects.

13063 For those interfaces that are consistently described in the reliance information and development
13064 information, and for which the guidance is consistent for the base component and the composed
13065 TOE, the required level of assurance has been provided.

13066 The following subsubclauses provide guidance on how to determine consistency between
13067 assurance gained in the base component, the evidence provided for the composed TOE, and the
13068 analysis performed by the evaluator in the instances where inconsistencies are identified.

13069 **16.3.1.2.3.1 Development**

13070 The reliance information identifies the interfaces in the dependent component that are to be
13071 matched by the base component. If an interface identified in the reliance information is not
13072 identified in the development information, then the composition rationale is to provide a
13073 justification of how the base component provides the required interfaces.

13074 If an interface identified in the reliance information is identified in the development information,
13075 but there are inconsistencies between the descriptions, further analysis is required. The evaluator
13076 identifies the differences in use of the base component as considered in the base component
13077 evaluation and the composed TOE evaluation. The evaluator will devise testing to be performed
13078 (during the conduct of Composed TOE testing (ACO_CTT)) to test the interface.

13079 The patch status of the base and dependent components as used in the composed TOE should be
13080 compared to the patch status of the components during the component evaluations. If any patches
13081 have been applied to the components, the composition rationale is to include details of the patches,

- 13082 including any potential impact to the SFRs of the evaluated component. The evaluator should
 13083 consider the details of the changes provided and verify the accuracy of the potential impact of the
 13084 change on the component SFRs. The evaluator should then consider whether the changes made by
 13085 the patch should be verified through testing, and will identify the necessary testing approach. The
 13086 testing may take the form of repeating the applicable evaluator/developer testing performed for
 13087 the component evaluation of the component or it may be necessary for the evaluator to devise new
 13088 tests to confirm the modified component.
- 13089 If any of the individual components have been the subject of assurance continuity activities since
 13090 the completion of the component evaluation, the evaluator will consider the changes assessed in
 13091 the assurance continuity activities during the independent vulnerability analysis activity for the
 13092 composed TOE (in Composition vulnerability analysis (ACO_VUL)).
- 13093 **16.3.1.2.3.2 Guidance**
- 13094 The guidance for the composed TOE is likely to make substantial reference out to the guidance for
 13095 the individual components. The minimal guidance expected to be necessary is the identification of
 13096 any ordering dependencies in the application of guidance for the dependent and base components,
 13097 particularly during the preparation (installation) of the composed TOE.
- 13098 In addition to the application of the Preparative procedures (AGD_PRE) and Operational user
 13099 guidance (AGD_OPE) families to the guidance for the composed TOE, it is necessary to analyse the
 13100 consistency between the guidance for the components and the composed TOE, to identify any
 13101 deviations.
- 13102 If the composed TOE guidance refers out to the base component and dependent component
 13103 guidance, then the consideration for consistency is limited to consistency between the guidance
 13104 documentation provided for each of the components (i.e. consistency between the base component
 13105 guidance and the dependent component guidance). However, if additional guidance is provided for
 13106 the composed TOE, to that provided for the components, greater analysis is required, as
 13107 consistency is also required between the guidance documentation for the components and
 13108 guidance documentation for the composed TOE.
- 13109 *Consistent* in this instance is understood to mean that either the guidance is the same or it places
 13110 additional constraints on the operation of the individual components when combined, in a similar
 13111 manner to *refinement* of functional/assurance components.
- 13112 With the information available (that used as input for Development evidence (ACO_DEV) or the
 13113 development aspects discussed above) the evaluator may be able to determine all possible impacts
 13114 of the deviation from the configuration of the base component specified in the component
 13115 evaluation. However, for high EALs (where evaluation of the base component included
 13116 requirements) it is possible that, unless detailed design abstractions for the base component are
 13117 delivered as part of the development information for the composed TOE, the possible impacts of
 13118 the modification to the guidance cannot be fully determined as the internals are unknown. In this
 13119 case the evaluator will report the residual risk of the analysis.
- 13120 These residual risks are to be included in any public evaluation report for the composed TOE.
- 13121 The evaluator will note these variances in the guidance for input into evaluator independent
 13122 testing activities (Composed TOE testing (ACO_CTT)).
- 13123 The guidance for the composed TOE may add to the guidance for the components, particularly in
 13124 terms of installation and the ordering of installation steps for the base component in relation to the
 13125 installation steps for the dependent component. The ordering of the steps for the installation of the
 13126 individual components should not change, however they may need to be interleaved. The evaluator
 13127 will examine this guidance to ensure that it still meets the requirement of the AGD_PRE activity
 13128 performed during the evaluations of the components.

13129 It may be the case that the reliance information identifies that interfaces of the base component, in
 13130 addition to those identified as TSFIs of the base component, are relied upon by the dependent
 13131 component are identified in the reliance information. It may be necessary for guidance to be
 13132 provided for the use of any such additional interfaces in the base component. Provided the
 13133 consumer of the composed TOE is to receive the guidance documentation for the base component,
 13134 then the results of the AGD_PRE and AGD_OPE verdicts for the base component can be reused for
 13135 those interfaces considered in the evaluation of the base component. However, for the additional
 13136 interfaces relied upon by the dependent component, the evaluator will need to determine that the
 13137 guidance documentation for the base component meets the requirements of AGD_PRE and
 13138 AGD_OPE, as applied in the base component evaluations.

13139 For those interfaces considered during the base component evaluation, and therefore, for which
 13140 assurance has already been gained, the evaluator will ensure that the guidance for the use of each
 13141 interface for the composed TOE is consistent with that provided for the base component. To
 13142 determine the guidance for the composed TOE is consistent with that for the base component, the
 13143 evaluator should perform a mapping for each interface to the guidance provided for both the
 13144 composed TOE and the base component. The evaluator then compares the guidance to determine
 13145 consistency.

13146 Examples of additional constraints provided in composed TOE guidance that would be considered
 13147 to be consistent with component guidance are (guidance for a component is given followed by an
 13148 example of guidance for a composed TOE that would be considered to provide additional
 13149 constraints):

13150 — Component: The password length must be set to a minimum of 8 characters length, including
 13151 alphabetic and numeric characters.

13152 — Composed TOE: The password length must be set to a minimum of 10 characters in length,
 13153 including alphabetic and numeric characters and *at least one of the following special characters:*
 13154 *(){}^<>-_*

13155 — NOTE: It would only be acceptable to increase the password length to [*integer > 8*] characters
 13156 while removing the mandate for the inclusion of both alphabetic and numeric characters for
 13157 the composed TOE, if the same or a higher metric was achieved for the strength rating (taking
 13158 into account the likelihood of the password being guessed).

13159 — Component: The following services are to be disabled in the registry settings: WWW
 13160 Publishing Service and ICDBReporter service.

13161 — Composed TOE: The following services are to be disabled in the registry settings: Publishing
 13162 Service, ICDBReporter service, Remote Procedure Call (RPC) Locator and Procedure Call (RPC)
 13163 Service.

13164 — Component: Select the following attributes to be included in the accounting log files: date, time,
 13165 type of event, subject identity and success/failure.

13166 — Composed TOE: Select the following attributes to be included in the accounting log files: date,
 13167 time, type of event, subject identity, success/failure, *event message and process thread*.

13168 If the guidance for the composed TOE deviates (is not a refinement) from that provided for the
 13169 base component, the evaluator will assess the potential risks of the modification to the guidance.
 13170 The evaluator will use the information available (including that provided in the public domain, the
 13171 architectural description of the base component in the public evaluation report (e.g. certification
 13172 report), the context of the guidance from the remainder of the guidance documentation) to identify
 13173 likely impact of the modification to the guidance on the SFRs of the composed TOE.

- 13174 If during the dependent component evaluation the trial installation used the base component to
 13175 satisfy the environment requirements of the dependent component this work unit for the
 13176 composed TOE is considered to be satisfied. If the base component was not used in satisfaction of
 13177 the work unit AGD_PRE.1-3 during the dependent component evaluation, the evaluator will apply
 13178 the user procedures provided for the composed TOE to prepare the composed TOE, in accordance
 13179 with the guidance specified in AGD_PRE.1-3. This will allow the evaluator to determine that the
 13180 preparative guidance provided for the composed TOE is sufficient to prepare the composed TOE
 13181 and its operational environment securely.
- 13182 **16.3.1.2.3.3 Life-cycle**
- 13183 **Delivery**
- 13184 If there is a different delivery mechanism used for the delivery of the composed TOE (i.e. the
 13185 components are not delivered to the consumer in accordance with the secure delivery procedures
 13186 defined and assessed during the evaluation of the components), the delivery procedures for the
 13187 composed TOE will require evaluation against the Delivery (ALC_DEL) requirements applied
 13188 during the components evaluations.
- 13189 The composed TOE may be delivered as an integrated product or may require the components to
 13190 be delivered separately.
- 13191 If the components are delivered separately, the results of the delivery of the base component and
 13192 dependent component are reused. The delivery of the base component is checked during the
 13193 evaluator trial installation of the dependent component, using the specified guidance and checking
 13194 the aspects of delivery that are the responsibility of the user, as described in the guidance
 13195 documentation for the base component.
- 13196 If the composed TOE is delivered as a new entity, then the method of delivery of that entity must be
 13197 considered in the composed TOE evaluation activities.
- 13198 The assessment of the delivery procedures for composed TOE items is to be performed in
 13199 accordance with the methodology for Delivery (ALC_DEL) as for any other [component] TOE,
 13200 ensuring any additional items (e.g. additional guidance documents for the composed TOE) are
 13201 considered in the delivery procedures.
- 13202 **CM Capabilities**
- 13203 The unique identification of the composed TOE is considered during the application of Evaluation
 13204 of sub-activity (ALC_CMC.1) and the items from which that composed TOE is comprised are
 13205 considered during the application of Evaluation of sub-activity (ALC_CMS.2).
- 13206 Although additional guidance may be produced for the composed TOE, the unique identification of
 13207 this guidance (considered as part of the unique identification of the composed TOE during
 13208 Evaluation of sub-activity (ALC_CMC.1)) is considered sufficient control of the guidance.
- 13209 The verdicts of the remaining (not considered above) Class ALC: Life-cycle support activities can be
 13210 reused from the base component evaluation, as no further development is performed during
 13211 integration of the composed TOE.
- 13212 There are no additional considerations for development security as the integration is assumed to
 13213 take place at either the consumer's site or, in the instance that the composed TOE is delivered as an
 13214 integrated product, at the site of the dependent component developer. Control at the consumer's
 13215 site is outside the consideration of ISO/IEC 15408. No additional requirements or guidance are
 13216 necessary if integration is at the same site as that for the dependent component, as all components
 13217 are considered to be configuration items for the composed TOE, and should therefore be
 13218 considered under the dependent component developer's security procedures anyway.

13219 Tools and techniques adopted during integration will be considered in the evidence provided by
13220 the dependent component developer. Any tools/techniques relevant to the base component will
13221 have been considered during the evaluation of the base component. For example, if the base
13222 component is delivered as source code and requires compilation by the consumer (e.g. dependent
13223 component developer who is performing integration) the compiler would have been specified and
13224 assessed, along with the appropriate arguments, during evaluation of the base component.

13225 There is no life-cycle definition applicable to the composed TOE, as no further development of
13226 items is taking place.

13227 The results of flaw remediation for a component are not applicable to the composed TOE. If flaw
13228 remediation is included in the assurance package for the composed TOE, then the Flaw
13229 remediation (ALC_FLR) requirements are to be applied during the composed TOE evaluation (as
13230 for any augmentation).

13231 **16.3.1.2.3.4 Tests**

13232 The composed TOE will have been tested during the conduct of the Class ATE: Tests activities for
13233 evaluation of the dependent component, as the configurations used for testing of the dependent
13234 component should have included the base component to satisfy the requirements for IT in the
13235 operational environment. If the base component was not used in the testing of the dependent
13236 component for the dependent component evaluation, or the configuration of either component
13237 varied from their evaluated configurations, then the developer testing performed for evaluation of
13238 the dependent component to satisfy the Class ATE: Tests requirements is to be repeated on the
13239 composed TOE.

13240 **16.4 Development evidence (ACO_DEV)**

13241 **16.4.1 Evaluation of sub-activity (ACO_DEV.1)**

13242 **16.4.1.1 Objectives**

13243 The objective of this sub-activity is to determine that the appropriate security functionality is
13244 provided by the base component to support the dependent component. This is achieved through
13245 examination of the interfaces of the base component to determine that they are consistent with the
13246 interfaces specified in the reliance information; those required by the dependent component.

13247 The description of the interfaces into the base component is to be provided at a level of detail
13248 consistent with Evaluation of sub-activity (ADV_FSP.2) although not all of the aspects necessary for
13249 satisfaction of Evaluation of sub-activity (ADV_FSP.2) are required for Evaluation of sub-activity
13250 (ACO_DEV.1), as once the interface has been identified and the purpose described the remaining
13251 detail of the interface specification can be reused from evaluation of the base component.

13252 **16.4.1.2 Input**

13253 The evaluation evidence for this sub-activity is:

- 13254 a) the composed ST;
- 13255 b) the development information;
- 13256 c) the reliance information.

13257 **16.4.1.3 Action ACO_DEV.1.1E**

13258 ISO/IEC 15408-3 ACO_DEV.1.1C: *The development information shall describe the purpose of each*
13259 *interface of the base component used in the composed TOE.*

- 13260 **16.4.1.3.1 Work unit ACO_DEV.1-1**
- 13261 The evaluator *shall examine* the development information to determine that it describes the
13262 purpose of each interface.
- 13263 The base component provides interfaces to support interaction with the dependent component in
13264 the provision of the dependent TSF. The purpose of each interface is to be described at the same
13265 level as the description of the interfaces to the dependent component TSF functionality, as would
13266 be provided between subsystems in the TOE design (Evaluation of sub-activity (ADV_TDS.1)). This
13267 description is to provide the reader with an understanding of how the base component provides
13268 the services required by the dependent component TSF.
- 13269 This work unit may be satisfied by the provision of the functional specification for the base
13270 component for those interfaces that are TSFIs of the base component.
- 13271 ISO/IEC 15408-3 ACO_DEV.1.2C: *The development information shall show correspondence between*
13272 *the interfaces, used in the composed TOE, of the base component and the dependent component to*
13273 *support the TSF of the dependent component.*
- 13274 **16.4.1.3.2 Work unit ACO_DEV.1-2**
- 13275 The evaluator *shall examine* the development information to determine the correspondence,
13276 between the interfaces of the base component and the interfaces on which the dependent
13277 component relies, is accurate.
- 13278 The correspondence between the interfaces of the base component and the interfaces on which the
13279 dependent component relies may take the form of a matrix or table. The interfaces that are relied
13280 upon by the dependent component are identified in the reliance information (as examined during
13281 Reliance of dependent component (ACO_REL) activity).
- 13282 There is, during this activity, no requirement to determine completeness of the coverage of
13283 interfaces that are relied upon by the dependent component, only that the correspondence is
13284 correct and ensuring that interfaces of the base component are mapped to interfaces required by
13285 the dependent component wherever possible. The completeness of the coverage is considered in
13286 Composition rationale (ACO_COR) activities.
- 13287 **16.4.1.4 Action ACO_DEV.1.2E**
- 13288 **16.4.1.4.1 Work unit ACO_DEV.1-3**
- 13289 The evaluator *shall examine* the development information and the reliance information to
13290 determine that the interfaces are described consistently.
- 13291 The evaluator's goal in this work unit is to determine that the interfaces described in the
13292 development information for the base component and the reliance information for the dependent
13293 component are represented consistently.
- 13294 **16.4.2 Evaluation of sub-activity (ACO_DEV.2)**
- 13295 **16.4.2.1 Objectives**
- 13296 The objective of this sub-activity is to determine that the appropriate security functionality is
13297 provided by the base component to support the dependent component. This is achieved through
13298 examination of the interfaces and associated security behaviour of the base component to
13299 determine that they are consistent with the interfaces specified in the reliance information; those
13300 required by the dependent component.

13301 **16.4.2.2 Input**

13302 The evaluation evidence for this sub-activity is:

- 13303 a) the composed ST;
- 13304 b) the development information;
- 13305 c) reliance information.

13306 **16.4.2.3 Action ACO_DEV.2.1E**

13307 ISO/IEC 15408-3 ACO_DEV.2.1C: *The development information shall describe the purpose and*
13308 *method of use of each interface of the base component used in the composed TOE.*

13309 **16.4.2.3.1 Work unit ACO_DEV.2-1**

13310 The evaluator **shall examine** the development information to determine that it describes the
13311 purpose of each interface.

13312 The base component provides interfaces to support interaction with the dependent component in
13313 the provision of the dependent TSF. The purpose of each interface is to be described at the same
13314 level as the description of the interfaces to the dependent component TSF functionality, as would
13315 be provided between subsystems in the TOE design (Evaluation of sub-activity (ADV_TDS.1)). This
13316 description is to provide the reader with an understanding of how the base component provides
13317 the services required by the dependent component TSF.

13318 This work unit may be satisfied by the provision of the functional specification for the base
13319 component for those interfaces that are TSFIs of the base component.

13320 **16.4.2.3.2 Work unit ACO_DEV.2-2**

13321 The evaluator **shall examine** the development information to determine that it describes the
13322 method of use for each interface.

13323 The method of use for an interface summarises how the interface is manipulated in order to invoke
13324 the operations and obtain results associated with the interface. The evaluator should be able to
13325 determine from reading this material in the development information how to use each interface.
13326 This does not necessarily mean that there needs to be a separate method of use for each interface,
13327 as it may be possible to describe in general how APIs are invoked, for instance, and then identify
13328 each interface using that general style.

13329 This work unit may be satisfied by the provision of the functional specification for the base
13330 component for those interfaces that are TSFIs of the base component.

13331 ISO/IEC 15408-3 ACO_DEV.2.2C: *The development information shall provide a high-level description*
13332 *of the behaviour of the base component, which supports the enforcement of the dependent component*
13333 *SFRs.*

13334 **16.4.2.3.3 Work unit ACO_DEV.2-3**

13335 The evaluator **shall examine** the development information to determine that it describes the
13336 behaviour of the base component that supports the enforcement of the dependent component SFRs.

13337 The dependent component invokes interfaces of the base component for the provision of services
13338 by the base component. For the interfaces of the base component that are invoked, the
13339 development information shall provide a high-level description of the associated security
13340 behaviour of the base component. The description of the base component security behaviour will

13341 outline how the base component provides the necessary service when the call to the interface is
 13342 made. This description is to be at a level similar to that provided for ADV_TDS.1.4C. Therefore, the
 13343 provision of the TOE design evidence from the base component evaluation would satisfy this work
 13344 unit, where the interfaces invoked by the dependent component are TSFI of the base component. If
 13345 the interfaces invoked by the dependent component are not TSFIs of the base component it is the
 13346 associated security behaviour will not necessarily be described in the base component TOE design
 13347 evidence.

13348 ISO/IEC 15408-3 ACO_DEV.2.3C: *The development information shall show correspondence between*
 13349 *the interfaces, used in the composed TOE, of the base component and the dependent component to*
 13350 *support the TSF of the dependent component.*

13351 **16.4.2.3.4 Work unit ACO_DEV.2-4**

13352 The evaluator *shall examine* the development information to determine the correspondence,
 13353 between the interfaces of the base component and the interfaces on which the dependent
 13354 component relies, is accurate.

13355 The correspondence between the interfaces of the base component and the interfaces on which the
 13356 dependent component relies may take the form of a matrix or table. The interfaces that are relied
 13357 upon by the dependent component are identified in the reliance information (as examined during
 13358 Reliance of dependent component (ACO_REL)).

13359 There is, during this activity, no requirement to determine completeness of the coverage of
 13360 interfaces that are relied upon by the dependent component, only that the correspondence is
 13361 correct and ensuring that interfaces of the base component are mapped to interfaces required by
 13362 the dependent component wherever possible. The completeness of the coverage is considered in
 13363 Composition rationale (ACO_COR) activities.

13364 **16.4.2.4 Action ACO_DEV.2.2E**

13365 **16.4.2.4.1 Work unit ACO_DEV.2-5**

13366 The evaluator *shall examine* the development information and the reliance information to
 13367 determine that the interfaces are described consistently.

13368 The evaluator's goal in this work unit is to determine that the interfaces described in the
 13369 development information for the base component and the reliance information for the dependent
 13370 component are represented consistently.

13371 **16.4.3 Evaluation of sub-activity (ACO_DEV.3)**

13372 **16.4.3.1 Objectives**

13373 The objective of this sub-activity is to determine that the appropriate security functionality is
 13374 provided by the base component to support the dependent component. This is achieved through
 13375 examination of the interfaces and associated security behaviour of the base component to
 13376 determine that they are consistent with the interfaces specified in the reliance information; those
 13377 required by the dependent component.

13378 In addition to the interface description, the subsystems of the base component that provide the
 13379 security functionality required by the dependent component will be described to enable the
 13380 evaluator to determine whether or not that interface formed part of the TSF of the base component.

13381 **16.4.3.2 Input**

13382 The evaluation evidence for this sub-activity is:

13383 a) the composed ST;

13384 b) the development information;

13385 c) reliance information.

13386 **16.4.3.3 Action ACO_DEV.3.1E**

13387 ISO/IEC 15408-3 ACO_DEV.3.1C: *The development information shall describe the purpose and*
13388 *method of use of each interface of the base component used in the composed TOE.*

13389 **16.4.3.3.1 Work unit ACO_DEV.3-1**

13390 The evaluator **shall examine** the development information to determine that it describes the
13391 purpose of each interface.

13392 The base component provides interfaces to support interaction with the dependent component in
13393 the provision of the dependent TSF. The purpose of each interface is to be described at the same
13394 level as the description of the interfaces to the dependent component TSF functionality, as would
13395 be provided between subsystems in the TOE design (Evaluation of sub-activity (ADV_TDS.1)). This
13396 description is to provide the reader with an understanding of how the base component provides
13397 the services required by the dependent component TSF.

13398 This work unit may be satisfied by the provision of the functional specification for the base
13399 component for those interfaces that are TSFIs of the base component.

13400 **16.4.3.3.2 Work unit ACO_DEV.3-2**

13401 The evaluator **shall examine** the development information to determine that it describes the
13402 method of use for each interface.

13403 The method of use for an interface summarises how the interface is manipulated in order to invoke
13404 the operations and obtain results associated with the interface. The evaluator should be able to
13405 determine from reading this material in the development information how to use each interface.
13406 This does not necessarily mean that there needs to be a separate method of use for each interface,
13407 as it may be possible to describe in general how APIs are invoked, for instance, and then identify
13408 each interface using that general style.

