

COMMITTEE DRAFT		Reference document: SC 27 N18701	
ISO/IEC CD 15408-2, revision			
Date: 2018-06-25		Supersedes document WG 3 N1466	
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: 2018-08-20  Please submit your comments via the online balloting application by the due date indicated.	
ISO/IEC CD 15408-2, revision			
Title: IT Security techniques – Evaluation criteria for IT security — Part 2: Security functional components			
Project: ISO/IEC 15408-2 (revision)			
Explanatory Report			
Status	SC 27 Decision	Reference documents	
		Input	Output
For details regarding previous development stages refer to 2 <sup>nd</sup> page of this explanatory report.			
ISO/IEC NP 15408-2 (revision) Evaluation criteria for IT security -- Part 2 NWIP	53 <sup>rd</sup> 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 (N16964 [replaces N16883]).
ISO/IEC 15408-2 1 <sup>st</sup> WD	54 <sup>th</sup> WG 3 meeting, April 2017, Recommendations 5,10, 11, 14 (N17041 = WG 3 N1413).	Results of call f. editor (N17276); SoV (N17026).	Call f. project editor (N17319); Liaisons to: CCDB (WG 3 N1391); The Open Group (WG 3 N1394); ISO/TC 22/SC 32 (N17373); Text f. 1 <sup>st</sup> WD (WG 3 N1436).
ISO/IEC NP 15408-2 (revision) 2 <sup>nd</sup> WD	55 <sup>th</sup> WG 3 meeting, October / November 2017, Recommendations 8, 10 (N17666 = WG 3 N1494).	Results of call f. editor (N17389); SoCom (WG 3 N1464); Draft DoC (WG 3 N1501).	Call / NB nomination for /of (N17319 / N17389); Editor's report (WG 3 N1465); Liaisons to: CCDB (WG 3 N1455); ISO/TC 22/SC 32 (N18103); DoC (WG 3 N1462); Text f. 2nd WD (WG 3 N1466).
ISO/IEC 15408-2 1 <sup>st</sup> CD	56 <sup>th</sup> WG 3 meeting, April 2018, Recommendations 8, 10 (N18471 = WG 3 N1557) / 30 <sup>th</sup> SC 27 Plenary, April 2018, Resolution 6 (N18710)	SoCom (WG 3 N1528); Late Com (WG 3 N1563).	Liaison to: CCDB (WG 3 N1521); DoC (WG 3 N1527); Text f. 1 <sup>st</sup> CD (N18701).
CD Registration and Consideration			
In accordance with resolution 6 (see SC 27 N18710) of the 30th SC 27 Plenary meeting held in Wuhan, China, 2018-04-23/24 the hereby attached document has been registered with the ISO Central Secretariat (ITTF) as 1st Committee Draft (CD) and is being circulated for a 1st CD 8 weeks letter ballot closing by			
2018-08-20			
Medium: <a href="http://isotc.iso.org/livelink/livelink/open/jtc1sc27">http://isotc.iso.org/livelink/livelink/open/jtc1sc27</a>			
No. of pages: 2 + 290			

Explanatory Report (2 <sup>nd</sup> page)			
Status	SC 27 Decision	Reference documents	
		Input	Output
Study Period IT security testing, evaluation and assurance standards and techniques	51 <sup>st</sup> WG 3 meeting, Oct. 2015, Recommendations 5, 6 (N15594 = WG 3 N1251).		Terms of Reference (WG 5 N1258); 1 <sup>st</sup> /2 <sup>nd</sup> call f. contr. (WG 3 N1259 /1317)..
	52 <sup>nd</sup> WG 3 meeting, April 2016, Recommendation 5, 7 (N16026 = WG 3 N1296).	Expert contr. (WG 3 N1299, 1301).	3 <sup>rd</sup> call f. contr. (WG 3 N1377); Rapporteur's report (WG 3 N1320); Liaison to: CCDB (WG 3 N = N1266).

ISO/IEC JTC 1/SC 27/WG 3 N18071

Date: 2018-06-22

ISO/IEC WD 15408-2:####(EN)

ISO/IEC JTC 1/SC 27 IT Security techniques

Secretariat: DIN

IT security techniques — Evaluation criteria for IT security — Part 2: Security  
functional components

*Techniques de sécurité IT — Critères d'évaluation pour la sécurité des technologies de  
l'information — Partie 2 : Composants fonctionnels de sécurité*

CD stage

**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.

**READ ME FIRST**

Editors general notes for this draft.

Red text in a box are the Editors comments.

In this draft the editors highlighted the keywords relating to the ISO verbal forms, **shall, should, may, can** and **must** using green text in order to highlight these words. This convention will be removed before the FDIS level documents.

In this first CD the editors have reviewed the use of the above verbal forms and have made a few recommended changes that reflect the correct usage within ISO documents. Reviewers should pay attention to this in case the editors have made mistakes in their determination. These have been indicated with the old form in strikeout and the suggested change. E.g. "~~shall~~ must" Indicating that the editors recommend replacing "shall" by "must"

The Editors are prepared to organize a meeting on this topic, as well as the normative/informative status of the annexes.

Some editorial changes have also been introduced in order to comply with the [ISO/IEC Directives part 2:2018](#)

The Editors have restructured the document in order to present the information more effectively and simplified the use of English vocabulary and grammar for consistency. This document is read by many people whose first language is not English and that the document will be translated into other languages.

The editors are aware that the figures are of low quality. In the final documents high quality images will be used. The Editors hope that they are legible in this draft

The Editors thank the WG 3 contributors for their contributions and support during the editing cycle.

**Legal Notice:**

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

44	<b>Contents</b>	<b>Page</b>
45	<b>Foreword .....</b>	<b>xxi</b>
46	<b>Introduction.....</b>	<b>xxii</b>
47	<b>1 Scope .....</b>	<b>1</b>
48	<b>2 Normative references.....</b>	<b>1</b>
49	<b>3 Terms and Definitions.....</b>	<b>1</b>
50	<b>4 Overview .....</b>	<b>1</b>
51	<b>4.1 Organization of this document .....</b>	<b>2</b>
52	<b>5 Functional requirements paradigm .....</b>	<b>2</b>
53	<b>6 Security functional components.....</b>	<b>6</b>
54	<b>6.1 Overview .....</b>	<b>6</b>
55	<b>6.1.1 Class structure.....</b>	<b>6</b>
56	<b>6.1.2 Family structure .....</b>	<b>7</b>
57	<b>6.1.3 Component structure .....</b>	<b>9</b>
58	<b>6.2 Component catalogue.....</b>	<b>10</b>
59	<b>6.2.1 Component changes highlighting.....</b>	<b>11</b>
60	<b>7 Class FAU: Security audit.....</b>	<b>12</b>
61	<b>7.1 Class description.....</b>	<b>12</b>
62	<b>7.2 Security audit automatic response (FAU_ARP) .....</b>	<b>13</b>
63	<b>7.2.1 Family behaviour.....</b>	<b>13</b>
64	<b>7.2.2 Components leveling and description .....</b>	<b>13</b>
65	<b>7.2.3 Management of FAU_ARP.1 .....</b>	<b>13</b>
66	<b>7.2.4 Audit of FAU_ARP.1 .....</b>	<b>13</b>
67	<b>7.2.5 FAU_ARP.1 Security alarms.....</b>	<b>13</b>
68	<b>7.3 Security audit data generation (FAU_GEN).....</b>	<b>13</b>
69	<b>7.3.1 Family behaviour.....</b>	<b>13</b>
70	<b>7.3.2 Components leveling and description .....</b>	<b>14</b>
71	<b>7.3.3 Management of FAU_GEN.1, FAU_GEN.2.....</b>	<b>14</b>
72	<b>7.3.4 Audit of FAU_GEN.1, FAU_GEN.2 .....</b>	<b>14</b>
73	<b>7.3.5 FAU_GEN.1 Audit data generation.....</b>	<b>14</b>
74	<b>7.3.6 FAU_GEN.2 User identity association .....</b>	<b>15</b>
75	<b>7.4 Security audit analysis (FAU_SAA).....</b>	<b>15</b>
76	<b>7.4.1 Family behaviour.....</b>	<b>15</b>
77	<b>7.4.2 Components leveling and description .....</b>	<b>15</b>
78	<b>7.4.3 Management of FAU_SAA.1.....</b>	<b>16</b>
79	<b>7.4.4 Management of FAU_SAA.2.....</b>	<b>16</b>
80	<b>7.4.5 Management of FAU_SAA.3.....</b>	<b>16</b>