13409 This work unit may be satisfied by the provision of the functional specification for the base
13410 component for those interfaces that are TSFIs of the base component.

13411 ISO/IEC 15408-3 ACO_DEV.3.2C: *The development information shall identify the subsystems of the*
13412 *base component that provide interfaces of the base component used in the composed TOE.*

13413 **16.4.3.3.3 Work unit ACO_DEV.3-3**

13414 The evaluator **shall examine** the development information to determine that all subsystems of the
13415 base component that provide interfaces to the dependent component are identified.

13416 For those interfaces that are considered to form part of the TSFI of the base component, the
13417 subsystems associated with the interface will be subsystems considered in the TOE design
13418 (ADV_TDS) activity during the base component evaluation. The interfaces on which the dependent
13419 component relies that did not form part of the TSFI of the base component will map to subsystems
13420 outside of the base component TSF.

13421 ISO/IEC 15408-3 ACO_DEV.3.3C: *The development information shall provide a high-level description*
13422 *of the behaviour of the base component subsystems, which support the enforcement of the dependent*
13423 *component SFRs.*

13424 **16.4.3.3.4 Work unit ACO_DEV.3-4**

13425 The evaluator **shall examine** the development information to determine that it describes the
 13426 behaviour of the base component subsystems that support the enforcement of the dependent
 13427 component SFRs.

13428 The dependent component invokes interfaces of the base component for the provision of services
 13429 by the base component. For the interfaces of the base component that are invoked, the
 13430 development information shall provide a high-level description of the associated security
 13431 behaviour of the base component. The description of the base component security behaviour will
 13432 outline how the base component provides the necessary service when the call to the interface is
 13433 made. This description is to be at a level similar to that provided for ADV_TDS.1.4C. Therefore, the
 13434 provision of the TOE design evidence from the base component evaluation would satisfy this work
 13435 unit, where the interfaces invoked by the dependent component are TSFI of the base component. If
 13436 the interfaces invoked by the dependent component are not TSFIs of the base component it is the
 13437 associated security behaviour will not necessarily be described in the base component TOE design
 13438 evidence.

13439 ISO/IEC 15408-3 ACO_DEV.3.4C: *The development information shall provide a mapping from the*
 13440 *interfaces to the subsystems of the base component.*

13441 **16.4.3.3.5 Work unit ACO_DEV.3-5**

13442 The evaluator **shall examine** the development information to determine that the correspondence
 13443 between the interfaces and subsystems of the base component is accurate.

13444 If the TOE design and functional specification evidence from the base component evaluation is
 13445 available, this can be used to verify the accuracy of the correspondence between the interfaces and
 13446 subsystems of the base component as used in the composed TOE. Those interfaces of the base
 13447 component, which formed part of the base component TSFI will be described in the base
 13448 component functional specification, and the associated subsystems will be described in the base
 13449 component TOE design evidence. The tracing between the two will be provided in the base
 13450 component TOE design evidence.

13451 If, however, the base component interface did not form part of the TSFI of the base component, the
 13452 description of the subsystem behaviour provided in the development information will be used to
 13453 verify the accuracy of the correspondence.

13454 ISO/IEC 15408-3 ACO_DEV.3.5C: *The development information shall show correspondence between*
 13455 *the interfaces, used in the composed TOE, of the base component and the dependent component to*
 13456 *support the TSF of the dependent component.*

13457 **16.4.3.3.6 Work unit ACO_DEV.3-6**

13458 The evaluator **shall examine** the development information to determine the correspondence,
 13459 between the interfaces of the base component and the interfaces on which the dependent
 13460 component relies, is accurate.

13461 The correspondence between the interfaces of the base component and the interfaces on which the
 13462 dependent component relies may take the form of a matrix or table. The interfaces that are relied
 13463 upon by the dependent component are identified in the reliance information (as examined during
 13464 Reliance of dependent component (ACO_REL)).

13465 There is, during this activity, no requirement to determine completeness of the coverage of
 13466 interfaces that are relied upon by the dependent component, only that the correspondence is
 13467 correct and ensuring that interfaces of the base component are mapped to interfaces required by
 13468 the dependent component wherever possible. The completeness of the coverage is considered in
 13469 Composition rationale (ACO_COR) activities.

13470 **16.4.3.4 Action ACO_DEV.3.2E**

13471 **16.4.3.4.1 Work unit ACO_DEV.3-7**

13472 The evaluator *shall examine* the development information and the reliance information to
13473 determine that the interfaces are described consistently.

13474 The evaluator's goal in this work unit is to determine that the interfaces described in the
13475 development information for the base component and the reliance information for the dependent
13476 component are represented consistently.

13477 **16.5 Reliance of dependent component (ACO_REL)**

13478 **16.5.1 Evaluation of sub-activity (ACO_REL.1)**

13479 **16.5.1.1 Objectives**

13480 The objectives of this sub-activity are to determine whether the developer's reliance evidence
13481 provides sufficient information to determine that the necessary functionality is available in the
13482 base component, and the means by which that functionality is invoked. These are provided in
13483 terms of a high-level description.

13484 **16.5.1.2 Input**

13485 The evaluation evidence for this sub-activity is:

- 13486 a) the composed ST;
- 13487 b) the dependent component functional specification;
- 13488 c) the dependent component design;
- 13489 d) the dependent component architectural design;
- 13490 e) the reliance information.

13491 **16.5.1.3 Application notes**

13492 A dependent component whose TSF interacts with the base component requires functionality
13493 provided by that base component (e.g., remote authentication, remote audit data storage). In these
13494 cases, those invoked services need to be described for those charged with configuring the
13495 composed TOE for end users. The rationale for requiring this documentation is to aid integrators of
13496 the composed TOE to determine what services in the base component might have adverse effects
13497 on the dependent component, and to provide information against which to determine the
13498 compatibility of the components when applying the Development evidence (ACO_DEV) family.

13499 **16.5.1.4 Action ACO_REL.1.1E**

13500 ISO/IEC 15408-3 ACO_REL.1.1C: *The reliance information shall describe the functionality of the base*
13501 *component hardware, firmware and/or software that is relied upon by the dependent component TSF.*

13502 **16.5.1.4.1 Work unit ACO_REL.1-1**

13503 The evaluator *shall check* the reliance information to determine that it describes the functionality
13504 of the base dependent hardware, firmware and/or software that is relied upon by the dependent
13505 component TSF.

- 13506 The evaluator assesses the description of the security functionality that the dependent component
 13507 TSF requires to be provided by the base component's hardware, firmware and software. The
 13508 emphasis of this work unit is on the level of detail of this description, rather than on an assessment
 13509 of the information's accuracy. (The assessment of the accuracy of the information is the focus of the
 13510 next work unit.)
- 13511 This description of the base component's functionality need not be any more detailed than the level
 13512 of the description of a component of the TSF, as would be provided in the TOE Design (TOE design
 13513 (ADV_TDS))
- 13514 **16.5.1.4.2 Work unit ACO_REL.1-2**
- 13515 The evaluator *shall examine* the reliance information to determine that it accurately reflects the
 13516 objectives specified for the operational environment of the dependent component.
- 13517 The reliance information contains the description of the base component's security functionality
 13518 relied upon by the dependent component. To ensure that the reliance information is consistent
 13519 with the expectations of the operational environment of the dependent component, the evaluator
 13520 compares the reliance information with the statement of objectives for the environment in the ST
 13521 for the dependent component.
- 13522 For example, if the reliance information claims that the dependent component TSF relies upon the
 13523 base component to store and protect audit data, yet other evaluation evidence (e.g. the dependent
 13524 component design) makes it clear that the dependent component TSF itself is storing and
 13525 protecting the audit data, this would indicate an inaccuracy.
- 13526 It should be noted that the objectives for the operational environment may include objectives that
 13527 can be met by non-IT measures. While the services that the base component environment is
 13528 expected to provide may be described in the description of IT objectives for the operational
 13529 environment in the dependent component ST, it is not required that all such expectations on the
 13530 environment be described in the reliance information.
- 13531 ISO/IEC 15408-3 ACO_REL.1.2C: *The reliance information shall describe all interactions through*
 13532 *which the dependent component TSF requests services from the base component.*
- 13533 **16.5.1.4.3 Work unit ACO_REL.1-3**
- 13534 The evaluator *shall examine* the reliance information to determine that it describes all
 13535 interactions between the dependent component and the base component, through which the
 13536 dependent component TSF requests services from the base component.
- 13537 The dependent component TSF may request services of the base component that were not within
 13538 the TSF of the base component (see **B.3, Interactions between composed IT entities** in ISO/IEC
 13539 15408-3).
- 13540 The interfaces to the base component's functionality are described at the same level as the
 13541 description of the interfaces to the dependent component TSF functionality, as would be provided
 13542 between subsystems in the TOE design (Evaluation of sub-activity (ADV_TDS.1)).
- 13543 The purpose of describing the interactions between the dependent component and the base
 13544 component is to provide an understanding of how the dependent component TSF relies upon the
 13545 base component for the provision of services to support the operation of security functionality of
 13546 the dependent component. These interactions do not need to be characterised at the
 13547 implementation level (e.g. parameters passed from one routine in a component to a routine in
 13548 another component), but the data elements identified for a particular component that are going to
 13549 be used by another component should be covered in this description. The statement should help
 13550 the reader understand in general why the interaction is necessary.

13551 Accuracy and completeness of the interfaces is based on the security functionality that the TSF
13552 requires to be provided by the base component, as assessed in work units ACO_REL.1-1 and
13553 ACO_REL.1-2. It should be possible to map all of the functionality described in the earlier work
13554 units to the interfaces identified in this work unit, and vice versa. An interface that does not
13555 correspond to described functionality would also indicate an inadequacy.

13556 ISO/IEC 15408-3 ACO_REL.1.3C: *The reliance information shall describe how the dependent TSF*
13557 *protects itself from interference and tampering by the base component.*

13558 **16.5.1.4.4 Work unit ACO_REL.1-4**

13559 The evaluator **shall examine** the reliance information to determine that it describes how the
13560 dependent TSF protects itself from interference and tampering by the base component.

13561 The description of how the dependent component protects itself from interference and tampering
13562 by the base component is to be provided at the same level of detail as necessary for ADV_ARC.1-4.

13563 **16.5.2 Evaluation of sub-activity (ACO_REL.2)**

13564 **16.5.2.1 Objectives**

13565 The objectives of this sub-activity are to determine whether the developer's reliance evidence
13566 provides sufficient information to determine that the necessary functionality is available in the
13567 base component, and the means by which that functionality is invoked. This is provided in terms of
13568 the interfaces between the dependent and base component and the return values from those
13569 interfaces called by the dependent component.

13570 **16.5.2.2 Input**

13571 The evaluation evidence for this sub-activity is:

- 13572 a) the composed ST;
- 13573 b) the dependent component functional specification;
- 13574 c) the dependent component design;
- 13575 d) the dependent component implementation representation;
- 13576 e) the dependent component architectural design;
- 13577 f) the reliance information.

13578 **16.5.2.3 Application notes**

13579 A dependent component whose TSF interacts with the base component requires functionality
13580 provided by that base component (e.g., remote authentication, remote audit data storage). In these
13581 cases, those invoked services need to be described for those charged with configuring the
13582 composed TOE for end users. The rationale for requiring this documentation is to aid integrators of
13583 the composed TOE to determine what services in the base component might have adverse effects
13584 on the dependent component, and to provide information against which to determine the
13585 compatibility of the components when applying the Development evidence (ACO_DEV) family.

13586 **16.5.2.4 Action ACO_REL.2.1E**

13587 ISO/IEC 15408-3 ACO_REL.2.1C: *The reliance information shall describe the functionality of the base*
13588 *component hardware, firmware and/or software that is relied upon by the dependent component TSF.*

13589 **16.5.2.4.1 Work unit ACO_REL.2-1**

13590 The evaluator **shall check** the reliance information to determine that it describes the functionality
13591 of the base dependent hardware, firmware and/or software that is relied upon by the dependent
13592 component TSF.

13593 The evaluator assesses the description of the security functionality that the dependent component
13594 TSF requires to be provided by the base component's hardware, firmware and software. The
13595 emphasis of this work unit is on the level of detail of this description, rather than on an assessment
13596 of the information's accuracy. (The assessment of the accuracy of the information is the focus of the
13597 next work unit.)

13598 This description of the base component's functionality need not be any more detailed than the level
13599 of the description of a component of the TSF, as would be provided in the TOE Design (TOE design
13600 (ADV_TDS))

13601 **16.5.2.4.2 Work unit ACO_REL.2-2**

13602 The evaluator **shall examine** the reliance information to determine that it accurately reflects the
13603 objectives specified for the operational environment of the dependent component.

13604 The reliance information contains the description of the base component's security functionality
13605 relied upon by the dependent component. To ensure that the reliance information is consistent
13606 with the expectations of the operational environment of the dependent component, the evaluator
13607 compares the reliance information with the statement of objectives for the environment in the ST
13608 for the dependent component.

13609 For example, if the reliance information claims that the dependent component TSF relies upon the
13610 base component to store and protect audit data, yet other evaluation evidence (e.g. the dependent
13611 component design) makes it clear that the dependent component TSF itself is storing and
13612 protecting the audit data, this would indicate an inaccuracy.

13613 It should be noted that the objectives for the operational environment may include objectives that
13614 can be met by non-IT measures. While the services that the base component environment is
13615 expected to provide may be described in the description of IT objectives for the operational
13616 environment in the dependent component ST, it is not required that all such expectations on the
13617 environment be described in the reliance information.

13618 ISO/IEC 15408-3 ACO_REL.2.2C: *The reliance information shall describe all interactions through
13619 which the dependent component TSF requests services from the base component.*

13620 **16.5.2.4.3 Work unit ACO_REL.2-3**

13621 The evaluator **shall examine** the reliance information to determine that it describes all
13622 interactions between the dependent component and the base component, through which the
13623 dependent component TSF requests services from the base component.

13624 The dependent component TSF may request services of the base component that were not within
13625 the TSF of the base component (see Annex B.3, **Interactions between composed IT entities** in
13626 ISO/IEC 15408-3).

13627 The interfaces to the base component's functionality are described at the same level as the
13628 description of the interfaces to the dependent component TSF functionality, as would be provided
13629 between subsystems in the TOE design (Evaluation of sub-activity (ADV_TDS.1)).

13630 The purpose of describing the interactions between the dependent component and the base
13631 component is to provide an understanding of how the dependent component TSF relies upon the
13632 base component for the provision of services to support the operation of security functionality of

13633 the dependent component. These interactions do not need to be characterised at the
13634 implementation level (e.g. parameters passed from one routine in a component to a routine in
13635 another component), but the data elements identified for a particular component that are going to
13636 be used by another component should be covered in this description. The statement should help
13637 the reader understand in general why the interaction is necessary.

13638 Accuracy and completeness of the interfaces is based on the security functionality that the TSF
13639 requires to be provided by the base component, as assessed in work units ACO_REL.2-1 and
13640 ACO_REL.2-2. It should be possible to map all of the functionality described in the earlier work
13641 units to the interfaces identified in this work unit, and vice versa. An interface that does not
13642 correspond to described functionality would also indicate an inadequacy.

13643 ISO/IEC 15408-3 ACO_REL.2.3C: *The reliance information shall describe each interaction in terms of*
13644 *the interface used and the return values from those interfaces.*

13645 **16.5.2.4.4 Work unit ACO_REL.2-4**

13646 The reliance information shall describe each interaction in terms of the interface used and the
13647 return values from those interfaces.

13648 The identification of the interfaces used by the dependent component TSF when making services
13649 requests of the base component allows an integrator to determine whether the base component
13650 provides all the necessary corresponding interfaces. This understanding is further gained through
13651 the specification of the return values expected by the dependent component. The evaluator ensures
13652 that interfaces are described for each interaction specified (as analysed in ACO_REL.2-3).

13653 ISO/IEC 15408-3 ACO_REL.2.4C: *The reliance information shall describe how the dependent TSF*
13654 *protects itself from interference and tampering by the base component.*

13655 **16.5.2.4.5 Work unit ACO_REL.2-5**

13656 The evaluator *shall examine* the reliance information to determine that it describes how the
13657 dependent TSF protects itself from interference and tampering by the base component.

13658 The description of how the dependent component protects itself from interference and tampering
13659 by the base component is to be provided at the same level of detail as necessary for ADV_ARC.1-4.

13660 **16.6 Composed TOE testing (ACO_CTT)**

13661 **16.6.1 Evaluation of sub-activity (ACO_CTT.1)**

13662 **16.6.1.1 Objectives**

13663 The objective of this sub-activity is to determine whether the developer correctly performed and
13664 documented tests for each of the base component interfaces on which the dependent component
13665 relies. As part of this determination the evaluator repeats a sample of the tests performed by the
13666 developer and performs any additional tests required to ensure the expected behaviour of all
13667 composed TOE SFRs and interfaces of the base component relied upon by the dependent
13668 component is demonstrated.

13669 **16.6.1.2 Input**

13670 The evaluation evidence for this sub-activity is:

- 13671 a) the composed TOE suitable for testing;
- 13672 b) the composed TOE testing evidence;

- 13673 c) the reliance information;
- 13674 d) the development information.
- 13675 **16.6.1.3 Action ACO_CTT.1.1E**
- 13676 ISO/IEC 15408-3 ACO_CTT.1.1C: *The composed TOE and base component interface test*
13677 *documentation shall consist of test plans, expected test results and actual test results.*
- 13678 **16.6.1.3.1 Work unit ACO_CTT.1-1**
- 13679 The evaluator ***shall examine*** the composed TOE test documentation to determine that it consists
13680 of test plans, expected test results and actual test results.
- 13681 This work unit may be satisfied by provision of the test evidence from the evaluation of the
13682 dependent component if the base component was used to satisfy the requirements for IT in the
13683 operational environment of the dependent component.
- 13684 All work units necessary for the satisfaction of ATE_FUN.1.1E will be applied to determine:
- 13685 a) that the test documentation consist of test plans expected test results and actual test
13686 results;
- 13687 b) that the test documentation contains the information necessary to ensure the tests are
13688 repeatable;
- 13689 c) the level of developer effort that was applied to testing of the base component.
- 13690 **16.6.1.3.2 Work unit ACO_CTT.1-2**
- 13691 The evaluator ***shall examine*** the base component interface test documentation to determine that it
13692 consists of test plans, expected test results and actual test results.
- 13693 This work unit may be satisfied by provision of the test evidence from the evaluation of the base
13694 component for those interfaces relied upon in the composed TOE by the dependent component are
13695 TSFIs of the successfully evaluated base component. The determination of whether the interfaces
13696 of the base component relied upon by the dependent component were in fact TSFIs of the
13697 evaluated base component is made during the ACO_COR activity.
- 13698 All work units necessary for the satisfaction of ATE_FUN.1.1E will be applied to determine:
- 13699 a) that the test documentation consist of test plans expected test results and actual test
13700 results;
- 13701 b) that the test documentation contains the information necessary to ensure the tests are
13702 repeatable;
- 13703 c) the level of developer effort that was applied to testing of the base component.
- 13704 ISO/IEC 15408-3 ACO_CTT.1.2C: *The test documentation from the developer execution of the*
13705 *composed TOE tests shall demonstrate that the TSF behaves as specified.*
- 13706 **16.6.1.3.3 Work unit ACO_CTT.1-3**
- 13707 The evaluator ***shall examine*** the test documentation to determine that the developer execution of
13708 the composed TOE tests shall demonstrate that the TSF behaves as specified.

13709 The evaluator should construct a mapping between the tests described in the test plan and the
13710 SFRs specified for the composed TOE to identify which SFRs have been tested by the developer.

13711 Guidance on this work unit can be found in:

13712 a) Clause 14.2.1.

13713 b) Clause 14.2.2.

13714 The outputs from the successful execution of the tests (as assessed for ATE_FUN.1.3C can be
13715 compared with the mapping to determine that the SFRs of the composed TOE, as tested by the
13716 developer, behave as expected.

13717 ISO/IEC 15408-3 ACO_CTT.1.3C: *The test documentation from the developer execution of the base*
13718 *component interface tests shall demonstrate that the base component interface relied upon by the*
13719 *dependent component behaves as specified.*

13720 **16.6.1.3.4 Work unit ACO_CTT.1-4**

13721 The evaluator **shall examine** the test documentation to determine that the developer execution of
13722 the base component interface tests shall demonstrate that the base component interfaces relied
13723 upon by the dependent component behave as specified.

13724 The evaluator should construct a mapping between the tests described in the test plan and the
13725 interfaces of the base component relied upon by the dependent component (as specified in the
13726 reliance information, examined under ACO_REL) to identify which base component interfaces have
13727 been tested by the developer.

13728 Guidance on this work unit can be found in:

13729 a) Clause 14.2.1.

13730 b) Clause 14.2.2.

13731 The outputs from the successful execution of the tests (as assessed for ATE_FUN.1.3C can be
13732 compared with the mapping to determine that the interfaces of the base component, as tested by
13733 the developer, behave as expected.

13734 ISO/IEC 15408-3 ACO_CTT.1.4C: *The base component shall be suitable for testing.*

13735 **16.6.1.3.5 Work unit ACO_CTT.1-5**

13736 The evaluator **shall examine** the composed TOE to determine that it has been installed properly
13737 and is in a known state.

13738 To determine that the composed TOE has been installed properly and is in a known state the
13739 ATE_IND.2-1 and ATE_IND.2-2 work units will be applied to the TOE provided by the developer for
13740 testing.

13741 **16.6.1.3.6 Work unit ACO_CTT.1-6**

13742 The evaluator **shall examine** the set of resources provided by the developer to determine that they
13743 are equivalent to the set of resources used by the base component developer to functionally test
13744 the base component.

13745 To determine that the set of resources provided are equivalent to those used to functionally test
13746 the base component as used in the composed TOE, the ATE_IND.2-3 work unit will be applied.

13747 **16.6.1.4 Action ACO_CTT.1.2E**13748 **16.6.1.4.1 Work unit ACO_CTT.1-7**

13749 The evaluator *shall perform* testing in accordance with ATE_IND.2.2E, for a subset of the SFRs
13750 specified in the composed security target, to verify the developer test results.

13751 The evaluator will apply all work units necessary for the satisfaction of ATE_IND.2.2E, reporting in
13752 the ETR for the composed TOE all analysis, results and verdicts as dictated by the associated work
13753 units.

13754 **16.6.1.5 Action ACO_CTT.1.3E**13755 **16.6.1.5.1 Work unit ACO_CTT.1-8**

13756 The evaluator *shall perform* testing in accordance with ATE_IND.2.3E, for a subset of the SFRs
13757 specified in the composed security target, to confirm that the TSF operates as specified.

13758 The evaluator will apply all work units necessary for the satisfaction of ATE_IND.2.3E, reporting in
13759 the ETR for the composed TOE all analysis, results and verdicts as dictated by the work units.

13760 When selecting interfaces of the TSF of the composed TOE to test, the evaluator should take into
13761 account any modifications to the components from the evaluated version or configuration.
13762 Modifications to the component from that evaluated may include patches introduced, a different
13763 configuration as a result of modified guidance documentation, reliance an additional portion of the
13764 component that was not within the TSF of the component. These modifications will have been
13765 identified during the Composition rationale (ACO_COR) activity.

13766 **16.6.2 Evaluation of sub-activity (ACO_CTT.2)**13767 **16.6.2.1 Objectives**

13768 The objective of this sub-activity is to determine whether the developer correctly performed and
13769 documented tests for each of the base component interfaces on which the dependent component
13770 relies. As part of this determination the evaluator repeats a sample of the tests performed by the
13771 developer and performs any additional tests required to fully demonstrate the expected behaviour
13772 of the composed TOE and the interfaces of the base component relied upon by the dependent
13773 component.

13774 **16.6.2.2 Input**

13775 The evaluation evidence for this sub-activity is:

13776 a) the composed TOE suitable for testing;

13777 b) the composed TOE testing evidence;

13778 c) the reliance information;

13779 d) the development information.

13780 **16.6.2.3 Action ACO_CTT.2.1E**

13781 ISO/IEC 15408-3 ACO_CTT.2.1C: *The composed TOE and base component interface test*
13782 *documentation shall consist of test plans, expected test results and actual test results.*

13783 **16.6.2.3.1 Work unit ACO_CTT.2-1**

13784 The evaluator *shall examine* the composed TOE test documentation to determine that it consists
13785 of test plans, expected test results and actual test results.

13786 This work unit may be satisfied by provision of the test evidence from the evaluation of the
13787 dependent component if the base component was used to satisfy the requirements for IT in the
13788 operational environment of the dependent component.

13789 All work units necessary for the satisfaction of ATE_FUN.1.1E will be applied to determine:

13790 a) that the test documentation consist of test plans expected test results and actual test
13791 results;

13792 b) that the test documentation contains the information necessary to ensure the tests are
13793 repeatable;

13794 c) the level of developer effort that was applied to testing of the base component.

13795 **16.6.2.3.2 Work unit ACO_CTT.2-2**

13796 The evaluator *shall examine* the base component interface test documentation to determine that it
13797 consists of test plans, expected test results and actual test results.

13798 This work unit may be satisfied by provision of the test evidence from the evaluation of the base
13799 component for those interfaces relied upon in the composed TOE by the dependent component are
13800 TSFIs of the successfully evaluated base component. The determination of whether the interfaces
13801 of the base component relied upon by the dependent component were in fact TSFIs of the
13802 evaluated base component is made during the ACO_COR activity.

13803 All work units necessary for the satisfaction of ATE_FUN.1.1E will be applied to determine:

13804 a) that the test documentation consist of test plans expected test results and actual test
13805 results;

13806 b) that the test documentation contains the information necessary to ensure the tests are
13807 repeatable;

13808 c) the level of developer effort that was applied to testing of the base component.

13809 ISO/IEC 15408-3 ACO_CTT.2.2C: *The test documentation from the developer execution of the*
13810 *composed TOE tests shall demonstrate that the TSF behaves as specified and is complete.*

13811 **16.6.2.3.3 Work unit ACO_CTT.2-3**

13812 The evaluator *shall examine* the test documentation to determine that it provides accurate
13813 correspondence between the tests in the test documentation relating to the testing of the
13814 composed TOE and the composed TOE SFRs in the composed TOE security target.

13815 A simple cross-table may be sufficient to show test correspondence. The identification of
13816 correspondence between the tests and SFRs presented in the test documentation has to be
13817 unambiguous.

13818 **16.6.2.3.4 Work unit ACO_CTT.2-4**

13819 The evaluator *shall examine* the test documentation to determine that the developer execution of
13820 the composed TOE tests shall demonstrate that the TSF behaves as specified.

- 13821 Guidance on this work unit can be found in:
- 13822 a) Clause 14.2.1.
- 13823 b) Clause 14.2.2.
- 13824 The outputs from the successful execution of the tests (as assessed for ATE_FUN.1.3C can be
13825 compared with the mapping to determine that the SFRs of the composed TOE, as tested by the
13826 developer, behave as expected.
- 13827 ISO/IEC 15408-3 ACO_CTT.2.3C: *The test documentation from the developer execution of the base
13828 component interface tests shall demonstrate that the base component interface relied upon by the
13829 dependent component behaves as specified and is complete.*
- 13830 **16.6.2.3.5 Work unit ACO_CTT.2-5**
- 13831 The evaluator *shall examine* the test documentation to determine that it provides accurate
13832 correspondence between the tests in the test documentation relating to the testing of the base
13833 component interfaces relied upon by the dependent component and the interfaces specified in the
13834 reliance information.
- 13835 A simple cross-table may be sufficient to show test correspondence. The identification of
13836 correspondence between the tests and interfaces presented in the test documentation has to be
13837 unambiguous.
- 13838 **16.6.2.3.6 Work unit ACO_CTT.2-6**
- 13839 The evaluator *shall examine* the test documentation to determine that the developer execution of
13840 the base component interface tests shall demonstrate that the base component interfaces relied
13841 upon by the dependent component behave as specified.
- 13842 Guidance on this work unit can be found in:
- 13843 a) Clause 14.2.1.
- 13844 b) Clause 14.2.2.
- 13845 The outputs from the successful execution of the tests (as assessed for ATE_FUN.1.3C can be
13846 compared with the mapping to determine that the interfaces of the base component, as tested by
13847 the developer, behave as expected.
- 13848 ISO/IEC 15408-3 ACO_CTT.2.4C: *The base component shall be suitable for testing.*
- 13849 **16.6.2.3.7 Work unit ACO_CTT.2-7**
- 13850 The evaluator *shall examine* the composed TOE to determine that it has been installed properly
13851 and is in a known state.
- 13852 To determine that the composed TOE has been installed properly and is in a known state the
13853 ATE_IND.2-1 and ATE_IND.2-2 work units will be applied to the TOE provided by the developer for
13854 testing.
- 13855 **16.6.2.3.8 Work unit ACO_CTT.2-8**
- 13856 The evaluator *shall examine* the set of resources provided by the developer to determine that they
13857 are equivalent to the set of resources used by the base component developer to functionally test
13858 the base component.

13859 To determine that the set of resources provided are equivalent to those used to functionally test
13860 the base component as used in the composed TOE, the ATE_IND.2-3 work unit will be applied.

13861 **16.6.2.4 Action ACO_CTT.2.2E**

13862 **16.6.2.4.1 Work unit ACO_CTT.2-9**

13863 The tests are to be selected and executed in accordance with ATE_IND.2.2E, to demonstrate the
13864 correct behaviour of the SFRs specified in the composed TOE security target.

13865 The evaluator will apply all work units necessary for the satisfaction of ATE_IND.2.2E, reporting in
13866 the ETR for the composed TOE all analysis, results and verdicts as dictated by the associated work
13867 units.

13868 **16.6.2.5 Action ACO_CTT.2.3E**

13869 **16.6.2.5.1 Work unit ACO_CTT.2-10**

13870 The evaluator *shall perform* testing in accordance with ATE_IND.2.3E, for a subset of the SFRs
13871 specified in the composed security target, to confirm that the TSF operates as specified.

13872 The evaluator will apply all work units necessary for the satisfaction of ATE_IND.2.3E, reporting in
13873 the ETR for the composed TOE all analysis, results and verdicts as dictated by the work units.

13874 When selecting interfaces of the TSF of the composed TOE to test, the evaluator should take into
13875 account any modifications to the components from the evaluated version or configuration.
13876 Modifications to the component from that evaluated may include patches introduced, a different
13877 configuration as a result of modified guidance documentation, reliance an additional portion of the
13878 component that was not within the TSF of the component. These modifications will have been
13879 identified during the Composition rationale (ACO_COR) activity.

13880 **16.6.2.5.2 Work unit ACO_CTT.2-11**

13881 The evaluator *shall perform* testing, in accordance with Evaluation of sub-activity (ATE_IND.2), for
13882 a subset of the interfaces to the base component to confirm they operate as specified.

13883 The evaluator will apply all work units necessary for the satisfaction of ATE_IND.2.3E, reporting in
13884 the ETR for the composed TOE all analysis, results and verdicts as dictated by the work units.

13885 When selecting interfaces of the base component to test, the evaluator should take into account any
13886 modifications to the base component from the evaluated version or configuration. In particular, the
13887 evaluator should consider the development of tests to demonstrate the correct behaviour of
13888 interfaces of the base component that were not considered during the evaluation of the base
13889 component. These additional interfaces and other modifications to the base component will have
13890 been identified during the Composition rationale (ACO_COR) activity.

13891 **16.7 Composition vulnerability analysis (ACO_VUL)**

13892 **16.7.1 Evaluation of sub-activity (ACO_VUL.1)**

13893 **16.7.1.1 Objectives**

13894 The objective of this sub-activity is to determine whether the composed TOE, in its operational
13895 environment, has easily exploitable vulnerabilities.

13896 The developer provides details of any residual vulnerabilities reported from evaluation of the
13897 components. The evaluator performs an analysis of the disposition the residual vulnerabilities
13898 reported and also performs a search of the public domain, to identify any new potential

13899 vulnerabilities in the components (i.e. those issues that have been reported in the public domain
 13900 since evaluation of the base component). The evaluator then performs penetration testing to
 13901 demonstrate that the potential vulnerabilities cannot be exploited in the TOE, in its operational
 13902 environment, by an attacker with basic attack potential.

13903 **16.7.1.2 Input**

13904 The evaluation evidence for this sub-activity is:

- 13905 a) the composed TOE suitable for testing;
- 13906 b) the composed ST;
- 13907 c) the composition rationale;
- 13908 d) the guidance documentation;
- 13909 e) information publicly available to support the identification of possible security
 13910 vulnerabilities;
- 13911 f) residual vulnerabilities reported during evaluation of each component.

13912 **16.7.1.3 Application notes**

13913 See the application notes for Evaluation of sub-activity (AVA_VAN.1).

13914 **16.7.1.4 Action ACO_VUL.1.1E**

13915 ISO/IEC 15408-3 ACO_VUL.1.1C: *The composed TOE shall be suitable for testing.*

13916 **16.7.1.4.1 Work unit ACO_VUL.1-1**

13917 The evaluator **shall examine** the composed TOE to determine that it has been installed properly
 13918 and is in a known state.

13919 To determine that the composed TOE has been installed properly and is in a known state the
 13920 ATE_IND.2-1 and ATE_IND.2-2 work units will be applied to the composed TOE.

13921 If the assurance package includes a component from the ACO_CTT family, then the evaluator may
 13922 refer to the result of the work unit ACO_CTT*-1 to demonstrate this has been satisfied.

13923 **16.7.1.4.2 Work unit ACO_VUL.1-2**

13924 The evaluator **shall examine** the composed TOE configuration to determine that any assumptions
 13925 and objectives in the STs the components relating to IT entities for are fulfilled by the other
 13926 components.

13927 The STs for the component may include assumptions about other components that may use the
 13928 component to which the ST relates, e.g. the ST for an operating system used as a base component
 13929 may include an assumption that any applications loaded on the operating system do not run in
 13930 privileged mode. These assumptions and objectives are to be fulfilled by other components in the
 13931 composed TOE.

13932 **16.7.1.5 Action ACO_VUL.1.2E**

13933 **16.7.1.5.1 Work unit ACO_VUL.1-3**

13934 The evaluator *shall examine* the residual vulnerabilities from the base component evaluation to
13935 determine that they are not exploitable in the composed TOE in its operational environment.

13936 The list of vulnerabilities identified in the product during the evaluation of the base component,
13937 which were demonstrated to be non-exploitable in the base component, is to be used as an input
13938 into this activity. The evaluator will determine that the premise(s) on which a vulnerability was
13939 deemed to be non-exploitable is upheld in the composed TOE, or whether the combination has re-
13940 introduced the potential vulnerability. For example, if during evaluation of the base component it
13941 was assumed that a particular operating system service was disabled, which is enabled in the
13942 composed TOE evaluation, any potential vulnerabilities relating to that service previously scoped
13943 out should now be considered.

13944 Also, this list of known, non-exploitable vulnerabilities resulting from the evaluation of the base
13945 component should be considered in the light of any known, non-exploitable vulnerabilities for the
13946 other components (e.g. dependent component) within the composed TOE. This is to consider the
13947 case where a potential vulnerability that is non-exploitable in isolation is exploitable when
13948 integrated with an IT entity containing another potential vulnerability.

13949 **16.7.1.5.2 Work unit ACO_VUL.1-4**

13950 The evaluator *shall examine* the residual vulnerabilities from the dependent component
13951 evaluation to determine that they are not exploitable in the composed TOE in its operational
13952 environment.

13953 The list of vulnerabilities identified in the product during the evaluation of the dependent
13954 component, which were demonstrated to be non-exploitable in the dependent component, is to be
13955 used as an input into this activity. The evaluator will determine that the premise(s) on which a
13956 vulnerability was deemed to be non-exploitable is upheld in the composed TOE, or whether the
13957 combination has re-introduced the potential vulnerability. For example, if during evaluation of the
13958 dependent component it was assumed that IT meeting the operational environment requirements
13959 would not return a certain value in response to a service request, which is provided by the base
13960 component in the composed TOE evaluation, any potential vulnerabilities relating to that return
13961 value previously scoped out should now be considered.

13962 Also, this list of known, non-exploitable vulnerabilities resulting from the evaluation of the
13963 dependent component should be considered in the light of any known, non-exploitable
13964 vulnerabilities for the other components (e.g. base component) within the composed TOE. This is
13965 to consider the case where a potential vulnerability that is non-exploitable in isolation is
13966 exploitable when integrated with an IT entity containing another potential vulnerability.

13967 **16.7.1.6 Action ACO_VUL.1.3E**

13968 **16.7.1.6.1 Work unit ACO_VUL.1-5**

13969 The evaluator *shall examine* the sources of information publicly available to support the
13970 identification of possible security vulnerabilities in the base component that have become known
13971 since the completion of evaluation of the base component.

13972 The evaluator will use the information in the public domain as described in AVA_VAN.1-2 to search
13973 for vulnerabilities in the base component.

13974 Those potential vulnerabilities that were publicly available prior to the evaluation of the base
13975 component do not have to be further investigated unless it is apparent to the evaluator that the
13976 attack potential required by an attacker to exploit the potential vulnerability has been significantly

13977 reduced. This may be through the introduction of some new technology since the base component
13978 evaluation that means the exploitation of the potential vulnerability has been simplified.

13979 **16.7.1.6.2 Work unit ACO_VUL.1-6**

13980 The evaluator *shall examine* the sources of information publicly available to support the
13981 identification of possible security vulnerabilities in the dependent component that have become
13982 known since the completion of the dependent component evaluation.

13983 The evaluator will use the information in the public domain as described in AVA_VAN.1-2 to search
13984 for vulnerabilities in the dependent component.

13985 Those potential vulnerabilities that were publicly available prior to the evaluation of the
13986 dependent component do not have to be further investigated unless it is apparent to the evaluator
13987 that the attack potential required by an attacker to exploit the potential vulnerability has been
13988 significantly reduced. This may be through the introduction of some new technology since
13989 evaluation of the dependent component that means the exploitation of the potential vulnerability
13990 has been simplified.

13991 **16.7.1.6.3 Work unit ACO_VUL.1-7**

13992 The evaluator *shall record* in the ETR the identified potential security vulnerabilities that are
13993 candidates for testing and applicable to the composed TOE in its operational environment.

13994 The ST, guidance documentation and functional specification are used to determine whether the
13995 vulnerabilities are relevant to the composed TOE in its operational environment.

13996 The evaluator records any reasons for exclusion of vulnerabilities from further consideration if the
13997 evaluator determines that the vulnerability is not applicable in the operational environment.
13998 Otherwise the evaluator records the potential vulnerability for further consideration.

13999 A list of potential vulnerabilities applicable to the composed TOE in its operational environment,
14000 which can be used as an input into penetration testing activities (i.e. ACO_VUL.1.4E), shall be
14001 reported in the ETR by the evaluators.

14002 **16.7.1.7 Action ACO_VUL.1.4E**

14003 **16.7.1.7.1 Work unit ACO_VUL.1-8**

14004 The evaluator *shall conduct* penetration testing as detailed for AVA_VAN.1.3E.

14005 The evaluator will apply all work units necessary for the satisfaction of evaluator action
14006 AVA_VAN.1.3E, reporting in the ETR for the composed TOE all analysis and verdicts as dictated by
14007 the work units.

14008 The evaluator will also apply the work units for the evaluator action AVA_VAN.1.1E to determine
14009 that the composed TOE provided by the developer is suitable for testing.

14010 **16.7.2 Evaluation of sub-activity (ACO_VUL.2)**

14011 **16.7.2.1 Objectives**

14012 The objective of this sub-activity is to determine whether the composed TOE, in its operational
14013 environment, has vulnerabilities exploitable by attackers possessing basic attack potential.

14014 The developer provides an analysis of the disposition of any residual vulnerabilities reported for
14015 the components and of any vulnerabilities introduced through the combination of the base and
14016 dependent components. The evaluator performs a search of the public domain to identify any new

14017 potential vulnerabilities in the components (i.e. those issues that have been reported in the public
14018 domain since the completion of the evaluation of the components). The evaluator will also perform
14019 an independent vulnerability analysis of the composed TOE and penetration testing.

14020 **16.7.2.2 Input**

14021 The evaluation evidence for this sub-activity is:

14022 a) the composed TOE suitable for testing;

14023 b) the composed ST;

14024 c) the composition rationale;

14025 d) the reliance information;

14026 e) the guidance documentation;

14027 f) information publicly available to support the identification of possible security
14028 vulnerabilities.

14029 g) residual vulnerabilities reported during evaluation of each component.

14030 **16.7.2.3 Application notes**

14031 See the application notes for Evaluation of sub-activity (AVA_VAN.2).

14032 **16.7.2.4 Action ACO_VUL.2.1E**

14033 ISO/IEC 15408-3 ACO_VUL.2.1C: *The composed TOE shall be suitable for testing.*

14034 **16.7.2.4.1 Work unit ACO_VUL.2-1**

14035 The evaluator **shall examine** the composed TOE to determine that it has been installed properly
14036 and is in a known state.

14037 To determine that the composed TOE has been installed properly and is in a known state the
14038 ATE_IND.2-1 and ATE_IND.2-2 work units will be applied to the composed TOE.

14039 If the assurance package includes ACO_CTT family, then the evaluator may refer to the result of the
14040 work unit Composed TOE testing (ACO_CTT)*-1 to demonstrate this has been satisfied.

14041 **16.7.2.4.2 Work unit ACO_VUL.2-2**

14042 The evaluator **shall examine** the composed TOE configuration to determine that any assumptions
14043 and objectives in the STs the components relating to IT entities for are fulfilled by the other
14044 components.

14045 The STs for the component may include assumptions about other components that may use the
14046 component to which the ST relates, e.g. the ST for an operating system used as a base component
14047 may include an assumption that any applications loaded on the operating system do not run in
14048 privileged mode. These assumptions and objectives are to be fulfilled by other components in the
14049 composed TOE.

14050 **16.7.2.5 Action ACO_VUL.2.2E**14051 **16.7.2.5.1 Work unit ACO_VUL.2-3**

14052 The evaluator *shall examine* the residual vulnerabilities from the base component evaluation to
14053 determine that they are not exploitable in the composed TOE in its operational environment.

14054 The list of vulnerabilities identified in the product during the evaluation of the base component,
14055 which were demonstrated to be non-exploitable in the base component, is to be used as an input
14056 into this activity. The evaluator will determine that the premise(s) on which a vulnerability was
14057 deemed to be non-exploitable is upheld in the composed TOE, or whether the combination has re-
14058 introduced the potential vulnerability. For example, if during evaluation of the base component it
14059 was assumed that a particular operating system service was disabled, which is enabled in the
14060 composed TOE evaluation, any potential vulnerabilities relating to that service previously scoped
14061 out should now be considered.

14062 Also, this list of known, non-exploitable vulnerabilities resulting from the evaluation of the base
14063 component should be considered in the light of any known, non-exploitable vulnerabilities for the
14064 other components (e.g. dependent component) within the composed TOE. This is to consider the
14065 case where a potential vulnerability that is non-exploitable in isolation is exploitable when
14066 integrated with an IT entity containing another potential vulnerability.

14067 **16.7.2.5.2 Work unit ACO_VUL.2-4**

14068 The evaluator *shall examine* the residual vulnerabilities from the dependent component
14069 evaluation to determine that they are not exploitable in the composed TOE in its operational
14070 environment.

14071 The list of vulnerabilities identified in the product during the evaluation of the dependent
14072 component, which were demonstrated to be non-exploitable in the dependent component, is to be
14073 used as an input into this activity. The evaluator will determine that the premise(s) on which a
14074 vulnerability was deemed to be non-exploitable is upheld in the composed TOE, or whether the
14075 combination has re-introduced the potential vulnerability. For example, if during evaluation of the
14076 dependent component it was assumed that IT meeting the operational environment requirements
14077 would not return a certain value in response to a service request, which is provided by the base
14078 component in the composed TOE evaluation, any potential vulnerabilities relating to that return
14079 value previously scoped out should now be considered.

14080 Also, this list of known, non-exploitable vulnerabilities resulting from the evaluation of the
14081 dependent component should be considered in the light of any known, non-exploitable
14082 vulnerabilities for the other components (e.g. base component) within the composed TOE. This is
14083 to consider the case where a potential vulnerability that is non-exploitable in isolation is
14084 exploitable when integrated with an IT entity containing another potential vulnerability.

14085 **16.7.2.6 Action ACO_VUL.2.3E**14086 **16.7.2.6.1 Work unit ACO_VUL.2-5**

14087 The evaluator *shall examine* the sources of information publicly available to support the
14088 identification of possible security vulnerabilities in the base component that have become known
14089 since the completion of the base component evaluation.