81	<b>7.4.6</b>	<b>Management of FAU_SAA.4 .....</b>	<b>16</b>
82	<b>7.4.7</b>	<b>Audit of FAU_SAA.1, FAU_SAA.2, FAU_SAA.3, FAU_SAA.4.....</b>	<b>16</b>
83	<b>7.4.8</b>	<b>FAU_SAA.1 Potential violation analysis.....</b>	<b>16</b>
84	<b>7.4.9</b>	<b>FAU_SAA.2 Profile based anomaly detection.....</b>	<b>16</b>
85	<b>7.4.10</b>	<b>FAU_SAA.3 Simple attack heuristics.....</b>	<b>17</b>
86	<b>7.4.11</b>	<b>FAU_SAA.4 Complex attack heuristics .....</b>	<b>17</b>
87	<b>7.5</b>	<b>Security audit review (FAU_SAR).....</b>	<b>18</b>
88	<b>7.5.1</b>	<b>Family behaviour .....</b>	<b>18</b>
89	<b>7.5.2</b>	<b>Components leveling and description.....</b>	<b>18</b>
90	<b>7.5.3</b>	<b>Management of FAU_SAR.1 .....</b>	<b>18</b>
91	<b>7.5.4</b>	<b>Management of FAU_SAR.2, FAU_SAR.3.....</b>	<b>18</b>
92	<b>7.5.5</b>	<b>Audit of FAU_SAR.1 .....</b>	<b>18</b>
93	<b>7.5.6</b>	<b>Audit of FAU_SAR.2 .....</b>	<b>19</b>
94	<b>7.5.7</b>	<b>Audit of FAU_SAR.3 .....</b>	<b>19</b>
95	<b>7.5.8</b>	<b>FAU_SAR.1 Audit review.....</b>	<b>19</b>
96	<b>7.5.9</b>	<b>FAU_SAR.2 Restricted audit review.....</b>	<b>19</b>
97	<b>7.5.10</b>	<b>FAU_SAR.3 Selectable audit review.....</b>	<b>19</b>
98	<b>7.6</b>	<b>Security audit event selection (FAU_SEL).....</b>	<b>19</b>
99	<b>7.6.1</b>	<b>Family behaviour .....</b>	<b>19</b>
100	<b>7.6.2</b>	<b>Components leveling and description.....</b>	<b>20</b>
101	<b>7.6.3</b>	<b>Management of FAU_SEL.1 .....</b>	<b>20</b>
102	<b>7.6.4</b>	<b>Audit of FAU_SEL.1 .....</b>	<b>20</b>
103	<b>7.6.5</b>	<b>FAU_SEL.1 Selective audit .....</b>	<b>20</b>
104	<b>7.7</b>	<b>Security audit data storage (FAU_STG) .....</b>	<b>21</b>
105	<b>7.7.1</b>	<b>Family behaviour .....</b>	<b>21</b>
106	<b>7.7.2</b>	<b>Components leveling and description.....</b>	<b>21</b>
107	<b>7.7.3</b>	<b>Management of FAU_STG.1 .....</b>	<b>21</b>
108	<b>7.7.4</b>	<b>Management of FAU_STG.2 .....</b>	<b>21</b>
109	<b>7.7.5</b>	<b>Management of FAU_STG.3 .....</b>	<b>21</b>
110	<b>7.7.6</b>	<b>Management of FAU_STG.4 .....</b>	<b>21</b>
111	<b>7.7.7</b>	<b>Management of FAU_STG.5 .....</b>	<b>22</b>
112	<b>7.7.8</b>	<b>Audit of FAU_STG.1.....</b>	<b>22</b>
113	<b>7.7.9</b>	<b>Audit of FAU_STG.2, FAU_STG.4 .....</b>	<b>22</b>
114	<b>7.7.10</b>	<b>Audit of FAU_STG.3.....</b>	<b>22</b>
115	<b>7.7.11</b>	<b>Audit of FAU_STG.5.....</b>	<b>22</b>
116	<b>7.7.12</b>	<b>FAU_STG.1 Audit data storage location.....</b>	<b>22</b>
117	<b>7.7.13</b>	<b>FAU_STG.2 Protected audit data storage .....</b>	<b>22</b>
118	<b>7.7.14</b>	<b>FAU_STG.3 Guarantees of audit data availability .....</b>	<b>23</b>

119	7.7.15	FAU_STG.4 Prevention of audit data loss .....	23
120	7.7.16	FAU_STG.5 Action in case of possible audit data loss.....	23
121	8	Class FCO: Communication.....	24
122	8.1	Class description.....	24
123	8.2	Non-repudiation of origin (FCO_NRO) .....	24
124	8.2.1	Family behaviour.....	24
125	8.2.2	Components leveling and description .....	24
126	8.2.3	Management of FCO_NRO.1, FCO_NRO.2 .....	24
127	8.2.4	Audit of FCO_NRO.1 .....	25
128	8.2.5	Audit of FCO_NRO.2 .....	25
129	8.2.6	FCO_NRO.1 Selective proof of origin.....	25
130	8.2.7	FCO_NRO.2 Enforced proof of origin.....	25
131	8.3	Non-repudiation of receipt (FCO_NRR) .....	26
132	8.3.1	Family behaviour.....	26
133	8.3.2	Components leveling and description .....	26
134	8.3.3	Management of FCO_NRR.1, FCO_NRR.2 .....	26
135	8.3.4	Audit of FCO_NRR.1 .....	26
136	8.3.5	Audit of FCO_NRR.2 .....	27
137	8.3.6	FCO_NRR.1 Selective proof of receipt.....	27
138	8.3.7	FCO_NRR.2 Enforced proof of receipt.....	27
139	<del>8.4</del>	<del>Trusted channel (FCO_TCC) .....</del>	<del>28</del>
140	<del>8.4.1</del>	<del>Family behaviour.....</del>	<del>28</del>
141	<del>8.4.2</del>	<del>Components leveling and description .....</del>	<del>28</del>
142	<del>8.4.3</del>	<del>Management of FCO_TCC.1, FCO_TCC.2 .....</del>	<del>28</del>
143	<del>8.4.4</del>	<del>Audit of FCO_TCC.1, FCO_TCC.2 .....</del>	<del>28</del>
144	<del>8.4.5</del>	<del>FCO_TCC.1 Trusted Communication Channel with fixed security properties....</del>	<del>29</del>
145	<del>8.4.6</del>	<del>FCO_TCC.2 Trusted Communication Channel with selectable security</del>	
146		<del>properties .....</del>	<del>29</del>
147	9	Class FCS: Cryptographic support .....	31
148	9.1	Class description.....	31
149	9.2	Cryptographic key management (FCS_CKM).....	31
150	9.2.1	Family behaviour.....	31
151	9.2.2	Components leveling and description .....	32
152	9.2.3	Management of FCS_CKM.1, FCS_CKM.2, FCS_CKM.3, FCS_CKM.5, CKM.6 .....	32
153	9.2.4	Audit of FCS_CKM.1, FCS_CKM.2, FCS_CKM.3, FCS_CKM.5, CKM.6 .....	32
154	9.2.5	FCS_CKM.1 Cryptographic key generation.....	32
155	9.2.6	FCS_CKM.2 Cryptographic key distribution.....	33
156	9.2.7	FCS_CKM.3 Cryptographic key access.....	33

157	9.2.8	FCS_CKM.4 Cryptographic key destruction.....	33
158	9.2.9	FCS_CKM.5 Cryptographic key derivation.....	34
159	9.2.10	FCS_CKM.6 Timing and event of cryptographic key destruction.....	34
160	9.3	Cryptographic operation (FCS_COP).....	34
161	9.3.1	Family behaviour.....	34
162	9.3.2	Components leveling and description.....	35
163	9.3.3	Management of FCS_COP.1.....	35
164	9.3.4	Audit of FCS_COP.1.....	35
165	9.3.5	FCS_COP.1 Cryptographic operation.....	35
166	9.4	Random bit generation (FCS_RBG).....	35
167	9.4.1	Family behaviour.....	35
168	9.4.2	Components leveling and description.....	36
169	9.4.3	Management of FCS_RBG.1, FCS_RBG.2, FCS_RBG.3, FCS_RBG.4, FCS_RBG.5, 170 FCS_RBG.6.....	36
171	9.4.4	Audit of FCS_RBG.1, FCS_RBG.2.....	36
172	9.4.5	Audit of FCS_RBG.3, FCS_RBG.4, FCS_RBG.6, FCS_RBG.6.....	36
173	9.4.6	FCS_RBG.1 Random bit generation (RBG).....	37
174	9.4.7	FCS_RBG.2 Random bit generation (external seeding).....	37
175	9.4.8	FCS_RBG.3 Random bit generation (internal seeding – single source).....	37
176	9.4.9	FCS_RBG.4 Random bit generation (internal seeding – multiple sources).....	38
177	9.4.10	FCS_RBG.5 Random bit generation (combining entropy sources).....	38
178	9.4.11	FCS_RBG.6 Random bit generation service.....	38
179	9.5	Generation of random numbers (FCS_RNG).....	38
180	9.5.1	Family behaviour.....	39
181	9.5.2	Components leveling and description.....	39
182	9.5.3	Management of FCS_RNG.1.....	39
183	9.5.4	Audit of FCS_RNG.1.....	39
184	9.5.5	FCS_RNG.1 Random number generation.....	39
185	10	Class FDP: User data protection.....	40
186	10.1	Class description.....	40
187	10.2	Access control policy (FDP_ACC).....	41
188	10.2.1	Family behaviour.....	41
189	10.2.2	Components leveling and description.....	42
190	10.2.3	Management of FDP_ACC.1, FDP_ACC.2.....	42
191	10.2.4	Audit of FDP_ACC.1, FDP_ACC.2.....	42
192	10.2.5	FDP_ACC.1 Subset access control.....	42
193	10.2.6	FDP_ACC.2 Complete access control.....	42
194	10.3	Access control functions (FDP_ACF).....	43