14090 The evaluator will use the information in the public domain as described in AVA_VAN.2-2 to search
14091 for vulnerabilities in the base component.

14092 Those potential vulnerabilities that were publicly available prior to the evaluation of the base
14093 component do not have to be further investigated unless it is apparent to the evaluator that the
14094 attack potential required by an attacker to exploit the potential vulnerability has been significantly

14095 reduced. This may be through the introduction of some new technology since the base component
14096 evaluation that means the exploitation of the potential vulnerability has been simplified.

14097 **16.7.2.6.2 Work unit ACO_VUL.2-6**

14098 The evaluator *shall examine* the sources of information publicly available to support the
14099 identification of possible security vulnerabilities in the dependent component that have become
14100 known since the completion of the dependent component evaluation.

14101 The evaluator will use the information in the public domain as described in AVA_VAN.2-2 to search
14102 for vulnerabilities in the dependent component.

14103 Those potential vulnerabilities that were publicly available prior to the evaluation of the
14104 dependent component do not have to be further investigated unless it is apparent to the evaluator
14105 that the attack potential required by an attacker to exploit the potential vulnerability has been
14106 significantly reduced. This may be through the introduction of some new technology since
14107 evaluation of the dependent component that means the exploitation of the potential vulnerability
14108 has been simplified.

14109 **16.7.2.6.3 Work unit ACO_VUL.2-7**

14110 The evaluator *shall record* in the ETR the identified potential security vulnerabilities that are
14111 candidates for testing and applicable to the composed TOE in its operational environment.

14112 The ST, guidance documentation and functional specification are used to determine whether the
14113 vulnerabilities are relevant to the composed TOE in its operational environment.

14114 The evaluator records any reasons for exclusion of vulnerabilities from further consideration if the
14115 evaluator determines that the vulnerability is not applicable in the operational environment.
14116 Otherwise the evaluator records the potential vulnerability for further consideration.

14117 A list of potential vulnerabilities applicable to the composed TOE in its operational environment,
14118 which can be used as an input into penetration testing activities (ACO_VUL.2.5E), shall be reported
14119 in the ETR by the evaluators.

14120 **16.7.2.7 Action ACO_VUL.2.4E**

14121 **16.7.2.7.1 Work unit ACO_VUL.2-8**

14122 The evaluator *shall conduct* a search of the composed TOE ST, guidance documentation, reliance
14123 information and composition rationale to identify possible security vulnerabilities in the composed
14124 TOE.

14125 The consideration of the components of the composed TOE in the independent evaluator
14126 vulnerability analysis will take a slightly different form to that documented in AVA_VAN.2.3E for a
14127 component evaluation, as it will not necessarily consider all layers of design abstraction relevant to
14128 the assurance package. These will have already been considered during the evaluation of the
14129 components, but the evidence may not be available for the composed TOE evaluation. However, the
14130 general approach described in the work units associated with AVA_VAN.2.3E is applicable and
14131 should form the basis of the evaluator's search for potential vulnerabilities in the composed TOE.

14132 A vulnerability analysis of the individual components used in the composed TOE will have already
14133 been performed during evaluation of the individual components. The focus of the vulnerability
14134 analysis during the composed TOE evaluation is to identify any vulnerabilities introduced as a
14135 result of the integration of the components or due to any changes in the use of the components
14136 between the evaluated component configuration to the composed TOE configuration.

14137 The evaluator will use the understanding of the component's construction as detailed in the
 14138 reliance information for the dependent component, and the development information and
 14139 composition rationale for the base component, together with the dependent component design
 14140 information. This information will allow the evaluator to gain an understanding of how the base
 14141 component and dependent component interact and identify potential vulnerabilities that may be
 14142 introduced as a result of this interaction.

14143 The evaluator will consider any new guidance provided for the installation, start-up and operation
 14144 of the composed TOE to identify any potential vulnerabilities introduced through this revised
 14145 guidance.

14146 If any of the individual components have been through assurance continuity activities since the
 14147 completion of the component evaluation, the evaluator will consider the patch(es) in the
 14148 independent vulnerability analysis. Information related to the change provided in a public report of
 14149 the assurance continuity activities (e.g. Maintenance Report) will be the main source of input
 14150 material of the change. This will be supplemented by any updates to the guidance documentation
 14151 resulting from the change and any information regarding the change available in the public domain,
 14152 e.g. vendor website.

14153 Any risks identified due to the lack of evidence to establish the full impact of any patches or
 14154 deviations in the configuration of a component from the evaluated configuration are to be
 14155 documented in the evaluator's vulnerability analysis.

14156 **16.7.2.8 Action ACO_VUL.2.5E**

14157 **16.7.2.8.1 Work unit ACO_VUL.2-9**

14158 The evaluator *shall conduct* penetration testing as detailed for AVA_VAN.2.4E.

14159 The evaluator will apply all work units necessary for the satisfaction of evaluator action
 14160 AVA_VAN.2.4E, reporting in the ETR for the composed TOE all analysis and verdicts as dictated by
 14161 the work units.

14162 The evaluator will also apply the work units for the evaluator action AVA_VAN.2.1E to determine
 14163 that the composed TOE provided by the developer is suitable for testing.

14164 **16.7.3 Evaluation of sub-activity (ACO_VUL.3)**

14165 **16.7.3.1 Objectives**

14166 The objective of this sub-activity is to determine whether the composed TOE, in its operational
 14167 environment, has vulnerabilities exploitable by attackers possessing Enhanced-Basic attack
 14168 potential.

14169 The developer provides an analysis of the disposition of any residual vulnerabilities reported for
 14170 the components and of any vulnerabilities introduced through the combination of the base and
 14171 dependent components. The evaluator performs a search of the public domain to identify any new
 14172 potential vulnerabilities in the components (i.e. those issues that have been reported in the public
 14173 domain since the completion of the component evaluations). The evaluator will also perform an
 14174 independent vulnerability analysis of the composed TOE and penetration testing.

14175 **16.7.3.2 Input**

14176 The evaluation evidence for this sub-activity is:

14177 a) the composed TOE suitable for testing;

14178 b) the composed ST;

- 14179 c) the composition rationale;
- 14180 d) the reliance information;
- 14181 e) the guidance documentation;
- 14182 f) information publicly available to support the identification of possible security
14183 vulnerabilities.
- 14184 g) residual vulnerabilities reported during evaluation of each component.

14185 **16.7.3.3 Application notes**

14186 See the application notes for Evaluation of sub-activity (AVA_VAN.3).

14187 **16.7.3.4 Action ACO_VUL.3.1E**

14188 ISO/IEC 15408-3 ACO_VUL.3.1C: *The composed TOE shall be suitable for testing.*

14189 **16.7.3.4.1 Work unit ACO_VUL.3-1**

14190 The evaluator *shall examine* the composed TOE to determine that it has been installed properly
14191 and is in a known state.

14192 To determine that the composed TOE has been installed properly and is in a known state the
14193 ATE_IND.2-1 and ATE_IND.2-2 work units will be applied to the composed TOE.

14194 If the assurance package includes ACO_CTT family, then the evaluator may refer to the result of the
14195 work unit Composed TOE testing (ACO_CTT)*-1 to demonstrate this has been satisfied.

14196 **16.7.3.4.2 Work unit ACO_VUL.3-2**

14197 The evaluator *shall examine* the composed TOE configuration to determine that any assumptions
14198 and objectives in the STs the components relating to IT entities for are fulfilled by the other
14199 components.

14200 The STs for the component may include assumptions about other components that may use the
14201 component to which the ST relates, e.g. the ST for an operating system used as a base component
14202 may include an assumption that any applications loaded on the operating system do not run in
14203 privileged mode. These assumptions and objectives are to be fulfilled by other components in the
14204 composed TOE.

14205 **16.7.3.5 Action ACO_VUL.3.2E**

14206 **16.7.3.5.1 Work unit ACO_VUL.3-3**

14207 The evaluator *shall examine* the residual vulnerabilities from the base component evaluation to
14208 determine that they are not exploitable in the composed TOE in its operational environment.

14209 The list of vulnerabilities identified in the product during the evaluation of the base component,
14210 which were demonstrated to be non-exploitable in the base component, is to be used as an input
14211 into this activity. The evaluator will determine that the premise(s) on which a vulnerability was
14212 deemed to be non-exploitable is upheld in the composed TOE, or whether the combination has re-
14213 introduced the potential vulnerability. For example, if during evaluation of the base component it
14214 was assumed that a particular operating system service was disabled, which is enabled in the
14215 composed TOE evaluation, any potential vulnerabilities relating to that service previously scoped
14216 out should now be considered.

14217 Also, this list of known, non-exploitable vulnerabilities resulting from the evaluation of the base
 14218 component should be considered in the light of any known, non-exploitable vulnerabilities for the
 14219 other components (e.g. dependent component) within the composed TOE. This is to consider the
 14220 case where a potential vulnerability that is non-exploitable in isolation is exploitable when
 14221 integrated with an IT entity containing another potential vulnerability.

14222 **16.7.3.5.2 Work unit ACO_VUL.3-4**

14223 The evaluator *shall examine* the residual vulnerabilities from the dependent component
 14224 evaluation to determine that they are not exploitable in the composed TOE in its operational
 14225 environment.

14226 The list of vulnerabilities identified in the product during the evaluation of the dependent
 14227 component, which were demonstrated to be non-exploitable in the dependent component, is to be
 14228 used as an input into this activity. The evaluator will determine that the premise(s) on which a
 14229 vulnerability was deemed to be non-exploitable is upheld in the composed TOE, or whether the
 14230 combination has re-introduced the potential vulnerability. For example, if during evaluation of the
 14231 dependent component it was assumed that IT meeting the operational environment requirements
 14232 would not return a certain value in response to a service request, which is provided by the base
 14233 component in the composed TOE evaluation, any potential vulnerabilities relating to that return
 14234 value previously scoped out should now be considered.

14235 Also, this list of known, non-exploitable vulnerabilities resulting from the evaluation of the
 14236 dependent component should be considered in the light of any known, non-exploitable
 14237 vulnerabilities for the other components (e.g. base component) within the composed TOE. This is
 14238 to consider the case where a potential vulnerability that is non-exploitable in isolation is
 14239 exploitable when integrated with an IT entity containing another potential vulnerability.

14240 **16.7.3.6 Action ACO_VUL.3.3E**

14241 **16.7.3.6.1 Work unit ACO_VUL.3-5**

14242 The evaluator *shall examine* the sources of information publicly available to support the
 14243 identification of possible security vulnerabilities in the base component that have become known
 14244 since the completion of the base component evaluation.

14245 The evaluator will use the information in the public domain as described in AVA_VAN.3-2 to search
 14246 for vulnerabilities in the base component.

14247 Those potential vulnerabilities that were publicly available prior to the evaluation of the base
 14248 component do not have to be further investigated unless it is apparent to the evaluator that the
 14249 attack potential required by an attacker to exploit the potential vulnerability has been significantly
 14250 reduced. This may be through the introduction of some new technology since the base component
 14251 evaluation that means the exploitation of the potential vulnerability has been simplified.

14252 **16.7.3.6.2 Work unit ACO_VUL.3-6**

14253 The evaluator *shall examine* the sources of information publicly available to support the
 14254 identification of possible security vulnerabilities in the dependent component that have become
 14255 known since completion of the dependent component evaluation.

14256 The evaluator will use the information in the public domain as described in AVA_VAN.3-2 to search
 14257 for vulnerabilities in the dependent component.

14258 Those potential vulnerabilities that were publicly available prior to the evaluation of the
 14259 dependent component do not have to be further investigated unless it is apparent to the evaluator
 14260 that the attack potential required by an attacker to exploit the potential vulnerability has been
 14261 significantly reduced. This may be through the introduction of some new technology since

14262 evaluation of the dependent component that means the exploitation of the potential vulnerability
14263 has been simplified.

14264 **16.7.3.6.3 Work unit ACO_VUL.3-7**

14265 The evaluator *shall record* in the ETR the identified potential security vulnerabilities that are
14266 candidates for testing and applicable to the composed TOE in its operational environment.

14267 The ST, guidance documentation and functional specification are used to determine whether the
14268 vulnerabilities are relevant to the composed TOE in its operational environment.

14269 The evaluator records any reasons for exclusion of vulnerabilities from further consideration if the
14270 evaluator determines that the vulnerability is not applicable in the operational environment.
14271 Otherwise the evaluator records the potential vulnerability for further consideration.

14272 A list of potential vulnerabilities applicable to the composed TOE in its operational environment,
14273 which can be used as an input into penetration testing activities (ACO_VUL.3.5E), shall be reported
14274 in the ETR by the evaluators.

14275 **16.7.3.7 Action ACO_VUL.3.4E**

14276 **16.7.3.7.1 Work unit ACO_VUL.3-8**

14277 The evaluator *shall conduct* a search of the composed TOE ST, guidance documentation, reliance
14278 information and composition rationale to identify possible security vulnerabilities in the composed
14279 TOE.

14280 The consideration of the components in the independent evaluator vulnerability analysis will take
14281 a slightly different form to that documented in AVA_VAN.3.3E for a component evaluation, as it will
14282 not necessarily consider all layers of design abstraction relevant to the assurance package. These
14283 will have already been considered during the evaluation of the base component, but the evidence
14284 may not be available for the composed TOE evaluation. However, the general approach described
14285 in the work units associated with AVA_VAN.3.3E is applicable and should form the basis of the
14286 evaluator's search for potential vulnerabilities in the composed TOE.

14287 A vulnerability analysis of the individual components used in the composed TOE will have already
14288 been performed during evaluation of the components. The focus of the vulnerability analysis
14289 during the composed TOE evaluation is to identify any vulnerabilities introduced as a result of the
14290 integration of the components or due to any changes in the use of the components between the
14291 configuration of the component determined during the component evaluation and the composed
14292 TOE configuration.

14293 The evaluator will use the understanding of the component's construction as detailed in the
14294 reliance information for the dependent component, and the composition rationale and
14295 development information for the base component, together with the dependent component design
14296 information. This information will allow the evaluator to gain an understanding of how the base
14297 component and dependent component interact.

14298 The evaluator will consider any new guidance provided for the installation, start-up and operation
14299 of the composed TOE to identify any potential vulnerabilities introduced through this revised
14300 guidance.

14301 If any of the individual components have been through assurance continuity activities since the
14302 completion of the component evaluation, the evaluator will consider the patch in the independent
14303 vulnerability analysis. Information related to the change provided in a public report of the
14304 assurance continuity activities (e.g. Maintenance Report). This will be supplemented by any
14305 updates to the guidance documentation resulting from the change and any information regarding
14306 the change available in the public domain, e.g. vendor website.

- 14307 Any risks identified due to the lack of evidence to establish the full impact of any patches or
14308 deviations in the configuration of a component from the evaluated configuration are to be
14309 documented in the evaluator's vulnerability analysis.
- 14310 **16.7.3.8 Action ACO_VUL.3.5E**
- 14311 **16.7.3.8.1 Work unit ACO_VUL.3-9**
- 14312 The evaluator *shall conduct* penetration testing as detailed for AVA_VAN.3.4E.
- 14313 The evaluator will apply all work units necessary for the satisfaction of evaluator action
14314 AVA_VAN.3.4E, reporting in the ETR for the composed TOE all analysis and verdicts as dictated by
14315 the work units.
- 14316 The evaluator will also apply the work units for the evaluator action AVA_VAN.3.1E to determine
14317 that the composed TOE provided by the developer is suitable for testing.

14318 **Annex A**
14319 (informative)

14320 **General evaluation guidance**
14321

14322 **A.1 Objectives**

14323 The objective of this clause is to cover general guidance used to provide technical evidence of
14324 evaluation results. The use of such general guidance helps in achieving objectivity, repeatability
14325 and reproducibility of the work performed by the evaluator.

14326 **A.2 Sampling**

14327 This Subclause provides general guidance on sampling. Specific and detailed information is given in
14328 those work units under the specific evaluator action elements where sampling has to be performed.

14329 Sampling is a defined procedure of an evaluator whereby some subset of a required set of
14330 evaluation evidence is examined and assumed to be representative for the entire set. It allows the
14331 evaluator to gain enough confidence in the correctness of particular evaluation evidence without
14332 analysing the whole evidence. The reason for sampling is to conserve resources while maintaining
14333 an adequate level of assurance. Sampling of the evidence can provide two possible outcomes:

14334 a) The subset reveals no errors, allowing the evaluator to have some confidence that the
14335 entire set is correct.

14336 b) The subset reveals errors and therefore the validity of the entire set is called into
14337 question. Even the resolution of all errors that were found may be insufficient to provide
14338 the evaluator the necessary confidence and as a result the evaluator may have to increase
14339 the size of the subset, or stop using sampling for this particular evidence.

14340 Sampling is a technique which can be used to reach a reliable conclusion if a set of evidence is
14341 relatively homogeneous in nature, e.g. if the evidence has been produced during a well defined
14342 process.

14343 Sampling in the cases identified in ISO/IEC 15408, and in cases specifically covered in evaluation
14344 methodology work items, is recognised as a cost-effective approach to performing evaluator
14345 actions. Sampling in other areas is permitted only in exceptional cases, where performance of a
14346 particular activity in its entirety would require effort disproportionate to the other evaluation
14347 activities, and where this would not add correspondingly to assurance. In such cases a rationale for
14348 the use of sampling in that area will need to be made. Neither the fact that the TOE is large and
14349 complex, nor that it has many security functional requirements, is sufficient justification, since
14350 evaluations of large, complex TOEs can be expected to require more effort. Rather it is intended
14351 that this exception be limited to cases such as that where the TOE development approach yields
14352 large quantities of material for a particular ISO/IEC 15408 requirement that would normally all
14353 need to be checked or examined, and where such an action would not be expected to raise
14354 assurance correspondingly.

14355 Sampling needs to be justified taking into account the possible impact on the security objectives
14356 and threats of the TOE. The impact depends on what might be missed as a result of sampling.
14357 Consideration also needs to be given to the nature of the evidence to be sampled, and the
14358 requirement not to diminish or ignore any security functions.

14359 It should be recognised that sampling of evidence directly related to the implementation of the TOE
14360 (e.g. developer test results) requires a different approach to sampling, then sampling related to the

14361 determination of whether a process is being followed. In many cases the evaluator is required to
 14362 determine that a process is being followed, and a sampling strategy is recommended. The approach
 14363 for sampling a developer's test results will differ. This is because the former case is concerned with
 14364 ensuring that a process is in place, and the latter deals with determining correct implementation of
 14365 the TOE. Typically, larger sample sizes should be analysed in cases related to the correct
 14366 implementation of the TOE than would be necessary to ensure that a process is in place.

14367 In certain cases it may be appropriate for the evaluator to give greater emphasis to the repetition
 14368 of developer testing. For example if the independent tests left for the evaluator to perform would
 14369 be only superficially different from those included in an extensive developer test set (possibly
 14370 because the developer has performed more testing than necessary to satisfy the Coverage
 14371 (ATE_COV) and Depth (ATE_DPT) criteria) then it would be appropriate for the evaluator to give
 14372 greater focus to the repetition of developer tests. Note that this does not necessarily imply a
 14373 requirement for a high percentage sample for repetition of developer tests; indeed, given an
 14374 extensive developer test set, the evaluator may be able to justify a low percentage sample.

14375 Where the developer has used an automated test suite to perform functional testing, it will usually
 14376 be easier for the evaluator to re-run the entire test suite rather than repeat only a sample of
 14377 developer tests. However the evaluator does have an obligation to check that the automatic testing
 14378 does not give misrepresentative results. The implication is thus that this check must be performed
 14379 for a sample of the automatic test suite, with the principles for selecting some tests in preference to
 14380 others and ensuring a sufficient sample size applying equally in this case.

14381 The following principles should be followed whenever sampling is performed:

14382 a) Sampling should not be random, rather it should be chosen such that it is representative
 14383 of all of the evidence. The sample size and composition must always be justified.

14384 b) When sampling relates to the correct implementation of the TOE, the sample should be
 14385 representative of all aspects relevant to the areas that are sampled. In particular, the
 14386 selection should cover a variety of components, interfaces, developer and operational
 14387 sites (if more than one is involved) and hardware platform types (if more than one is
 14388 involved). The sample size should be commensurate with the cost effectiveness of the
 14389 evaluation and will depend on a number of TOE dependent factors (e.g. the size and
 14390 complexity of the TOE, the amount of documentation).

14391 c) Also, when sampling relates to specifically gaining evidence that the developer testing is
 14392 repeatable and reproducible the sample used must be sufficient to represent all distinct
 14393 aspects of developer testing, such as different test regimes. The sample used must be
 14394 sufficient to detect any systematic problem in the developer's functional testing process.
 14395 The evaluator contribution resulting from the combination of repeating developer tests
 14396 and performing independent tests must be sufficient to address the major points of
 14397 concern for the TOE.

14398 d) Where sampling relates to gaining evidence that a process (e.g. visitor control or design
 14399 review) the evaluator should sample sufficient information to gain reasonable confidence
 14400 that the procedure is being followed.

14401 e) The sponsor and developer should not be informed in advance of the exact composition of
 14402 the sample, subject to ensuring timely delivery of the sample and supporting deliverable,
 14403 e.g. test harnesses and equipment to the evaluator in accordance with the evaluation
 14404 schedule.

14405 f) The choice of the sample should be free from bias to the degree possible (one should not
 14406 always choose the first or last item). Ideally the sample selection should be done by
 14407 someone other than the evaluator.

14408 Errors found in the sample can be categorised as being either systematic or sporadic. If the error is
14409 systematic, the problem should be corrected and a complete new sample taken. If properly
14410 explained, sporadic errors might be solved without the need for a new sample, although the
14411 explanation should be confirmed. The evaluator should use judgement in determining whether to
14412 increase the sample size or use a different sample.

14413 **A.3 Dependencies**

14414 In general it is possible to perform the required evaluation activities, sub-activities, and actions in
14415 any order or in parallel. However, there are different kinds of dependencies which have to be
14416 considered by the evaluator. This Subclause provides general guidance on dependencies between
14417 different activities, sub-activities, and actions.

14418 **A.3.1 Dependencies between activities**

14419 For some cases the different assurance classes may recommend or even require a sequence for the
14420 related activities. A specific instance is the ST activity. The ST evaluation activity is started prior to
14421 any TOE evaluation activities since the ST provides the basis and context to perform them.
14422 However, a final verdict on the ST evaluation may not be possible until the TOE evaluation is
14423 complete, since changes to the ST may result from activity findings during the TOE evaluation.