195	<b>10.3.1 Family behaviour.....</b>	<b>43</b>
196	<b>10.3.2 Components leveling and description.....</b>	<b>43</b>
197	<b>10.3.3 Management of FDP_ACF.1.....</b>	<b>43</b>
198	<b>10.3.4 Audit of FDP_ACF.1.....</b>	<b>43</b>
199	<b>10.3.5 FDP_ACF.1 Security attribute-based access control.....</b>	<b>43</b>
200	<b>10.4 Data authentication (FDP_DAU).....</b>	<b>44</b>
201	<b>10.4.1 Family behaviour.....</b>	<b>44</b>
202	<b>10.4.2 Components leveling and description.....</b>	<b>44</b>
203	<b>10.4.3 Management of FDP_DAU.1, FDP_DAU.2.....</b>	<b>44</b>
204	<b>10.4.4 Audit of FDP_DAU.1.....</b>	<b>44</b>
205	<b>10.4.5 Audit of FDP_DAU.2.....</b>	<b>45</b>
206	<b>10.4.6 FDP_DAU.1 Basic Data Authentication.....</b>	<b>45</b>
207	<b>10.4.7 FDP_DAU.2 Data Authentication with Identity of Guarantor.....</b>	<b>45</b>
208	<b>10.5 Export from the TOE (FDP_ETC).....</b>	<b>45</b>
209	<b>10.5.1 Family behaviour.....</b>	<b>45</b>
210	<b>10.5.2 Components leveling and description.....</b>	<b>46</b>
211	<b>10.5.3 Management of FDP_ETC.1.....</b>	<b>46</b>
212	<b>10.5.4 Management of FDP_ETC.2.....</b>	<b>46</b>
213	<b>10.5.5 Audit of FDP_ETC.1, FDP_ETC.2.....</b>	<b>46</b>
214	<b>10.5.6 FDP_ETC.1 Export of user data without security attributes.....</b>	<b>46</b>
215	<b>10.5.7 FDP_ETC.2 Export of user data with security attributes.....</b>	<b>46</b>
216	<b>10.6 Information flow control policy (FDP_IFC).....</b>	<b>47</b>
217	<b>10.6.1 Family behaviour.....</b>	<b>47</b>
218	<b>10.6.2 Components leveling and description.....</b>	<b>47</b>
219	<b>10.6.3 Management of FDP_IFC.1, FDP_IFC.2.....</b>	<b>48</b>
220	<b>10.6.4 Audit of FDP_IFC.1, FDP_IFC.2.....</b>	<b>48</b>
221	<b>10.6.5 FDP_IFC.1 Subset information flow control.....</b>	<b>48</b>
222	<b>10.6.6 FDP_IFC.2 Complete information flow control.....</b>	<b>48</b>
223	<b>10.7 Information flow control functions (FDP_IFF).....</b>	<b>48</b>
224	<b>10.7.1 Family behaviour.....</b>	<b>48</b>
225	<b>10.7.2 Components leveling and description.....</b>	<b>49</b>
226	<b>10.7.3 Management of FDP_IFF.1, FDP_IFF.2.....</b>	<b>49</b>
227	<b>10.7.4 Management of FDP_IFF.3, FDP_IFF.4, FDP_IFF.5.....</b>	<b>49</b>
228	<b>10.7.5 Management of FDP_IFF.6.....</b>	<b>49</b>
229	<b>10.7.6 Audit of FDP_IFF.1, FDP_IFF.2, FDP_IFF.5.....</b>	<b>49</b>
230	<b>10.7.7 Audit of FDP_IFF.3, FDP_IFF.4, FDP_IFF.6.....</b>	<b>50</b>
231	<b>10.7.8 FDP_IFF.1 Simple security attributes.....</b>	<b>50</b>
232	<b>10.7.9 FDP_IFF.2 Hierarchical security attributes.....</b>	<b>51</b>

233	<b>10.7.10</b>	<b>FDP_IFF.3 Limited illicit information flows .....</b>	<b>52</b>
234	<b>10.7.11</b>	<b>FDP_IFF.4 Partial elimination of illicit information flows .....</b>	<b>52</b>
235	<b>10.7.12</b>	<b>FDP_IFF.5 No illicit information flows .....</b>	<b>52</b>
236	<b>10.7.13</b>	<b>FDP_IFF.6 Illicit information flow monitoring .....</b>	<b>52</b>
237	<b>10.8</b>	<b>Information Retention Control (FDP_IRC) .....</b>	<b>53</b>
238	<b>10.8.1</b>	<b>Family behaviour .....</b>	<b>53</b>
239	<b>10.8.2</b>	<b>Components leveling and description.....</b>	<b>53</b>
240	<b>10.8.3</b>	<b>Management of FDP_IRC.1 .....</b>	<b>54</b>
241	<b>10.8.4</b>	<b>Audit of FDP_IRC.1.....</b>	<b>54</b>
242	<b>10.8.5</b>	<b>FDP_IRC.1 Subset information control .....</b>	<b>54</b>
243	<b>10.8.6</b>	<b>FDP_IRC.2 Complete information control.....</b>	<b>54</b>
244	<b>10.9</b>	<b>Import from outside of the TOE (FDP_ITC) .....</b>	<b>54</b>
245	<b>10.9.1</b>	<b>Family behaviour .....</b>	<b>54</b>
246	<b>10.9.2</b>	<b>Components leveling and description.....</b>	<b>55</b>
247	<b>10.9.3</b>	<b>Management of FDP_ITC.1, FDP_ITC.2 .....</b>	<b>55</b>
248	<b>10.9.4</b>	<b>Audit of FDP_ITC.1, FDP_ITC.2 .....</b>	<b>55</b>
249	<b>10.9.5</b>	<b>FDP_ITC.1 Import of user data without security attributes .....</b>	<b>55</b>
250	<b>10.9.6</b>	<b>FDP_ITC.2 Import of user data with security attributes.....</b>	<b>56</b>
251	<b>10.10</b>	<b>Internal TOE transfer (FDP_ITT).....</b>	<b>56</b>
252	<b>10.10.1</b>	<b>Family behaviour .....</b>	<b>56</b>
253	<b>10.10.2</b>	<b>Components leveling and description.....</b>	<b>56</b>
254	<b>10.10.3</b>	<b>Management of FDP_ITT.1, FDP_ITT.2.....</b>	<b>57</b>
255	<b>10.10.4</b>	<b>Management of FDP_ITT.3, FDP_ITT.4.....</b>	<b>57</b>
256	<b>10.10.5</b>	<b>Audit of FDP_ITT.1, FDP_ITT.2 .....</b>	<b>57</b>
257	<b>10.10.6</b>	<b>Audit of FDP_ITT.3, FDP_ITT.4.....</b>	<b>57</b>
258	<b>10.10.7</b>	<b>FDP_ITT.1 Basic internal transfer protection .....</b>	<b>57</b>
259	<b>10.10.8</b>	<b>FDP_ITT.2 Transmission separation by attribute.....</b>	<b>58</b>
260	<b>10.10.9</b>	<b>FDP_ITT.3 Integrity monitoring.....</b>	<b>58</b>
261	<b>10.10.10</b>	<b>FDP_ITT.4 Attribute-based integrity monitoring.....</b>	<b>58</b>
262	<b>10.11</b>	<b>Residual information protection (FDP_RIP).....</b>	<b>59</b>
263	<b>10.11.1</b>	<b>Family behaviour .....</b>	<b>59</b>
264	<b>10.11.2</b>	<b>Components leveling and description.....</b>	<b>59</b>
265	<b>10.11.3</b>	<b>Management of FDP_RIP.1, FDP_RIP.2.....</b>	<b>59</b>
266	<b>10.11.4</b>	<b>Audit of FDP_RIP.1, FDP_RIP.2 .....</b>	<b>59</b>
267	<b>10.11.5</b>	<b>FDP_RIP.1 Subset residual information protection .....</b>	<b>59</b>
268	<b>10.11.6</b>	<b>FDP_RIP.2 Full residual information protection .....</b>	<b>60</b>
269	<b>10.12</b>	<b>Rollback (FDP_ROL).....</b>	<b>60</b>
270	<b>10.12.1</b>	<b>Family behaviour .....</b>	<b>60</b>