14424 **A.3.2 Dependencies between sub-activities**

14425 Dependencies identified between components in ISO/IEC 15408-3 have to be considered by the
14426 evaluator. Most dependencies are one way, e.g. Evaluation of sub-activity (AVA_VAN.1) claims a
14427 dependency on Evaluation of sub-activity (ADV_FSP.1) and Evaluation of sub-activity (AGD_OPE.1).
14428 There are also instances of mutual dependencies, where both components depend on each other.
14429 An example of this is Evaluation of sub-activity (ATE_FUN.1) and Evaluation of sub-activity
14430 (ATE_COV.1).

14431 A sub-activity can be assigned a pass verdict normally only if all those sub-activities are
14432 successfully completed on which it has a one-way dependency. For example, a pass verdict on
14433 Evaluation of sub-activity (AVA_VAN.1) can normally only be assigned if the sub-activities related
14434 to Evaluation of sub-activity (ADV_FSP.1) and Evaluation of sub-activity (AGD_OPE.1) are assigned
14435 a pass verdict too. In the case of mutual dependency the ordering of these components is down to
14436 the evaluator deciding which sub-activity to perform first. Note this indicates that pass verdicts can
14437 normally only be assigned once both sub-activities have been successful.

14438 So when determining whether a sub-activity will impact another sub-activity, the evaluator should
14439 consider whether this activity depends on potential evaluation results from any dependent sub-
14440 activities. Indeed, it may be the case that a dependent sub-activity will impact this sub-activity,
14441 requiring previously completed evaluator actions to be performed again.

14442 A significant dependency effect occurs in the case of evaluator-detected flaws. If a flaw is identified
14443 as a result of conducting one sub-activity, the assignment of a pass verdict to a dependent sub-
14444 activity may not be possible until all flaws related to the sub-activity upon which it depends are
14445 resolved.

14446 **A.3.3 Dependencies between actions**

14447 It may be the case, that results which are generated by the evaluator during one action are used for
14448 performing another action. For example, actions for completeness and consistency cannot be
14449 completed until the checks for content and presentation have been completed. This means for
14450 example that the evaluator is recommended to evaluate the PP/ST rationale after evaluating the
14451 constituent parts of the PP/ST.

14452 **A.4 Site Visits**

14453 **A.4.1 Introduction**

14454 The assurance class ALC includes requirements for

- 14455 a) the application of configuration management, ensuring that the integrity of the TOE is
14456 preserved;
- 14457 b) measures, procedures, and standards concerned with secure delivery of the TOE,
14458 ensuring that the security protection offered by the TOE is not compromised during the
14459 transfer to the user,
- 14460 c) security measures, used to protect the development environment.

14461 A development site visit is a useful means whereby the evaluator determines whether procedures
14462 are being followed in a manner consistent with that described in the documentation.

14463 Reasons for visiting sites include:

- 14464 a) to observe the use of the CM system as described in the CM plan;
- 14465 b) to observe the practical application of delivery procedures as described in the delivery
14466 documentation;
- 14467 c) to observe the application of security measures during development and maintenance of
14468 the TOE as described in the development security documentation.

14469 Specific and detailed information is given in work units for those activities where site visits are
14470 performed:

- 14471 a) CM capabilities (ALC_CMC).n with $n \geq 3$ (especially work unit ALC_CMC.3-10 =
14472 ALC_CMC.4-13 = ALC_CMC.5-19);
- 14473 b) Delivery (ALC_DEL) (especially work unit ALC_DEL.1-2);
- 14474 c) Development security (ALC_DVS) (especially work unit ALC_DVS.1-3 = ALC_DVS.2-4).

14475 **A.4.2 General Approach**

14476 During an evaluation, it is often necessary that the evaluator will meet the developer more than
14477 once and it is a question of good planning to combine the site visit with another meeting to reduce
14478 costs. For example, one might combine the site visits for configuration management, for the
14479 developer's security and for delivery. It may also be necessary to perform more than one site visit
14480 to the same site to allow the checking of all development phases. It should be considered that
14481 development could occur at multiple facilities within a single building, multiple buildings at the
14482 same site, or at multiple sites.

14483 The first site visit should be scheduled early during the evaluation. In the case of an evaluation
14484 which starts during the development phase of the TOE, this will allow corrective actions to be
14485 taken, if necessary. In the case of an evaluation which starts after the development of the TOE, an
14486 early site visit could allow corrective measures to be put in place if serious deficiencies in the
14487 applied procedures emerge. This avoids unnecessary evaluation effort.

14488 Interviews are also a useful means of determining whether the written procedures reflect what is
14489 done. In conducting such interviews, the evaluator aims to gain a deeper understanding of the
14490 analysed procedures at the development site, how they are used in practise and whether they are

14491 being applied as described in the provided evaluation evidence. Such interviews complement but
14492 do not replace the examination of evaluation evidence.

14493 As a first step preparing the site visits the evaluators should perform the evaluator work units
14494 concerning the assurance class ALC excluding the aspects describing the results of the site visit.
14495 Based on the information provided by the relevant developer documentation and the remaining
14496 open questions which were not answered by the documentation the evaluators compile a check list
14497 of the questions which are to be resolved by the site visits.

14498 The first version of the evaluation report concerning the ALC class and the check list serves as
14499 input for the consultation with the evaluation authority concerning the site visits.

14500 The check list serves as a guide line for the site visits, which questions are to be answered by
14501 inspection of the relevant measures, their application and results, and by interviews. Where
14502 appropriate, sampling is used for gaining the required level of confidence (see Subclause A.2).

14503 The results of the site visits are recorded and serve as input for the final version of the evaluation
14504 report concerning the assurance class ALC.

14505 Other approaches to gain confidence should be considered that provide an equivalent level of
14506 assurance (e.g. to analyse evaluation evidence). Any decision not to make a visit should be
14507 determined in consultation with the evaluation authority. Appropriate security criteria and a
14508 methodology should be based on other standards of the Information Security Management Systems
14509 area.

14510 **A.4.3 Orientation Guide for the Preparation of the Check List**

14511 In the following some keywords are provided, which topics should be checked during an audit.

14512 **A.4.3.1 Aspects of configuration management**

14513 Basic

14514 — Items of the configuration list, including TOE, source code, run time libraries, design
14515 documentation, development tools (ALC_CMC.3-8).

14516 — Tracking of design documentation, source code, user guidance to different versions of the TOE.

14517 — Integration of the configuration system in the design and development process, test planning,
14518 test analysis and quality management procedures.

14519 Test analysis

14520 — Tracking of test plans and results to specific configurations and versions of the TOE.

14521 Access control to development systems

14522 — Policies for access control and logging.

14523 — Policies for project specific assignment and changing of access rights.

14524 Clearance

14525 — Policies for clearance of the TOE and user guidance to the customer.

14526 — Policies for testing and approving of components and the TOE before deployment.

- 14527 **A.4.3.2 Aspects of development security**
- 14528 Infrastructure
- 14529 — Security measures for physical access control to the development site and rationale for the
14530 effectiveness of these measures.
- 14531 Organisational measures
- 14532 — Organisational structure of the company in respect of the security of the development
14533 environment.
- 14534 — Organisational separation between development, production, testing and quality assurance.
- 14535 Personal measures
- 14536 — Measures for education of the personnel in respect of development security.
- 14537 — Measures and legal agreements of non-disclosure of internal information.
- 14538 Access control
- 14539 — Assignment of secured objects (for instance TOE, source code, run time libraries, design
14540 documentation, development tools, user guidance) and security policies.
- 14541 — Policies and responsibilities concerning the access control and the handling of authentication
14542 information.
- 14543 — Policies for logging of any kind access to the development site and protection of the logging
14544 data.
- 14545 Input, processing and output of data
- 14546 — Security measures for protection of output and output devices (printer, plotter and displays).
- 14547 — Securing of local networks and communication connections.
- 14548 Storage, transfer and destruction of documents and data media.
- 14549 — Policies for handling of documents and data media.
- 14550 — Policies and responsibilities for destruction of sorted out documents and logging of these
14551 events.
- 14552 Data protection
- 14553 — Policies and responsibilities for data and information protection (e.g. for performing backups).
- 14554 Contingency plan
- 14555 — Practises in case of emergency and responsibilities.
- 14556 — Documentation of the contingency measures concerning access control.
- 14557 — Information of the personnel about applicable practises in extreme cases. protection (e.g. for
14558 performing backups).

14559 **A.4.4 Example of a checklist**

14560 The examples of checklists for site visits consist in tables for the preparation of an audit and for the
14561 presentation of the results of an audit.

14562 The checklist structure given in the following is preliminary. Dependent on the concrete contents
14563 of the new guideline, changes might become necessary.

14564 The checklist is divided into three subclauses according to the subjects indicated in the
14565 introduction (Subclause A.4.1).

14566 a) Configuration management system.

14567 b) Delivery procedures.

14568 c) Security measures during development.

14569 These subclauses correspond to the actual ISO/IEC 15408 class ALC, especially the families CM
14570 capabilities (ALC_CMC).n with n>=3, Delivery (ALC_DEL) and Development security (ALC_DVS).

14571 The subclauses are subdivided further into rows corresponding to the relevant work units of this
14572 International Standard.

14573 The columns of the checklist contain in turn

14574 — a consecutive number,

14575 — the referenced work unit,

14576 — the references to the corresponding developer documentation,

14577 — the explicit reproduction of the developer measures,

14578 — special remarks and questions to be clarified on the visit (beyond the standard evaluator task
14579 to verify the application of the indicated measures),

14580 — the result of the examinations during the visit.

14581 If it is decided to have separate checklists for preparation and reporting of the audit, the result
14582 column is omitted in the preparation list and the remarks and questions column is omitted in the
14583 reporting list. The remaining columns should be identical in both lists.

14584 **Table A.1 Example of a checklist at EAL 4 (extract)**

A. Examination of the CM system (ALC_CMC.4 and ALC_CMS.4)					
No.	Work Unit	Developer Documentation	Measures	Questions and Remarks	Result
A.1	ALC_CMC.4-11, ALC_CMC.4-12	“Configuration Management System”, ch. ...	The system automatically managing the source code files is capable of administering user profiles and graded access rights, and of	Does reading or updating of a source code file require a user authentication?	If a user has not the right to access a confidential document, it is not even displayed to him in the file list.

A. Examination of the CM system (ALC_CMC.4 and ALC_CMS.4)					
No.	Work Unit	Developer Documentation	Measures	Questions and Remarks	Result
			checking identification and authentication of users.		
...

B. Examination of the Delivery Procedures (ALC_DEL.1)					
No.	Work Unit	Developer Documentation	Measures	Questions and Remarks	Result
B.1	ALC_DEL.1-1, ALC_DEL.1-2	"Delivery of the TOE", ch. ...	The software is transmitted PGP-signed and encrypted to the customer.	---	The evaluators have checked the process and found it as described, additionally a checksum is transmitted.
...

C. Examination of the organisational and infrastructural developer security (ALC_DVS.1, ALC_LCD.1, ALC_TAT.1)					
No.	Work Unit	Developer Documentation	Measures	Questions and Remarks	Result
C.1	ALC_DVS.1-1, ALC_DVS.1-2	"Security of the development environment", ch. ... (Premises)	The premises are protected by security fencing.	Is the fencing sufficiently strong and high to prevent an easy intrusion into the premises?	The evaluators considered the fencing to be sufficiently strong and high.
C.2	ALC_DVS.1-1, ALC_DVS.1-2	"Security of the development environment", ch. ... (Building)	The building has the following access possibilities: The main entrance which is surveyed by the reception and is closed if the reception is not manned. And an access in the	Is the listing of the access possibilities complete?	Beyond the indicated access possibilities, there is an emergency exit that cannot be opened from the outside. The roller shutters mentioned before can be operated only from

C. Examination of the organisational and infrastructural developer security (ALC_DVS.1, ALC_LCD.1, ALC_TAT.1)					
No.	Work Unit	Developer Documentation	Measures	Questions and Remarks	Result
			goods reception which is secured by two roller shutters.		inside.
...

14587 **A.5 Scheme Responsibilities**

14588 This International Standard describes the minimum technical work that evaluations conducted
 14589 under oversight (scheme) bodies must perform. However, it also recognises (both explicitly and
 14590 implicitly) that there are activities or methods upon which mutual recognition of evaluation results
 14591 do not rely. For the purposes of thoroughness and clarity, and to better delineate where this
 14592 International Standard ends and an individual scheme's methodology begins, the following matters
 14593 are left up to the discretion of the schemes. Schemes may choose to provide the following, although
 14594 they may choose to leave some unspecified. (Every effort has been made to ensure this list is
 14595 complete; evaluators encountering a subject neither listed here nor addressed in this International
 14596 Standard should consult with their evaluation schemes to determine under whose auspices the
 14597 subject falls.)

14598 The matters that schemes may choose to specify include:

- 14599 a) what is required in ensuring that an evaluation was done sufficiently - every scheme has a
 14600 means of verifying the technical competence, understanding of work and the work of its
 14601 evaluators, whether by requiring the evaluators to present their findings to the oversight
 14602 body, by requiring the oversight body to redo the evaluator's work, or by some other
 14603 means that assures the scheme that all evaluation bodies are adequate and comparable;
- 14604 b) process for disposing of evaluation evidence upon completion of an evaluation;
- 14605 c) any requirements for confidentiality (on the part of the evaluator and the non-disclosure
 14606 of information obtained during evaluation);
- 14607 d) the course of action to be taken if a problem is encountered during the evaluation
 14608 (whether the evaluation continues once the problem is remedied, or the evaluation ends
 14609 immediately and the remedied product must be re-submitted for evaluation);
- 14610 e) any specific (natural) language in which documentation must be provided;
- 14611 f) any recorded evidence that must be submitted in the ETR - this International Standard
 14612 specifies the minimum to be reported in an ETR; however, individual schemes may
 14613 require additional information to be included;
- 14614 g) any additional reports (other than the ETR) required from the evaluators -for example,
 14615 testing reports;
- 14616 h) any specific ORs that may be required by the scheme, including the structure, recipients,
 14617 etc. of any such ORs;

- 14618 i) any specific content structure of any written report as a result from an ST evaluation - a
14619 scheme may have a specific format for all of its reports detailing results of an evaluation,
14620 be it the evaluation of a TOE or of an ST;
- 14621 j) any additional PP/ST identification information required;
- 14622 k) any activities to determine the suitability of explicitly-stated requirements in an ST;
- 14623 l) any requirements for provision of evaluator evidence to support re-evaluation and re-use
14624 of evidence;
- 14625 m) any specific handling of scheme identifiers, logos, trademarks, etc.;
- 14626 n) any specific guidance in dealing with cryptography;
- 14627 o) handling and application of scheme, national and international interpretations;
- 14628 p) a list or characterisations of suitable alternative approaches to testing where testing is
14629 infeasible;
- 14630 q) the mechanism by which an evaluation authority can determine what steps an evaluator
14631 took while testing;
- 14632 r) preferred test approach (if any): at internal interface or at external interface;
- 14633 s) a list or characterisation of acceptable means of conducting the evaluator's vulnerability
14634 analysis (e.g. flaw hypothesis methodology);
- 14635 t) information regarding any vulnerabilities and weaknesses to be considered.

14636 **Annex B**
14637 (informative)

14638 **Vulnerability Assessment (AVA)**
14639

14640 This annex provides an explanation of the AVA_VAN criteria and examples of their application. This
14641 annex does not define the AVA criteria; this definition can be found in ISO/IEC 15408-3 Subclause
14642 Class AVA: Vulnerability assessment.

14643 This annex consists of 2 major parts:

14644 a) *Guidance for completing an independent vulnerability analysis.* This is summarised in
14645 subclause B.1, and described in more detail in subclause B.2 . These subclauses describe
14646 how an evaluator should approach the construction of an independent Vulnerability
14647 Analysis.

14648 b) How to characterise and use assumed Attack Potential of an attacker. This is described in
14649 subclauses B.3 to B.5. These subclauses provide an example of how an attack potential
14650 can be characterised and should be used, and provide examples.

14651 **B.1 What is Vulnerability Analysis**

14652 The purpose of the vulnerability assessment activity is to determine the existence and
14653 exploitability of flaws or weaknesses in the TOE in the operational environment. This
14654 determination is based upon analysis performed by the evaluator, and is supported by evaluator
14655 testing.

14656 At the lowest levels of Vulnerability analysis (AVA_VAN) the evaluator simply performs a search of
14657 publicly available information to identify any known weaknesses in the TOE, while at the higher
14658 levels the evaluator performs a structured analysis of the TOE evaluation evidence.

14659 There are three main factors in performing a vulnerability analysis, namely:

14660 a) the identification of potential vulnerabilities;

14661 b) assessment to determine whether the identified potential vulnerabilities could allow an
14662 attacker with the relevant attack potential to violate the SFRs.

14663 c) penetration testing to determine whether the identified potential vulnerabilities are
14664 exploitable in the operational environment of the TOE.

14665 The identification of vulnerabilities can be further decomposed into the evidence to be searched
14666 and how hard to search that evidence to identify potential vulnerabilities. In a similar manner, the
14667 penetration testing can be further decomposed into analysis of the potential vulnerability to
14668 identify attack methods and the demonstration of the attack methods.

14669 These main factors are iterative in nature, i.e. penetration testing of potential vulnerabilities may
14670 lead to the identification of further potential vulnerabilities. Hence, these are performed as a single
14671 vulnerability analysis activity.

14672 **B.2 Evaluator construction of a Vulnerability Analysis**

14673 The evaluator vulnerability analysis is to determine that the TOE is resistant to penetration attacks
14674 performed by an attacker possessing a Basic (for AVA_VAN.1 and AVA_VAN.2), Enhanced-Basic (for

14675 AVA_VAN.3), Moderate (for AVA_VAN.4) or High (for AVA_VAN.5) attack potential. The evaluator
 14676 first assesses the exploitability of all identified potential vulnerabilities. This is accomplished by
 14677 conducting penetration testing. The evaluator should assume the role of an attacker with a Basic
 14678 (for AVA_VAN.1 and AVA_VAN.2), Enhanced-Basic (for AVA_VAN.3), Moderate (for AVA_VAN.4) or
 14679 High (for AVA_VAN.5) attack potential when attempting to penetrate the TOE.

14680 The evaluator considers potential vulnerabilities encountered by the evaluator during the conduct
 14681 of other evaluation activities. The evaluator penetration testing determining TOE resistance to
 14682 these potential vulnerabilities should be performed assuming the role of an attacker with a Basic
 14683 (for AVA_VAN.1 and AVA_VAN.2), Enhanced-Basic (for AVA_VAN.3), Moderate (for AVA_VAN.4) or
 14684 High (for AVA_VAN.5) attack potential.

14685 However, vulnerability analysis should not be performed as an isolated activity. It is closely linked
 14686 with ADV and AGD. The evaluator performs these other evaluation activities with a focus on
 14687 identifying potential vulnerabilities or “areas of concern”. Therefore, evaluator familiarity with the
 14688 generic vulnerability guidance (provided in Subclause B.2.1) is required.

14689 **B.2.1 Generic vulnerability guidance**

14690 The following five categories provide discussion of generic vulnerabilities.

14691 **B.2.1.1 Bypassing**

14692 Bypassing includes any means by which an attacker could avoid security enforcement, by:

- 14693 a) exploiting the capabilities of interfaces to the TOE, or of utilities which can interact with
 14694 the TOE;
- 14695 b) inheriting privileges or other capabilities that should otherwise be denied;
- 14696 c) (where confidentiality is a concern) reading sensitive data stored or copied to
 14697 inadequately protected areas.

14698 Each of the following should be considered (where relevant) in the evaluator’s independent
 14699 vulnerability analysis.

- 14700 a) Attacks based on exploiting the capabilities of interfaces or utilities generally take
 14701 advantage of the absence of the required security enforcement on those interfaces. For
 14702 example, gaining access to functionality that is implemented at a lower level than that at
 14703 which access control is enforced. Relevant items include:

- 14704 1) changing the predefined sequence of invocation of TSFI;
- 14705 2) invoking an additional TSFI;
- 14706 3) using a component in an unexpected context or for an unexpected purpose;
- 14707 4) using implementation detail introduced in less abstract representations;
- 14708 5) using the delay between time of access check and time of use.

- 14709 b) Changing the predefined sequence of invocation of components should be considered
 14710 where there is an expected order in which interfaces to the TOE (e.g. user commands) are
 14711 called to invoke a TSFI (e.g. opening a file for access and then reading data from it). If a
 14712 TSFI is invoked through one of the TOE interfaces (e.g. an access control check), the
 14713 evaluator should consider whether it is possible to bypass the control by performing the
 14714 call at a later point in the sequence or by missing it out altogether.

- 14715 c) Executing an additional component (in the predefined sequence) is a similar form of
14716 attack to the one described above, but involves the calling of some other TOE interface at
14717 some point in the sequence. It can also involve attacks based on interception of sensitive
14718 data passed over a network by use of network traffic analysers (the additional
14719 component here being the network traffic analyser).
- 14720 d) Using a component in an unexpected context or for an unexpected purpose includes using
14721 an unrelated TOE interface to bypass the TSF by using it to achieve a purpose that it was
14722 not designed or intended to achieve. Covert channels are an example of this type of attack
14723 (see B.2.1.4 for further discussion of covert channels). The use of undocumented
14724 interfaces, which may be insecure, also falls into this category. Such interfaces may
14725 include undocumented support and help facilities.
- 14726 e) Using implementation detail introduced in lower representations may allow an attacker
14727 to take advantage of additional functions, resources or attributes that are introduced to
14728 the TOE as a consequence of the refinement process. Additional functionality may include
14729 test harness code contained in software modules and back-doors introduced during the
14730 implementation process.
- 14731 f) Using the delay between time of check and time of use includes scenarios where an access
14732 control check is made and access granted, and an attacker is subsequently able to create
14733 conditions in which, had they applied at the time the access check was made, would have
14734 caused the check to fail. An example would be a user creating a background process to
14735 read and send highly sensitive data to the user's terminal, and then logging out and
14736 logging back in again at a lower sensitivity level. If the background process is not
14737 terminated when the user logs off, the MAC checks would have been effectively bypassed.
- 14738 g) Attacks based on inheriting privileges are generally based on illicitly acquiring the
14739 privileges or capabilities of some privileged component, usually by exiting from it in an
14740 uncontrolled or unexpected manner. Relevant items include:
- 14741 1) executing data not intended to be executable, or making it executable;
- 14742 2) generating unexpected input for a component;
- 14743 3) invalidating assumptions and properties on which lower-level components rely.
- 14744 h) Executing data not intended to be executable, or making it executable includes attacks
14745 involving viruses (e.g. putting executable code or commands in a file which are
14746 automatically executed when the file is edited or accessed, thus inheriting any privileges
14747 the owner of the file has).
- 14748 i) Generating unexpected input for a component can have unexpected effects which an
14749 attacker could take advantage of. For example, if the TSF could be bypassed if a user gains
14750 access to the underlying operating system, it may be possible to gain such access
14751 following the login sequence by exploring the effect of hitting various control or escape
14752 sequences whilst a password is being authenticated.
- 14753 j) Invalidating assumptions and properties on which lower level components rely includes
14754 attacks based on breaking out of the constraints of an application to gain access to an
14755 underlying operating system in order to bypass the TSF of an application. In this case the
14756 assumption being invalidated is that it is not possible for a user of the application to gain
14757 such access. A similar attack can be envisaged against an application on an underlying
14758 database management system: again the TSF could be bypassed if an attacker can break
14759 out of the constraints of the application.