271	<b>10.12.2</b>	<b>Components leveling and description .....</b>	<b>60</b>
272	<b>10.12.3</b>	<b>Management of FDP_ROL.1, FDP_ROL.2 .....</b>	<b>60</b>
273	<b>10.12.4</b>	<b>Audit of FDP_ROL.1, FDP_ROL.2 .....</b>	<b>60</b>
274	<b>10.12.5</b>	<b>FDP_ROL.1 Basic rollback.....</b>	<b>61</b>
275	<b>10.12.6</b>	<b>FDP_ROL.2 Advanced rollback.....</b>	<b>61</b>
276	<b>10.13</b>	<b>Stored data confidentiality (FDP_SDC) .....</b>	<b>61</b>
277	<b>10.13.1</b>	<b>Family behaviour.....</b>	<b>61</b>
278	<b>10.13.2</b>	<b>Components leveling and description .....</b>	<b>61</b>
279	<b>10.13.3</b>	<b>Management of FDP_SDC.1, FDP_SDC.2.....</b>	<b>62</b>
280	<b>10.13.4</b>	<b>Audit of FDP_SDC.1, FDP_SDC.2 .....</b>	<b>62</b>
281	<b>10.13.5</b>	<b>FDP_SDC.1 Stored data confidentiality .....</b>	<b>62</b>
282	<b>10.13.6</b>	<b>FDP_SDC.2 Stored data confidentiality with dedicated method .....</b>	<b>62</b>
283	<b>10.13.7</b>	<b>FDP_SDC.3 Stored data confidentiality with user credential.....</b>	<b>62</b>
284	<b>10.14</b>	<b>Stored data integrity (FDP_SDI).....</b>	<b>63</b>
285	<b>10.14.1</b>	<b>Family behaviour.....</b>	<b>63</b>
286	<b>10.14.2</b>	<b>Components leveling and description .....</b>	<b>63</b>
287	<b>10.14.3</b>	<b>Management of FDP_SDI.1.....</b>	<b>63</b>
288	<b>10.14.4</b>	<b>Management of FDP_SDI.2.....</b>	<b>63</b>
289	<b>10.14.5</b>	<b>Audit of FDP_SDI.1 .....</b>	<b>63</b>
290	<b>10.14.6</b>	<b>Audit of FDP_SDI.2 .....</b>	<b>63</b>
291	<b>10.14.7</b>	<b>FDP_SDI.1 Stored data integrity monitoring.....</b>	<b>64</b>
292	<b>10.14.8</b>	<b>FDP_SDI.2 Stored data integrity monitoring and action.....</b>	<b>64</b>
293	<b>10.15</b>	<b>Inter-TSF user data confidentiality transfer protection (FDP_UCT) .....</b>	<b>64</b>
294	<b>10.15.1</b>	<b>Family behaviour.....</b>	<b>64</b>
295	<b>10.15.2</b>	<b>Components leveling and description .....</b>	<b>64</b>
296	<b>10.15.3</b>	<b>Management of FDP_UCT.1 .....</b>	<b>64</b>
297	<b>10.15.4</b>	<b>Audit of FDP_UCT.1.....</b>	<b>65</b>
298	<b>10.15.5</b>	<b>FDP_UCT.1 Basic data exchange confidentiality .....</b>	<b>65</b>
299	<b>10.16</b>	<b>Inter-TSF user data integrity transfer protection (FDP_UIT).....</b>	<b>65</b>
300	<b>10.16.1</b>	<b>Family behaviour.....</b>	<b>65</b>
301	<b>10.16.2</b>	<b>Components leveling and description .....</b>	<b>65</b>
302	<b>10.16.3</b>	<b>Management of FDP_UIT.1, FDP_UIT.2, FDP_UIT.3 .....</b>	<b>66</b>
303	<b>10.16.4</b>	<b>Audit of FDP_UIT.1.....</b>	<b>66</b>
304	<b>10.16.5</b>	<b>Audit of FDP_UIT.2, FDP_UIT.3 .....</b>	<b>66</b>
305	<b>10.16.6</b>	<b>FDP_UIT.1 Data exchange integrity .....</b>	<b>66</b>
306	<b>10.16.7</b>	<b>FDP_UIT.2 Source data exchange recovery .....</b>	<b>67</b>
307	<b>10.16.8</b>	<b>FDP_UIT.3 Destination data exchange recovery .....</b>	<b>67</b>
308	<b>11</b>	<b>Class FIA: Identification and authentication.....</b>	<b>68</b>