- 14760 k) Attacks based on reading sensitive data stored in inadequately protected areas
 14761 (applicable where confidentiality is a concern) include the following issues which should
 14762 be considered as possible means of gaining access to sensitive data:
- 14763 1) disk scavenging;
 - 14764 2) access to unprotected memory;
 - 14765 3) exploiting access to shared writable files or other shared resources (e.g. swap files);
 - 14766 4) Activating error recovery to determine what access users can obtain. For example, after a
 14767 crash an automatic file recovery system may employ a lost and found directory for
 14768 headerless files, which are on disk without labels. If the TOE implements mandatory
 14769 access controls, it is important to investigate at what security level this directory is kept
 14770 (e.g. at system high), and who has access to this directory.

14771 There are a number of different methods through which an evaluator may identify a back-door,
 14772 including two main techniques. Firstly, by the evaluator inadvertently identifying during testing an
 14773 interface that can be misused. Secondly, through testing each external interface of the TSF in a
 14774 debugging mode to identify any modules that are not called as a part of testing the documented
 14775 interfaces and then inspecting the code that is not called to consider whether it is a back-door.

14776 For a software TOE where Evaluation of sub-activity (ADV_IMP.2) and ALC_TAT.2 or higher
 14777 components are included in the assurance package, the evaluator may consider during their
 14778 analysis of the tools the libraries and packages that are linked by the compiler at compilation stage
 14779 to determine that back-doors are not introduced at this stage.

14780 **B.2.1.2 Tampering**

14781 Tampering includes any attack based on an attacker attempting to influence the behaviour of the
 14782 TSF (i.e. corruption or de-activation), for example by:

- 14783 a) accessing data on whose confidentiality or integrity the TSF relies;
- 14784 b) forcing the TOE to cope with unusual or unexpected circumstances;
- 14785 c) disabling or delaying security enforcement;
- 14786 d) physical modification the TOE.

14787 Each of the following should be considered (where relevant) in the evaluator's independent
 14788 vulnerability analysis.

- 14789 a) Attacks based on accessing data, whose confidentiality or integrity are protected, include:
 - 14790 1) reading, writing or modifying internal data directly or indirectly;
 - 14791 2) using a component in an unexpected context or for an unexpected purpose;
 - 14792 3) using interfaces between components that are not visible at a higher level of abstraction.
- 14793 b) Reading, writing or modifying internal data directly or indirectly includes the following
 14794 types of attack which should be considered:
 - 14795 1) reading "secrets" stored internally, such as user passwords;
 - 14796 2) spoofing internal data that security enforcing mechanisms rely upon;

- 14797 3) modifying environment variables (e.g. logical names), or data in configuration files or
14798 temporary files.
- 14799 c) It may be possible to deceive a trusted process into modifying a protected file that it
14800 wouldn't normally access.
- 14801 d) The evaluator should also consider the following "dangerous features":
- 14802 1) source code resident on the TOE along with a compiler (for instance, it may be possible to
14803 modify the login source code);
- 14804 2) an interactive debugger and patch facility (for instance, it may be possible to modify the
14805 executable image);
- 14806 3) the possibility of making changes at device controller level, where file protection does not
14807 exist;
- 14808 4) diagnostic code which exists in the source code and that may be optionally included;
- 14809 5) developer's tools left in the TOE.
- 14810 e) Using a component in an unexpected context or for an unexpected purpose includes (for
14811 example), where the TOE is an application built upon an operating system, users
14812 exploiting knowledge of a word processor package or other editor to modify their own
14813 command file (e.g. to acquire greater privileges).
- 14814 f) Using interfaces between components which are not visible at a higher level of
14815 abstraction includes attacks exploiting shared access to resources, where modification of
14816 a resource by one component can influence the behaviour of another (trusted)
14817 component, e.g. at source code level, through the use of global data or indirect
14818 mechanisms such as shared memory or semaphores.
- 14819 g) Attacks based on forcing the TOE to cope with unusual or unexpected circumstances
14820 should always be considered. Relevant items include:
- 14821 1) generating unexpected input for a component;
- 14822 2) invalidating assumptions and properties on which lower-level components rely.
- 14823 h) Generating unexpected input for a component includes investigating the behaviour of the
14824 TOE when:
- 14825 1) command input buffers overflow (possibly "crashing the stack" or overwriting other
14826 storage, which an attacker may be able to take advantage of, or forcing a crash dump that
14827 may contain sensitive information such as clear-text passwords);
- 14828 2) invalid commands or parameters are entered (including supplying a read-only parameter
14829 to an interface which expects to return data via that parameter and supplying improperly
14830 formatted input that should fail parsing such as SQL-injection, format strings);
- 14831 3) an end-of-file marker (e.g. CTRL-Z or CTRL-D) or null character is inserted in an audit
14832 trail.
- 14833 i) Invalidating assumptions and properties on which lower-level components rely includes
14834 attacks taking advantage of errors in the source code where the code assumes (explicitly
14835 or implicitly) that security relevant data is in a particular format or has a particular range
14836 of values. In these cases the evaluator should determine whether they can invalidate such

- 14837 assumptions by causing the data to be in a different format or to have different values,
14838 and if so whether this could confer advantage to an attacker.
- 14839 j) The correct behaviour of the TSF may be dependent on assumptions that are invalidated
14840 under extreme circumstances where resource limits are reached or parameters reach
14841 their maximum value. The evaluator should consider (where practical) the behaviour of
14842 the TOE when these limits are reached, for example:
- 14843 1) changing dates (e.g. examining how the TOE behaves when a critical date threshold is
14844 passed);
- 14845 2) filling disks;
- 14846 3) exceeding the maximum number of users;
- 14847 4) filling the audit log;
- 14848 5) saturating security alarm queues at a console;
- 14849 6) overloading various parts of a multi-user TOE which relies heavily upon communications
14850 components;
- 14851 7) swamping a network, or individual hosts, with traffic;
- 14852 8) filling buffers or fields.
- 14853 k) Attacks based on disabling or delaying security enforcement include the following items:
- 14854 1) using interrupts or scheduling functions to disrupt sequencing;
- 14855 2) disrupting concurrence;
- 14856 3) using interfaces between components which are not visible at a higher level of
14857 abstraction.
- 14858 l) Using interrupts or scheduling functions to disrupt sequencing includes investigating the
14859 behaviour of the TOE when:
- 14860 1) a command is interrupted (with CTRL-C, CTRL-Y, etc.);
- 14861 2) a second interrupt is issued before the first is acknowledged.
- 14862 m) The effects of terminating security critical processes (e.g. an audit daemon) should be
14863 explored. Similarly, it may be possible to delay the logging of audit records or the issuing
14864 or receipt of alarms such that it is of no use to an administrator (since the attack may
14865 already have succeeded).
- 14866 n) Disrupting concurrence includes investigating the behaviour of the TOE when two or
14867 more subjects attempt simultaneous access. It may be that the TOE can cope with the
14868 interlocking required when two subjects attempt simultaneous access, but that the
14869 behaviour becomes less well defined in the presence of further subjects. For example, a
14870 critical security process could be put into a resource-wait state if two other processes are
14871 accessing a resource which it requires.
- 14872 o) Using interfaces between components which are not visible at a higher level of
14873 abstraction may provide a means of delaying a time-critical trusted process.

- 14874 p) Physical attacks can be categorised into physical probing, physical manipulation, physical
14875 modification, and substitution.
- 14876 1) Physical probing by penetrating the TOE targeting internals of the TOE, e.g. reading at
14877 internal communication interfaces, lines or memories.
- 14878 2) Physical manipulation can be with the TOE internals aiming at internal modifications of
14879 the TOE (e.g. by using optical fault induction as an interaction process), at the external
14880 interfaces of the TOE (e.g. by power or clock glitches) and at the TOE environment (e.g. by
14881 modifying temperature).
- 14882 3) Physical modification of TOE internal security enforcing attributes to inherit privileges or
14883 other capabilities that should be denied in regular operation. Such modifications can be
14884 caused, e.g., by optical fault induction. Attacks based on physical modification may also
14885 yield a modification of the TSF itself, e.g. by causing faults at TOE internal program data
14886 transfers before execution. Note, that such kind of bypassing by modifying the TSF itself
14887 can jeopardise every TSF unless there are other measures (possibly environmental
14888 measures) that prevent an attacker from gaining physical access to the TOE.
- 14889 4) Physical substitution to replace the TOE with another IT entity, during delivery or
14890 operation of the TOE. Substitution during delivery of the TOE from the development
14891 environment to the user should be prevented through application of secure delivery
14892 procedures (such as those considered under Development security (ALC_DVS)).
14893 Substitution of the TOE during operation may be considered through a combination of
14894 user guidance and the operational environment, such that the user is able to be confident
14895 that they are interacting with the TOE.

14896 **B.2.1.3 Direct attacks**

14897 Direct attack includes the identification of any penetration tests necessary to test the strength of
14898 permutational or probabilistic mechanism and other mechanisms to ensure they withstand direct
14899 attack.

14900 For example, it may be a flawed assumption that a particular implementation of a pseudo-random
14901 number generator will possess the required entropy necessary to seed the security mechanism.

14902 Where a probabilistic or permutational mechanism relies on selection of security attribute value
14903 (e.g. selection of password length) or entry of data by a human user (e.g. choice of password), the
14904 assumptions made should reflect the worst case.

14905 Probabilistic or permutational mechanisms should be identified during examination of evaluation
14906 evidence required as input to this sub-activity (security target, functional specification, TOE design
14907 and implementation representation subset) and any other TOE (e.g. guidance) documentation may
14908 identify additional probabilistic or permutational mechanisms.

14909 Where the design evidence or guidance includes assertions or assumptions (e.g. about how many
14910 authentication attempts are possible per minute), the evaluator should independently confirm that
14911 these are correct. This may be achieved through testing or through independent analysis.

14912 Direct attacks reliant upon a weakness in a cryptographic algorithm should not be considered
14913 under Vulnerability analysis (AVA_VAN), as this is outside the scope of ISO/IEC 15408. Correctness
14914 of the implementation of the cryptographic algorithm is considered during the ADV and ATE
14915 activities.

14916 **B.2.1.4 Monitoring**

14917 Information is an abstract view on relation between the properties of entities, i.e. a signal contains
 14918 information for a system, if the TOE is able to react to this signal. The TOE resources processes and
 14919 stores information represented by user data. Therefore:

14920 a) information may flow with the user data between subjects by internal TOE transfer or
 14921 export from the TOE;

14922 b) information may be generated and passed to other user data;

14923 c) information may be gained through monitoring the operations on data representing the
 14924 information.

14925 The information represented by user data may be characterised by security attributes like
 14926 "classification level" having values, for example unclassified, confidential, secret, top secret, to
 14927 control operations to the data. This information and therefore the security attributes may be
 14928 changed by operations e.g. FDP_ACC.2 may describe decrease of the level by "sanitisation" or
 14929 increase of level by combination of data. This is one aspects of an information flow analysis focused
 14930 on controlled operations of controlled subjects on controlled objects.

14931 The other aspect is the analysis of *illicit information flow*. This aspect is more general than the
 14932 direct access to objects containing user data addressed by the FDP_ACC family. An *unenforced*
 14933 signalling channel carrying information under control of the information flow control policy can
 14934 also be caused by monitoring of the processing of any object containing or related to this
 14935 information (e.g. side channels). An *enforced* signalling channels may be identified in terms of the
 14936 subjects manipulating resources and the subject or user that observe such manipulation.
 14937 Classically, covert channels have been identified as timing or storage channels, according to the
 14938 resource being modified or modulated. As for other monitoring attacks, the use of the TOE is in
 14939 accordance with the SFRs.

14940 Covert channels are normally applicable in the case when the TOE has unobservability AND multi-
 14941 level separation policy requirements. Covert channels may be routinely spotted during
 14942 vulnerability analysis and design activities, and should therefore be tested. However, generally
 14943 such monitoring attacks are only identified through specialised analysis techniques commonly
 14944 referred to as "covert channel analysis". These techniques have been the subject of much research
 14945 and there are many papers published on this subject. Guidance for the conduct of covert channel
 14946 analysis should be sought from the evaluation authority.

14947 *Unenforced* information flow monitoring attacks include passive analysis techniques aiming at
 14948 disclosure of sensitive internal data of the TOE by operating the TOE in the way that corresponds
 14949 to the guidance documents.

14950 Side Channel Analysis includes crypt analytical techniques based on physical leakage of the TOE.
 14951 Physical leakage can occur by timing information, power consumption or power emanation during
 14952 computation of a TSF. Timing information can be collected also by a remote-attacker (having
 14953 network access to the TOE), power based information channels requires that the attacker is in the
 14954 near-by environment of the TOE.

14955 Eavesdropping techniques include interception of all forms of energy, e.g., electromagnetic or
 14956 optical emanation of computer displays, not necessarily in the near-field of the TOE.

14957 Monitoring also includes exploits of protocol flaws, e.g., an attack on SSL implementation.

14958 **B.2.1.5 Misuse**

14959 Misuse may arise from:

14960 a) incomplete guidance documentation;

14961 b) unreasonable guidance;

14962 c) unintended misconfiguration of the TOE;

14963 d) forced exception behaviour of the TOE.

14964 If the guidance documentation is incomplete the user may not know how to operate the TOE in
14965 accordance with the SFRs. The evaluator should apply familiarity with the TOE gained from
14966 performing other evaluation activities to determine that the guidance is complete. In particular, the
14967 evaluator should consider the functional specification. The TSF described in this document should
14968 be described in the guidance as required to permit secure administration and use through the TSFI
14969 available to human users. In addition, the different modes of operation should be considered to
14970 ensure that guidance is provided for all modes of operation.

14971 The evaluator may, as an aid, prepare an informal mapping between the guidance and these
14972 documents. Any omissions in this mapping may indicate incompleteness.

14973 The guidance is considered to be unreasonable if it makes demands on the TOE's usage or
14974 operational environment that are inconsistent with the ST or unduly onerous to maintain security.

14975 A TOE may use a variety of ways to assist the consumer in effectively using that TOE in accordance
14976 with the SFRs and prevent unintentional misconfiguration. A TOE may employ functionality
14977 (features) to alert the consumer when the TOE is in a state that is inconsistent with the SFRs, whilst
14978 other TOEs may be delivered with enhanced guidance containing suggestions, hints, procedures,
14979 etc. on using the existing security features most effectively; for instance, guidance on using the
14980 audit feature as an aid for detecting when the SFRs are being compromised; namely insecure.

14981 The evaluator considers the TOE's functionality, its purpose and security objectives for the
14982 operational environment to arrive at a conclusion of whether or not there is reasonable
14983 expectation that use of the guidance would permit transition into an insecure state to be detected
14984 in a timely manner.

14985 The potential for the TOE to enter into insecure states may be determined using the evaluation
14986 deliverables, such as the ST, the functional specification and any other design representations
14987 provided as evidence for components included in the assurance package for the TOE (e.g. the
14988 TOE/TSF design specification if a component from TOE design (ADV_TDS) is included).

14989 Instances of forced exception behaviour of the TSF could include, but are not limited to, the
14990 following:

14991 a) behaviour of the TOE when start-up, close-down or error recovery is activated;

14992 b) behaviour of the TOE under extreme circumstances (sometimes termed overload or
14993 asymptotic behaviour), particularly where this could lead to the de-activation or
14994 disabling of parts of the TSF;

14995 c) any potential for unintentional misconfiguration or insecure use arising from attacks
14996 noted in the subclause on tampering above.

14997 **B.2.2 Identification of Potential Vulnerabilities**

14998 Potential vulnerabilities may be identified by the evaluator during different activities. They may
14999 become apparent during an evaluation activity or they may be identified as a result of analysis of
15000 evidence to search for vulnerabilities.

15001 **B.2.2.1 Encountered**

15002 The encountered identification of vulnerabilities is where potential vulnerabilities are identified by
 15003 the evaluator during the conduct of evaluation activities, i.e. the evidence are not being analysed
 15004 with the express aim of identifying potential vulnerabilities.

15005 The encountered method of identification is dependent on the evaluator's experience and
 15006 knowledge; which is monitored and controlled by the evaluation authority. It is not reproducible in
 15007 approach, but will be documented to ensure repeatability of the conclusions from the reported
 15008 potential vulnerabilities.

15009 There are no formal analysis criteria required for this method. Potential vulnerabilities are
 15010 identified from the evidence provided as a result of knowledge and experience. However, this
 15011 method of identification is not constrained to any particular subset of evidence.

15012 Evaluator is assumed to have knowledge of the TOE-type technology and known security flaws as
 15013 documented in the public domain. The level of knowledge assumed is that which can be gained
 15014 from a security e-mail list relevant to the TOE type, the regular bulletins (bug, vulnerability and
 15015 security flaw lists) published by those organisations researching security issues in products and
 15016 technologies in widespread use. This knowledge is not expected to extend to specific conference
 15017 proceedings or detailed theses produced by university research for AVA_VAN.1 or AVA_VAN.2.
 15018 However, to ensure the knowledge applied is up to date, the evaluator may need to perform a
 15019 search of public domain material.

15020 For AVA_VAN.3 to AVA_VAN.5 the search of publicly available information is expected to include
 15021 conference proceeding and theses produced during research activities by universities and other
 15022 relevant organisations.

15023 Examples of how these may arise (how the evaluator may encounter potential vulnerabilities):

15024 a) while the evaluator is examining some evidence, it sparks a memory of a potential
 15025 vulnerability identified in a similar product type, that the evaluator believes to also be
 15026 present in the TOE under evaluation;

15027 b) while examining some evidence, the evaluator spots a flaw in the specification of an
 15028 interface, that reflects a potential vulnerability.

15029 This may include becoming aware of a potential vulnerability in a TOE through reading about
 15030 generic vulnerabilities in a particular product type in an IT security publication or on a security e-
 15031 mail list to which the evaluator is subscribed.

15032 Attack methods can be developed directly from these potential vulnerabilities. Therefore, the
 15033 encountered potential vulnerabilities are collated at the time of producing penetration tests based
 15034 on the evaluator's vulnerability analysis. There is no explicit action for the evaluator to encounter
 15035 potential vulnerabilities. Therefore, the evaluator is directed through an implicit action specified in
 15036 AVA_VAN.1.2E and AVA_VAN.*.4E.

15037 Current information regarding public domain vulnerabilities and attacks may be provided to the
 15038 evaluator by, for example, an evaluation authority. This information is to be taken into account by
 15039 the evaluator when collating encountered vulnerabilities and attack methods when developing
 15040 penetration tests.

15041 **B.2.2.2 Analysis**

15042 The following types of analysis are presented in terms of the evaluator actions.

15043 **B.2.2.2.1 Unstructured Analysis**

15044 The unstructured analysis to be performed by the evaluator (for Evaluation of sub-activity
15045 (AVA_VAN.2)) permits the evaluator to consider the generic vulnerabilities (as discussed in B.2.1).
15046 The evaluator will also apply their experience and knowledge of flaws in similar technology types.

15047 **B.2.2.2.2 Focused**

15048 During the conduct of evaluation activities, the evaluator may also identify areas of concern. These
15049 are specific portions of the TOE evidence that the evaluator has some reservation about, although
15050 the evidence meets the requirements for the activity with which the evidence is associated. For
15051 example, a particular interface specification looks particularly complex, and therefore may be
15052 prone to error either in the development of the TOE or in the operation of the TOE. There is no
15053 potential vulnerability apparent at this stage, further investigation is required. This is beyond the
15054 bounds of encountered, as further investigation is required.

15055 Difference between potential vulnerability and area of concern:

15056 a) Potential vulnerability - The evaluator knows a method of attack that can be used to
15057 exploit the weakness or the evaluator knows of vulnerability information that is relevant
15058 to the TOE.

15059 b) Area of concern - The evaluator may be able to discount concern as a potential
15060 vulnerability based on information provided elsewhere. While reading interface
15061 specification, the evaluator identifies that due to the extreme (unnecessary) complexity
15062 of an interface a potential vulnerability may lay within that area, although it is not
15063 apparent through this initial examination.

15064 The focused approach to the identification of vulnerabilities is an analysis of the evidence with the
15065 aim of identifying any potential vulnerabilities evident through the contained information. It is an
15066 unstructured analysis, as the approach is not predetermined. This approach to the identification of
15067 potential vulnerabilities can be used during the independent vulnerability analysis required by
15068 Evaluation of sub-activity (AVA_VAN.3).

15069 This analysis can be achieved through different approaches, that will lead to commensurate levels
15070 of confidence. None of the approaches have a rigid format for the examination of evidence to be
15071 performed.

15072 The approach taken is directed by the results of the evaluator's assessment of the evidence to
15073 determine it meets the requirements of the AVA/AGD sub-activities. Therefore, the investigation of
15074 the evidence for the existence of potential vulnerabilities may be directed by any of the following:

15075 a) areas of concern identified during examination of the evidence during the conduct of
15076 evaluation activities;

15077 b) reliance on particular functionality to provide separation, identified during the analysis of
15078 the architectural design (as in Evaluation of sub-activity (ADV_ARC.1)), requiring further
15079 analysis to determine it cannot be bypassed;

15080 c) representative examination of the evidence to hypothesise potential vulnerabilities in the
15081 TOE.

15082 The evaluator will report what actions were taken to identify potential vulnerabilities in the
15083 evidence. However, the evaluator may not be able to describe the steps in identifying potential
15084 vulnerabilities before the outset of the examination. The approach will evolve as a result of the
15085 outcome of evaluation activities.

- 15086 The areas of concern may arise from examination of any of the evidence provided to satisfy the
15087 SARs specified for the TOE evaluation. The information publicly accessible is also considered.
- 15088 The activities performed by the evaluator can be repeated and the same conclusions, in terms of
15089 the level of assurance in the TOE, can be reached although the steps taken to achieve those
15090 conclusions may vary. As the evaluator is documenting the form the analysis took, the actual steps
15091 taken to achieve those conclusions are also reproducible.
- 15092 **B.2.2.2.3 Methodical**
- 15093 The methodical analysis approach takes the form of a structured examination of the evidence. This
15094 method requires the evaluator to specify the structure and form the analysis will take (i.e. the
15095 manner in which the analysis is performed is predetermined, unlike the focused identification
15096 method). The method is specified in terms of the information that will be considered and how/why
15097 it will be considered. This approach to the identification of potential vulnerabilities can be used
15098 during the independent vulnerability analysis required by Evaluation of sub-activity (AVA_VAN.4)
15099 and Evaluation of sub-activity (AVA_VAN.5).
- 15100 This analysis of the evidence is deliberate and pre-planned in approach, considering all evidence
15101 identified as an input into the analysis.
- 15102 All evidence provided to satisfy the (ADV) assurance requirements specified in the assurance
15103 package are used as input to the potential vulnerability identification activity.
- 15104 The “methodical” descriptor for this analysis has been used in an attempt to capture the
15105 characterisation that this identification of potential vulnerabilities is to take an ordered and
15106 planned approach. A “method” or “system” is to be applied in the examination. The evaluator is to
15107 describe the method to be used in terms of what evidence will be considered, the information
15108 within the evidence that is to be examined, the manner in which this information is to be
15109 considered; and the hypothesis that is to be generated.
- 15110 The following provide some examples that a hypothesis may take:
- 15111 a) consideration of malformed input for interfaces available to an attacker at the external
15112 interfaces;
- 15113 b) examination of a security mechanism, such as domain separation, hypothesising internal
15114 buffer overflows leading to degradation of separation;
- 15115 c) analysis to identify any objects created in the TOE implementation representation that
15116 are then not fully controlled by the TSF, and could be used by an attacker to undermine
15117 the SFRs.
- 15118 For example, the evaluator may identify that interfaces are a potential area of weakness in the TOE
15119 and specify an approach to the analysis that “all interface specifications provided in the functional
15120 specification and TOE design will be analysed to hypothesise potential vulnerabilities” and go on to
15121 explain the methods used in the hypothesis.
- 15122 This identification method will provide a plan of attack of the TOE, that would be performed by an
15123 evaluator completing penetration testing of potential vulnerabilities in the TOE. The rationale for
15124 the method of identification would provide the evidence for the coverage and depth of exploitation
15125 determination that would be performed on the TOE.

15126 **B.3 When attack potential is used**

15127 **B.3.1 Developer**

15128 Attack potential is used by a PP/ST author during the development of the PP/ST, in consideration
 15129 of the threat environment and the selection of assurance components. This may simply be a
 15130 determination that the attack potential possessed by the assumed attackers of the TOE is
 15131 generically characterised as Basic, Enhanced-Basic, Moderate or High. Alternatively, the PP/ST may
 15132 wish to specify particular levels of individual factors assumed to be possessed by attackers. (e.g.
 15133 the attackers are assumed to be experts in the TOE technology type, with access to specialised
 15134 equipment.)

15135 The PP/ST author considers the threat profile developed during a risk assessment (outside the
 15136 scope of ISO/IEC 15408, but used as an input into the development of the PP/ST in terms of the
 15137 Security Problem Definition or in the case of Direct Rationale STs, the requirements statement).
 15138 Consideration of this threat profile in terms of one of the approaches discussed in the following
 15139 subclauses will permit the specification of the attack potential the TOE is to resist.

15140 **B.3.2 Evaluator**

15141 Attack potential is especially considered by the evaluator in two distinct ways during the ST
 15142 evaluation and the vulnerability assessment activities.

15143 Attack potential is used by an evaluator during the conduct of the vulnerability analysis sub-
 15144 activity to determine whether or not the TOE is resistant to attacks assuming a specific attack
 15145 potential of an attacker. If the evaluator determines that a potential vulnerability is exploitable in
 15146 the TOE, they have to confirm that it is exploitable considering all aspects of the intended
 15147 environment, including the attack potential assumed by an attacker.

15148 Therefore, using the information provided in the threat statement of the Security Target, the
 15149 evaluator determines the minimum attack potential required by an attacker to effect an attack, and
 15150 arrives at some conclusion about the TOE's resistance to attacks. Table B.1 demonstrates the
 15151 relationship between this analysis and attack potential.

Vulnerability Component	TOE resistant to attacker with attack potential of:	Residual vulnerabilities only exploitable by attacker with attack potential of:
VAN.5	High	Beyond High
VAN.4	Moderate	High
VAN.3	Enhanced-Basic	Moderate
VAN.2	Basic	Enhanced-Basic
VAN.1	Basic	Enhanced-Basic

15152 **Table B.1 Vulnerability testing and attack potential**

15153 The “beyond high” entry in the residual vulnerabilities column of the above table represents those
 15154 potential vulnerabilities that would require an attacker to have an attack potential greater than
 15155 that of “high” in order to exploit the potential vulnerability. A vulnerability classified as residual in
 15156 this instance reflects the fact that a known weakness exists in the TOE, but in the current
 15157 operational environment, with the assumed attack potential, the weakness cannot be exploited.

15158 At any level of attack potential a potential vulnerability may be deemed “infeasible” due to a
 15159 countermeasure in the operational environment that prevents the vulnerability from being
 15160 exploited.

15161 A vulnerability analysis applies to all TSFI, including ones that access probabilistic or
 15162 permutational mechanisms. No assumptions are made regarding the correctness of the design and
 15163 implementation of the TSFI; nor are constraints placed on the attack method or the attacker's

15164 interaction with the TOE - if an attack is possible, then it is to be considered during the
 15165 vulnerability analysis. As shown in Table B.1, successful evaluation against a vulnerability
 15166 assurance component reflects that the TSF is designed and implemented to protect against the
 15167 required level of threat.

15168 It is not necessary for an evaluator to perform an attack potential calculation for each potential
 15169 vulnerability. In some cases, it is apparent when developing the attack method whether or not the
 15170 attack potential required to develop and run the attack method is commensurate with that
 15171 assumed of the attacker in the operational environment. For any vulnerabilities for which an
 15172 exploitation is determined, the evaluator performs an attack potential calculation to determine that
 15173 the exploitation is appropriate to the level of attack potential assumed for the attacker.

15174 The approach described below is to be applied whenever it is necessary to calculate attack
 15175 potential, unless the evaluation authority provides mandatory guidance that an alternative
 15176 approach is to be applied. The values given in Tables B.2 and B.3 below are not mathematically
 15177 proven. Therefore, the values given in these example tables may need to be adjusted according to
 15178 the technology type and specific environments. The evaluator should seek guidance from the
 15179 evaluation authority.

15180 **B.4 Calculating attack potential**

15181 **B.4.1 Application of attack potential**

15182 Attack potential is a function of expertise, resources and motivation. There are multiple methods of
 15183 representing and quantifying these factors. Also, there may be other factors that are applicable for
 15184 particular TOE types.

15185 **B.4.1.1 Treatment of motivation**

15186 Motivation is an attack potential factor that can be used to describe several aspects related to the
 15187 attacker and the assets the attacker desires. Firstly, motivation can imply the likelihood of an attack
 15188 - one can infer from a threat described as highly motivated that an attack is imminent, or that no
 15189 attack is anticipated from an un-motivated threat. However, except for the two extreme levels of
 15190 motivation, it is difficult to derive a probability of an attack occurring from motivation.

15191 Secondly, motivation can imply the value of the asset, monetarily or otherwise, to either the
 15192 attacker or the asset holder. An asset of very high value is more likely to motivate an attack
 15193 compared to an asset of little value. However, other than in a very general way, it is difficult to
 15194 relate asset value to motivation because the value of an asset is subjective - it depends largely upon
 15195 the value an asset holder places on it.

15196 Thirdly, motivation can imply the expertise and resources with which an attacker is willing to
 15197 effect an attack. One can infer that a highly-motivated attacker is likely to acquire sufficient
 15198 expertise and resources to defeat the measures protecting an asset. Conversely, one can infer that
 15199 an attacker with significant expertise and resources is not willing to effect an attack using them if
 15200 the attacker's motivation is low.

15201 During the course of preparing for and conducting an evaluation, all three aspects of motivation are
 15202 at some point considered. The first aspect, likelihood of attack, is what may inspire a developer to
 15203 pursue an evaluation. If the developer believes that the attackers are sufficiently motivated to
 15204 mount an attack, then an evaluation can provide assurance of the ability of the TOE to thwart the
 15205 attacker's efforts. Where the operational environment is well defined, for example in a system
 15206 evaluation, the level of motivation for an attack may be known, and will influence the selection of
 15207 countermeasures.

15208 Considering the second aspect, an asset holder may believe that the value of the assets (however
 15209 measured) is sufficient to motivate attack against them. Once an evaluation is deemed necessary,

15210 the attacker's motivation is considered to determine the methods of attack that may be attempted,
15211 as well as the expertise and resources used in those attacks. Once examined, the developer is able
15212 to choose the appropriate assurance level, in particular the AVA requirement components,
15213 commensurate with the attack potential for the threats. During the course of the evaluation, and in
15214 particular as a result of completing the vulnerability assessment activity, the evaluator determines
15215 whether or not the TOE, operating in its operational environment, is sufficient to thwart attackers
15216 with the identified expertise and resources.

15217 It may be possible for a PP author to quantify the motivation of an attacker, as the PP author has
15218 greater knowledge of the operational environment in which the TOE (conforming to the
15219 requirements of the PP) is to be placed. Therefore, the motivation could form an explicit part of the
15220 expression of the attack potential in the PP, along with the necessary methods and measures to
15221 quantify the motivation.

15222 **B.4.2 Characterising attack potential**

15223 This subclause examines the factors that determine attack potential, and provides some guidelines
15224 to help remove some of the subjectivity from this aspect of the evaluation process.

15225 **B.4.2.1 Determining the attack potential**

15226 The determination of the attack potential for an attack corresponds to the identification of the
15227 effort required to create the attack, and to demonstrate that it can be successfully applied to the
15228 TOE (including setting up or building any necessary test equipment), thereby exploiting the
15229 vulnerability in the TOE. The demonstration that the attack can be successfully applied needs to
15230 consider any difficulties in expanding a result shown in the laboratory to create a useful attack. For
15231 example, where an experiment reveals some bits or bytes of a confidential data item (such as a key),
15232 it is necessary to consider how the remainder of the data item would be obtained (in this example
15233 some bits might be measured directly by further experiments, while others might be found by a
15234 different technique such as exhaustive search). It may not be necessary to carry out all of the
15235 experiments to identify the full attack, provided it is clear that the attack actually proves that
15236 access has been gained to a TOE asset, and that the complete attack could realistically be carried
15237 out in exploitation according to the AVA_VAN component targeted. In some cases, the only way to
15238 prove that an attack can realistically be carried out in exploitation according to the AVA_VAN
15239 component targeted is to perform completely the attack and then rate it based upon the resources
15240 actually required. One of the outputs from the identification of a potential vulnerability is assumed
15241 to be a script that gives a step-by-step description of how to carry out the attack that can be used in
15242 the exploitation of the vulnerability on another instance of the TOE.

15243 In many cases, the evaluators will estimate the parameters for exploitation, rather than carry out
15244 the full exploitation. The estimates and their rationale will be documented in the ETR.

15245 **B.4.2.2 Factors to be considered**

15246 The following factors should be considered during analysis of the attack potential required to
15247 exploit a vulnerability:

- 15248 a) Time taken to identify and exploit (*Elapsed Time*);
- 15249 b) Specialist technical expertise required (*Specialist Expertise*);
- 15250 c) Knowledge of the TOE design and operation (*Knowledge of the TOE*);
- 15251 d) Window of opportunity;
- 15252 e) IT hardware/software or other equipment required for exploitation.

15253 In many cases these factors are not independent, but may be substituted for each other in varying
 15254 degrees. For example, expertise or hardware/software may be a substitute for time. A discussion of
 15255 these factors follows. (The levels of each factor are discussed in increasing order of magnitude.)
 15256 When it is the case, the less “expensive” combination is considered in the exploitation phase.

15257 **Elapsed time** is the total amount of time taken by an attacker to identify that a particular potential
 15258 vulnerability may exist in the TOE, to develop an attack method and to sustain effort required to
 15259 mount the attack against the TOE. When considering this factor, the worst-case scenario is used to
 15260 estimate the amount of time required. The identified amount of time is as follows:

- 15261 a) less than one day;
- 15262 b) between one day and one week;
- 15263 c) between one week and two weeks;
- 15264 d) between two weeks and one month;
- 15265 e) each additional month up to 6 months leads to an increased value;
- 15266 f) more than 6 months.

15267 **Specialist expertise** refers to the level of generic knowledge of the underlying principles, product
 15268 type or attack methods (e.g. Internet protocols, Unix operating systems, buffer overflows). The
 15269 identified levels are as follows:

- 15270 a) Laymen are unknowledgeable compared to experts or proficient persons, with no
 15271 particular expertise;
- 15272 b) Proficient persons are knowledgeable in that they are familiar with the security
 15273 behaviour of the product or system type;
- 15274 c) Experts are familiar with the underlying algorithms, protocols, hardware, structures,
 15275 security behaviour, principles and concepts of security employed, techniques and tools
 15276 for the definition of new attacks, cryptography, classical attacks for the product type,
 15277 attack methods, etc. implemented in the product or system type.
- 15278 d) The level “Multiple Expert” is introduced to allow for a situation, where different fields of
 15279 expertise are required at an Expert level for distinct steps of an attack.

15280 It may occur that several types of expertise are required. By default, the higher of the different
 15281 expertises factors is chosen. In very specific cases, the “multiple expert” level could be used but it
 15282 should be noted that the expertise must concern fields that are strictly different like for example
 15283 HW manipulation and cryptography.

15284 **Knowledge of the TOE** refers to specific expertise in relation to the TOE. This is distinct from
 15285 generic expertise, but not unrelated to it. Identified levels are as follows:

- 15286 a) Public information concerning the TOE (e.g. as gained from the Internet);
- 15287 b) Restricted information concerning the TOE (e.g. knowledge that is controlled within the
 15288 developer organisation and shared with other organisations under a non-disclosure
 15289 agreement);
- 15290 c) Sensitive information about the TOE (e.g. knowledge that is shared between discreet
 15291 teams within the developer organisation, access to which is constrained only to members
 15292 of the specified teams);

- 15293 d) Critical information about the TOE (e.g. knowledge that is known by only a few
15294 individuals, access to which is very tightly controlled on a strict need to know basis and
15295 individual undertaking).
- 15296 The knowledge of the TOE may graduate according to design abstraction, although this can only be
15297 done on a TOE by TOE basis. Some TOE designs may be public source (or heavily based on public
15298 source) and therefore even the design representation would be classified as public or at most
15299 restricted, while the implementation representation for other TOEs is very closely controlled as it
15300 would give an attacker information that would aid an attack and is therefore considered to be
15301 sensitive or even critical.
- 15302 It may occur that several types of knowledge are required. In such cases, the higher of the different
15303 knowledge factors is chosen.
- 15304 **Window of opportunity** (Opportunity) is also an important consideration, and has a relationship
15305 to the **Elapsed Time** factor. Identification or exploitation of a vulnerability may require
15306 considerable amounts of access to a TOE that may increase the likelihood of detection. Some attack
15307 methods may require considerable effort off-line, and only brief access to the TOE to exploit. Access
15308 may also need to be continuous, or over a number of sessions.
- 15309 For some TOEs the **Window of opportunity** may equate to the number of samples of the TOE that
15310 the attacker can obtain. This is particularly relevant where attempts to penetrate the TOE and
15311 undermine the SFRs may result in the destruction of the TOE preventing use of that TOE sample for
15312 further testing, e.g. hardware devices. Often in these cases distribution of the TOE is controlled and
15313 so the attacker must apply effort to obtain further samples of the TOE.
- 15314 For the purposes of this discussion:
- 15315 a) unnecessary/unlimited access means that the attack doesn't need any kind of opportunity
15316 to be realised because there is no risk of being detected during access to the TOE and it is
15317 no problem to access the number of TOE samples for the attack;
- 15318 b) easy means that access is required for less than a day and that the number of TOE
15319 samples required to perform the attack is less than ten;
- 15320 c) moderate means that access is required for less than a month and that the number of TOE
15321 samples required to perform the attack is less than one hundred;
- 15322 d) difficult means that access is required for at least a month or that the number of TOE
15323 samples required to perform the attack is at least one hundred;
- 15324 e) none means that the opportunity window is not sufficient to perform the attack (the
15325 length for which the asset to be exploited is available or is sensitive is less than the
15326 opportunity length needed to perform the attack - for example, if the asset key is changed
15327 each week and the attack needs two weeks); another case is, that a sufficient number of
15328 TOE samples needed to perform the attack is not accessible to the attacker - for example
15329 if the TOE is a hardware and the probability to destroy the TOE during the attack instead
15330 of being successful is very high and the attacker has only access to one sample of the TOE.
- 15331 Consideration of this factor may result in determining that it is not possible to complete the exploit,
15332 due to requirements for time availability that are greater than the opportunity time.
- 15333 **IT hardware/software or other equipment** refers to the equipment required to identify or exploit
15334 a vulnerability.
- 15335 a) Standard equipment is readily available to the attacker, either for the identification of a
15336 vulnerability or for an attack. This equipment may be a part of the TOE itself (e.g. a

15337 debugger in an operating system), or can be readily obtained (e.g. Internet downloads,
15338 protocol analyser or simple attack scripts).

15339 b) Specialised equipment is not readily available to the attacker, but could be acquired
15340 without undue effort. This could include purchase of moderate amounts of equipment
15341 (e.g. power analysis tools, use of hundreds of PCs linked across the Internet would fall
15342 into this category), or development of more extensive attack scripts or programs. If
15343 clearly different test benches consisting of specialised equipment are required for
15344 distinct steps of an attack this would be rated as bespoke.

15345 c) Bespoke equipment is not readily available to the public as it may need to be specially
15346 produced (e.g. very sophisticated software), or because the equipment is so specialised
15347 that its distribution is controlled, possibly even restricted. Alternatively, the equipment
15348 may be very expensive.

15349 d) The level “Multiple Bespoke” is introduced to allow for a situation, where different types
15350 of bespoke equipment are required for distinct steps of an attack.

15351 Specialist expertise and **Knowledge of the TOE** are concerned with the information required for
15352 persons to be able to attack a TOE. There is an implicit relationship between an attacker's expertise
15353 (where the attacker may be one or more persons with complementary areas of knowledge) and the
15354 ability to effectively make use of equipment in an attack. The weaker the attacker's expertise, the
15355 lower the potential to use equipment (IT hardware/software or other equipment). Likewise, the
15356 greater the expertise, the greater the potential for equipment to be used in the attack. Although
15357 implicit, this relationship between expertise and the use of equipment does not always apply, for
15358 instance, when environmental measures prevent an expert attacker's use of equipment, or when,
15359 through the efforts of others, attack tools requiring little expertise to be effectively used are
15360 created and freely distributed (e.g. via the Internet).

15361 **B.4.2.3 Calculation of attack potential**

15362 Table B.2 identifies the factors discussed in the previous subclause and associates numeric values
15363 with the total value of each factor.

15364 Where a factor falls close to the boundary of a range the evaluator should consider use of an
15365 intermediate value to those in the table. For example, if twenty samples are required to perform
15366 the attack then a value between one and four may be selected for that factor, or if the design is
15367 based on a publicly available design but the developer has made some alterations then a value
15368 between zero and three should be selected according to the evaluator's view of the impact of those
15369 design changes. The table is intended as a guide.

15370 The “**” specification in the table in considering **Window of Opportunity** is not to be seen as a
15371 natural progression from the timescales specified in the preceding ranges associated with this
15372 factor. This specification identifies that for a particular reason the potential vulnerability cannot be
15373 exploited in the TOE in its intended operational environment. For example, access to the TOE may
15374 be detected after a certain amount of time in a TOE with a known environment (i.e. in the case of a
15375 system) where regular patrols are completed, and the attacker could not gain access to the TOE for
15376 the required two weeks undetected. However, this would not be applicable to a TOE connected to
15377 the network where remote access is possible, or where the physical environment of the TOE is
15378 unknown.

Factor	Value
Elapsed Time	
<= one day	0
<= one week	1
<= two weeks	2

- 15394 d) Divide all attack scenarios {AS1, AS2, ..., ASn} into two groups:
- 15395 1) the attack scenarios having been successful (i.e. those that have been used to successfully
15396 undermine the SFRs), and
- 15397 2) the attack scenarios that have been demonstrated to be unsuccessful.
- 15398 e) For each successful attack scenario, apply Table B.3 and determine, whether there is a
15399 contradiction between the resistance of the TOE and the chosen AVA_VAN assurance
15400 component, see the last column of Table B.3.
- 15401 f) Should one contradiction be found, the vulnerability assessment will fail, e.g. the author of
15402 the ST chose the component AVA_VAN.5 and an attack scenario with an attack potential
15403 of 21 points (high) has broken the security of the TOE. In this case, the TOE is resistant to
15404 attacker with attack potential 'Moderate', this contradicts to AVA_VAN.5, hence, the
15405 vulnerability assessment fails.
- 15406 The "Values" column of Table B.3 indicates the range of attack potential values (calculated using
15407 Table B.2) of an attack scenario that results in the SFRs being undermined.

Values	Attack potential required to exploit scenario:	Meets assurance components:	Failure of components:
0-9	Basic	-	AVA_VAN.1, AVA_VAN.2, AVA_VAN.3, AVA_VAN.4, AVA_VAN.5
10-13	Enhanced-Basic	AVA_VAN.1, AVA_VAN.2	AVA_VAN.3, AVA_VAN.4, AVA_VAN.5
14-19	Moderate	AVA_VAN.1, AVA_VAN.2, AVA_VAN.3	AVA_VAN.4, AVA_VAN.5
20-24	High	AVA_VAN.1, AVA_VAN.2, AVA_VAN.3, AVA_VAN.4	AVA_VAN.5
=>25	Beyond High	AVA_VAN.1, AVA_VAN.2, AVA_VAN.3, AVA_VAN.4, AVA_VAN.5	-

15408

Table B.3 Rating of vulnerabilities and TOE resistance

15409 An approach such as this cannot take account of every circumstance or factor, but should give a
15410 better indication of the level of resistance to attack required to achieve the standard ratings. Other
15411 factors, such as the reliance on unlikely chance occurrences are not included in the basic model, but
15412 can be used by an evaluator as justification for a rating other than those that the basic model might
15413 indicate.

15414 It should be noted that whereas a number of vulnerabilities rated individually may indicate high
15415 resistance to attack, collectively the combination of vulnerabilities may indicate that overall a
15416 lower rating is applicable. The presence of one vulnerability may make another easier to exploit.

15417 If a PP/ST author wants to use the attack potential table for the determination of the level of attack
 15418 the TOE should withstand (selection of Vulnerability analysis (AVA_VAN) component), they should
 15419 proceed as follows: For all different attack scenarios (i.e. for all different types of attacker and/or
 15420 different types of attack the author has in mind) which must not violate the SFRs, several passes
 15421 through Table B.2 should be made to determine the different values of attack potential assumed for
 15422 each such unsuccessful attack scenario. The PP/ST author then chooses the highest value of them
 15423 in order to determine the level of the TOE resistance to be claimed from Table B.3: the TOE
 15424 resistance must be at least equal to this highest value determined. For example, the highest value of
 15425 attack potentials of all attack scenarios, which must not undermine the TOE security policy,
 15426 determined in such a way is Moderate; hence, the TOE resistance shall be at least Moderate (i.e.
 15427 Moderate or High); therefore, the PP/ST author can choose either AVA_VAN.4 (for Moderate) or
 15428 AVA_VAN.5 (for High) as the appropriate assurance component.

15429 **B.5 Example calculation for direct attack**

15430 Mechanisms subject to direct attack are often vital for system security and developers often
 15431 strengthen these mechanisms. As an example, a TOE might use a simple pass number
 15432 authentication mechanism that can be overcome by an attacker who has the opportunity to
 15433 repeatedly guess another user's pass number. The system can strengthen this mechanism by
 15434 restricting pass numbers and their use in various ways. During the course of the evaluation an
 15435 analysis of this direct attack could proceed as follows:

15436 Information gleaned from the ST and design evidence reveals that identification and authentication
 15437 provides the basis upon which to control access to network resources from widely distributed
 15438 terminals. Physical access to the terminals is not controlled by any effective means. The duration of
 15439 access to a terminal is not controlled by any effective means. Authorised users of the system choose
 15440 their own pass numbers when initially authorised to use the system, and thereafter upon user
 15441 request. The system places the following restrictions on the pass numbers selected by the user:

- 15442 a) the pass number must be at least four and no greater than six digits long;
- 15443 b) consecutive numerical sequences are disallowed (such as 7,6,5,4,3);
- 15444 c) repeating digits is disallowed (each digit must be unique).

15445 Guidance provided to the users at the time of pass number selection is that pass numbers should be
 15446 as random as possible and should not be affiliated with the user in some way - a date of birth, for
 15447 instance.

15448 The pass number space is calculated as follows:

- 15449 a) Patterns of human usage are important considerations that can influence the approach to
 15450 searching a password space. Assuming the worst-case scenario and the user chooses a
 15451 number comprising only four digits, the number of pass number permutations assuming
 15452 that each digit must be unique is:

15453
$$7(8)(9)(10) = 5040$$

- 15454 b) The number of possible increasing sequences is seven, as is the number of decreasing
 15455 sequences. The pass number space after disallowing sequences is:

15456
$$5040 - 14 = 5026$$

15457 Based on further information gleaned from the design evidence, the pass number mechanism is
 15458 designed with a terminal locking feature. Upon the sixth failed authentication attempt the terminal
 15459 is locked for one hour. The failed authentication count is reset after five minutes so that an attacker

15460 can at best attempt five pass number entries every five minutes, or 60 pass number entries every
15461 hour.

15462 On average, an attacker would have to enter 2513 pass numbers, over 2513 minutes, before
15463 entering the correct pass number. The average successful attack would, as a result, occur in slightly
15464 less than:

$$\frac{2513min}{60\frac{min}{hour}} \approx 42hours$$

15465

15466 Using the approach to calculate the attack potential, described in the previous subclause, identifies
15467 that it is possible that a layman can defeat the mechanism within days (given easy access to the
15468 TOE), with the use of standard equipment, and with no knowledge of the TOE, giving a value of 1.
15469 Given the resulting sum, 1, the attack potential required to effect a successful attack is not rated, as
15470 it falls below that considered to be Basic.

Annex C

15471
15472
15473
15474

Evaluation Techniques and Tools (informative)

15475 **C.1 Semiformal and formal methods**

15476 In ISO/IEC 15408-3, Annex A.5, supplementary material on formal methods is provided.

15477 **C.1.1 Description of styles**

15478 This section provides general guidance on specification styles. Specific and detailed information is
15479 in those work units under the specific evaluator action elements where examination of the style of
15480 specifications, TSP model and correspondence demonstrations has to be performed.

15481 The ADV class mandates three types of specification styles: informal, semiformal and formal. These
15482 styles are briefly described in the application notes to the ADV class in ISO/IEC 15408-2 The
15483 functional specification and design specification will be written using one or more of these
15484 specification styles. The TSF representations (in the following referred to as specifications) may
15485 use one or more notations in semiformal and formal style. The level of formality of the
15486 correspondence representation depends on the style of the adjacent pair of provided TSF
15487 representation (see the ADV_TDS family for details).

15488 The hierarchy of components within these families increase the formality of the styles

15489 - to reduce the ambiguity of the TSF representation through the hierarchy of
15490 components within the families,

15491 - to reduce the likelihood of refinement errors in the available TSF
15492 representations,

15493 - to strengthen the evidence for correctness of the TSF representations and the
15494 methods for their examination.

15495 The styles are shortly characterised by

15496 - informal style- defined semantics

15497 - semiformal style - defined semantics and syntax

15498 - formal style - defined semantics, syntax and rules of inference.

15499 Regarding the notions of semantics and syntax the degree of precision varies with the style of
15500 description.

15501 Informal descriptions require the semantics to provide meaning to all terms with the help of
15502 natural language explanations.

15503 Semiformal descriptions restrict the syntax formation of terms to well defined expressions having
15504 a precise meaning in the technical community.

15505 Formal style descriptions restrict the semantics and syntax even further: The formation of
15506 syntactical terms follows a formal language description required to be decidable. Examples include
15507 well established implicit formation rules being as precise as the formation of terms and formulas in
15508 first order predicate calculus or formal meta language descriptions using Extended Backus Naur
15509 Form. Apart from informal descriptions the semantics of formal terms is restricted to well
15510 established mathematical models. Formal derivation of theorems is restricted to predefined

15511 inference rules, which are based on well known logical reasoning (classical logic, intuitionistic logic,
 15512 modal logic, temporal logic, etc.). Algorithmic model checking can serve as a substitute for theorem
 15513 proving whenever the reference to well established model checkers is clear and appropriate meta
 15514 theorems are given to guarantee the equivalence to an inference by proof rules.

15515 In the context of the level of formality informal, semiformal and formal styles are considered to be
 15516 hierarchical in nature. Thus, requirements for a informal or semiformal style of specification may
 15517 also be met with either a semiformal or formal specification style provided, that is supported by
 15518 informal, explanatory text where appropriate. The set of presentation elements, syntactic and
 15519 semantic rules is referred in the following as notation. A formal style of presentation uses a formal
 15520 notation and rules of inference which is referred to in the following as formal system.

15521 The content and presentation elements of ADV_FSP and ADV_TDS components describe the style
 15522 in which the presentation of evidence shall be provided by the developer. The evaluator action
 15523 element ADV_x.y.1E requires the evaluator to confirm that the information provided meets all
 15524 requirements for presentation of evidence. If the content and presentation elements require an
 15525 informal style the evaluator may perform the work units for the evaluator action elements in
 15526 parallel with the other work units examining the content of evidence. If the content and
 15527 presentation elements require a semiformal or a formal style this implies the application of
 15528 semiformal or formal methods to examine the content. Therefore it is recommended to perform
 15529 the work units for the evaluator action elements concerning the correct use of the method and its
 15530 rigour before the analysis of the content of evidence. If a notation or their usage in the
 15531 documentation does not provide the level of formality the necessary rigorous methods of analysis
 15532 may be not applicable. The work unit for the evaluator action elements examining the necessary
 15533 informal explanatory text may be performed in parallel with the other work units. Of course the
 15534 evaluator might detect errors in the presentation of evidence during the evaluator action as well
 15535 which result in a fail verdict for the evaluator action elements.

15536 The following text provides a guidance for the examination of specification styles and their use for
 15537 correspondence demonstration in the sub-activities for the assurance families ADV_FSP, ADV_TDS
 15538 and ADV_SPM.

15539

15540 Informal style

15541 An informal specification is one that is expressed in a natural language. If content and presentation
 15542 elements require an informal specification the work unit
 15543 for the evaluator action elements will require the evaluator to determine that it contains all
 15544 necessary informal explanatory text. The evaluator should examine the specification to make sure
 15545 that it

15546 - provides defined meanings of terms, abbreviations and acronyms that are
 15547 used in a context other than that accepted by normal usage,

15548 - if semiformal or formal notations are used appropriate informal, explanatory
 15549 text shall support the understanding.

15550 This enforces the informal specification to provide defined **semantics** of its statements. An
 15551 informal specification uses the ordinary conventions for the natural language i.e. any common
 15552 spoken tongue. It may use figures and semiformal elements of presentation like data flow diagrams
 15553 to illustrate the informal specification. If the specification uses a semiformal notation it will be
 15554 accompanied by supporting explanatory informal text appropriate for unambiguous common
 15555 understanding.

15556 Examples for the use of informal style are:

15557 - ISO/IEC 15408-1 identifies a glossary of terms specific to ISO/IEC
 15558 15408 and reserved terms in accordance with the ISO definitions contained in

- 15559 ISO/IEC Directives Part 2, Rules for the structure and drafting of International
15560 Standards. This clarifies the use of the verbs “shall”, “should”, “may” and “can” in
15561 the context of ISO/IEC 15408
- 15562 - International standards and the Request for Interpretation (RFC) are
15563 specified in an informal style. They use semiformal notations as well e.g. the
15564 abstract syntax notation ASN.1 for specification of message formats.
- 15565 Informal style does not excuse the absence of precision or informal definitions. The evaluator’s
15566 verdict fails if some technical term remains undefined, the evaluators lack of information prevents
15567 decision, or ambiguous interpretations cause confusion.
- 15568
- 15569 **Semiformal style**
- 15570 A semiformal specification is expressed in a restricted syntax language with defined semantics. It
15571 reduces the ambiguity of specification and strengthens the method of analysis.
- 15572 The evaluator should examine the identified notations to make sure that
- 15573 - The syntax rules are defined or a definition is referenced.
- 15574 - The notations with the explanatory text provide a defined **semantics** which
15575 is characterised by
- 15576 a) defined meanings of terms, abbreviations and acronyms that are used
15577 in a context other than that accepted by normal usage,
- 15578 b) the use of a semiformal notation is accompanied by supporting
15579 explanatory text in informal style appropriate for unambiguous meaning,
- 15580 c) expression of rules and characteristics of applicable policies, security
15581 functionality and interfaces (providing details of effects, exceptions and error
15582 messages) of TSF, their subsystems or modules to be specified for the assurance
15583 family for which the notations are used.
- 15584 - The notations contain a restricted **syntax** language which means
- 15585 d) a set of conventions must be supplied to define the restrictions
15586 imposed on the syntax.
- 15587 Examples for the use of semiformal style are:
- 15588 - The restricted syntax language may be a natural language with restricted
15589 sentence structure and keywords with special meanings. -> ISO/IEC 15408-1 and
15590 ISO/IEC 15408-2 provide a semiformal notation for the security functional
15591 requirements consisting of classes, families and components together with rules for
15592 permitted operations. As required by the ECD families of classes ASE and APE, an
15593 explicitly stated IT security requirement shall use the CC requirements components,
15594 families and classes as a model for presentation.
- 15595 - Formally specified languages may be used to define the data structures for
15596 the use of TSFI or an interface of subsystems or modules in semiformal style. Thus
15597 e.g. ISO/IEC 8824 and 8825 define the abstract syntax notation ASN.1 and ISO/IEC
15598 8834 the semantic of the object identifier (OID). ASN.1 makes possible extracting
15599 the encoded information by automated tools (parser). The interface specification
15600 may describe the complete details of all effects caused by interface usage by means
15601 of other semiformal notations e.g. state-transition diagrams.

- 15602 - Diagrams are commonly used for the specification of data-flow, state-
 15603 transition, entity-relation-ship, data or process or program structures in a semiformal
 15604 style, e.g. the Unified Modelling Language (UML) for object-oriented analysis and
 15605 design includes model diagrams, their semantics and an interchange format between
 15606 case tools. The graphical presentation assists the understanding of interaction and
 15607 behaviour of entities depending on events. The abstraction accompanied by the
 15608 graphical presentation normally needs to be compensated by informal description.
 15609 Data-flow and state-transition diagrams may be very helpful, e.g. for the precise
 15610 description and the analysis of protocols.
- 15611 - Programming languages like ANSI C defines a strong syntax and well-
 15612 defined semantics. The source code together with supporting explanatory text and
 15613 documentation of well-defined development tools provides an unambiguous
 15614 semiformal description of the TSF implementation, their security features and
 15615 interfaces. Although having a very high level of formality programming languages
 15616 may be of semiformal styles only because of missing inference rules. But some
 15617 software development tools support also formal methods in software design
 15618 including theorem prover.
- 15619 These examples show that semiformal style covers a wide range of capabilities and level of
 15620 formality. The developer should use appropriate notation for presentation of evidence depending
 15621 on the type of TOE (e.g. hardware, software), the development methodology and the purpose of the
 15622 specification.
- 15623 The semiformal style supports a structured analysis of the content, the consistency, the
 15624 completeness and the correspondence of the representation. A semiformal analysis is one that
 15625 results from a structured approach with a substantial degree of rigor in terms of completeness and
 15626 correctness.
- 15627 A semiformal interface specification supports the evaluator in analysing and assessing the external
 15628 behaviour of a TSF, their subsystems or modules for any input (e.g. to decide about acceptance or
 15629 rejection of a message and its content analysis). Semiformal evidence for conservation of
 15630 properties can be obtained by means of flow charts and state transition diagrams visualizing the
 15631 uniquely defined states and their interrelationship during the course of security preserving
 15632 transitions. The developer may use semiformal notations like software specification languages to
 15633 ensure correct refinement of the specifications from functional specification via high and low level
 15634 design down to the implementation level.
- 15635 This way the semiformal presentation clearly establishes its accuracy and superiority over
 15636 informal descriptions.
- 15637
- 15638 **Formal style**
- 15639 A formal specification is expressed within a formal system based upon well-established
 15640 mathematical concepts. These mathematical concepts are used to define well-defined semantics,
 15641 syntax and rules of inference. A formal system is an abstract system of identities and relations that
 15642 can be described by specifying a formal alphabet, a formal language over that alphabet which is
 15643 based on a formal syntax, and a set of formal rules of inference for constructing derivations of
 15644 sentences in the formal language.
- 15645 The evaluator should examine the identified formal systems to make sure that
- 15646 - The semantics, syntax and inference rules of the formal system are defined
 15647 or a definition is referenced.
- 15648 - Each formal system with the explanatory text provides a defined **semantics**
 15649 which

- 15650 a) provides defined meanings of terms, abbreviations and acronyms that
15651 are used in a context other than that accepted by normal usage,
- 15652 b) the use of a formal system and semiformal notation if any use is
15653 accompanied by supporting explanatory text in informal style appropriate for
15654 unambiguous meaning,
- 15655 c) the formal system is able to express rules and characteristics of
15656 applicable policies, security functionality and interfaces (providing details of
15657 effects, exceptions and error messages) of the TSF, their subsystems or modules to
15658 be specified for the assurance family for which the notations are used.
- 15659 d) the notation provides rules to determine the meaning of syntactical
15660 valid constructs.
- 15661 - Each formal system uses a formal **syntax** that
- 15662 e) provides rules to unambiguously recognise constructs.
- 15663 - Each formal system provides **proof rules** which
- 15664 f) support logical reasoning of well-established mathematical concepts,
15665 g) help to prevent derivation of contradictions.
- 15666 If the developer uses a formal system which is already accepted by the certification body the
15667 evaluator can rely on the level of formality and strength of the system and focus on the
15668 instantiation of the formal system to the TOE specifications and correspondence proofs.

15669 The formal style supports mathematical proofs of the security properties based on the security
15670 features, the consistency of refinements and the correspondence of the representations. Formal
15671 tool support seems adequate whenever manual derivations would otherwise become long winded
15672 and incomprehensible. Formal tools are also apt to reduce the error probability inherent in manual
15673 derivations.

15674 C.1.2 Security policy models and styles

15675 The assurance family Security policy modelling ADV_SPM requires in their components an
15676 increasing level of formality of the TSP model and correspondence demonstration between the TSP
15677 model and the functional specification. The following section provides some guidance how the
15678 general requirements on styles applies to the TSP models.

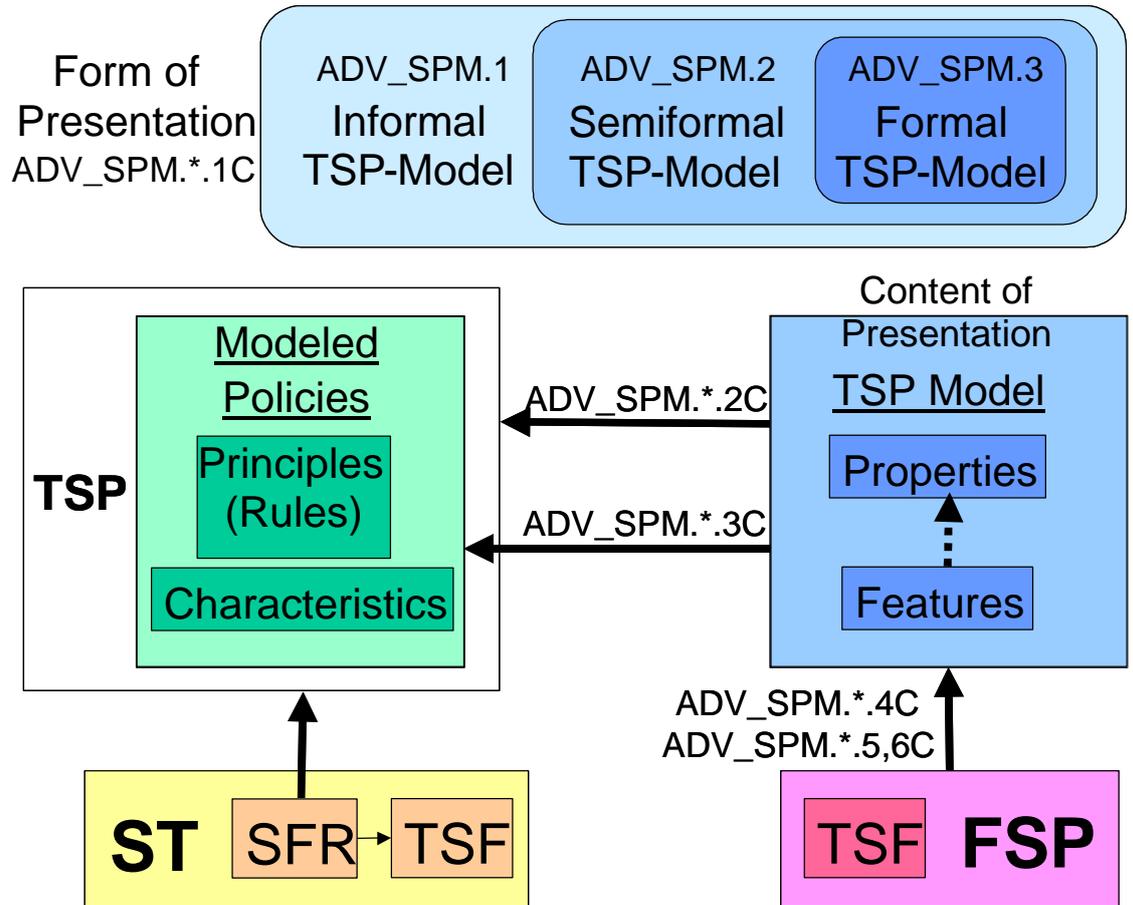
15679 The TOE Security Policy (TSP) is a set of rules and characteristics that regulate how assets are
15680 managed, protected and distributed within a TOE. The TSP can be explicitly stated in the ST by the
15681 SFR (e.g. of families FDP_ACC or FDP_AFC) or be drawn from other SFR (e.g. of classes FAU, FIA or
15682 FPR) claimed in the ST. Although these TSF are provided in semiformal style the policies are
15683 normally described by rules and characteristics in informal style. A TOE security policy model is a
15684 structured representation of security policies to be enforced by the TOE.

15685 According to ADV_SPM.*.2C the TSP model shall model all security policies of the TSP that can be
15686 modelled by the respective level defined by ADV_SPM.*.1C or a rationale shall be given why a lower
15687 level of formality is applied. Thus the TSP model may contain for policy sets of the TSP different
15688 models of different levels of formality as state of the art.

15689 An informal TSP model is a description of the TSP enforced by the security functional requirements
15690 claimed in the ST. All TSP in the ST can be informally modelled.

15691 Modelling means to describe the rules and characteristics of the policies by the properties and
15692 features in the TSP model and to provide evidence that the features imply these properties. The
15693 strength of this evidence depends on the level of formality: an informal model may provide a

15694 rationale but a formal model shall provide a formal proof that the security features imply the
 15695 security properties.



15696
 15697

15698

Figure 1.3 TOE security policy models and correspondence demonstration

15699 The possibility of formally modelling TSPs is dependent on the state of the art. A wide range of
 15700 examples have already been given in the past for successfully modelling Access Control including
 15701 Identification and Authentication. Hence inclusion of access control policies almost always requires
 15702 the developer to provide the model in a formal style.

15703 Whenever in doubt the evaluator should negotiate the type of style (formal, semiformal or
 15704 informal) with the certification body in advance in order to agree upon the state of the art for the
 15705 specific policy under question.

15706