309	<b>11.1 Class description .....</b>	<b>68</b>
310	<b>11.2 Authentication failures (FIA_AFL).....</b>	<b>69</b>
311	<b>11.2.1 Family behaviour .....</b>	<b>69</b>
312	<b>11.2.2 Components leveling and description.....</b>	<b>69</b>
313	<b>11.2.3 Management of FIA_AFL.1.....</b>	<b>69</b>
314	<b>11.2.4 Audit of FIA_AFL.1 .....</b>	<b>69</b>
315	<b>11.2.5 FIA_AFL.1 Authentication failure handling.....</b>	<b>69</b>
316	<b>11.3 Authentication proof of identity (FIA_API) .....</b>	<b>70</b>
317	<b>11.3.1 Family behaviour .....</b>	<b>70</b>
318	<b>11.3.2 Components leveling and description.....</b>	<b>70</b>
319	<b>11.3.3 Management of FIA_API.1 .....</b>	<b>70</b>
320	<b>11.3.4 Management of FIA_API.1 .....</b>	<b>70</b>
321	<b>11.3.5 Audit of FIA_API.1.....</b>	<b>70</b>
322	<b>11.3.6 FIA_API.1 Authentication proof of identity .....</b>	<b>70</b>
323	<b>11.4 User attribute definition (FIA_ATD).....</b>	<b>70</b>
324	<b>11.4.1 Family behaviour .....</b>	<b>70</b>
325	<b>11.4.2 Components leveling and description.....</b>	<b>71</b>
326	<b>11.4.3 Management of FIA_ATD.1.....</b>	<b>71</b>
327	<b>11.4.4 Audit of FIA_ATD.1 .....</b>	<b>71</b>
328	<b>11.4.5 FIA_ATD.1 User attribute definition.....</b>	<b>71</b>
329	<b>11.5 Specification of secrets (FIA_SOS).....</b>	<b>71</b>
330	<b>11.5.1 Family behaviour .....</b>	<b>71</b>
331	<b>11.5.2 Components leveling and description.....</b>	<b>71</b>
332	<b>11.5.3 Management of FIA_SOS.1 .....</b>	<b>72</b>
333	<b>11.5.4 Management of FIA_SOS.2 .....</b>	<b>72</b>
334	<b>11.5.5 Audit of FIA_SOS.1, FIA_SOS.2.....</b>	<b>72</b>
335	<b>11.5.6 FIA_SOS.1 Verification of secrets.....</b>	<b>72</b>
336	<b>11.5.7 FIA_SOS.2 TSF Generation of secrets.....</b>	<b>72</b>
337	<b>11.6 User authentication (FIA_UAU) .....</b>	<b>72</b>
338	<b>11.6.1 Family behaviour .....</b>	<b>72</b>
339	<b>11.6.2 Components leveling and description.....</b>	<b>73</b>
340	<b>11.6.3 Management of FIA_UAU.1.....</b>	<b>73</b>
341	<b>11.6.4 Management of FIA_UAU.2.....</b>	<b>73</b>
342	<b>11.6.5 Management of FIA_UAU.3, FIA_UAU.4, FIA_UAU.7 .....</b>	<b>73</b>
343	<b>11.6.6 Management of FIA_UAU.5.....</b>	<b>74</b>
344	<b>11.6.7 Management of FIA_UAU.6.....</b>	<b>74</b>
345	<b>11.6.8 Management of FIA_UAU.7.....</b>	<b>74</b>
346	<b>11.6.9 Audit of FIA_UAU.1 .....</b>	<b>74</b>

347	<b>11.6.10</b>	<b>Audit of FIA_UAU.2 .....</b>	<b>74</b>
348	<b>11.6.11</b>	<b>Audit of FIA_UAU.3 .....</b>	<b>74</b>
349	<b>11.6.12</b>	<b>Audit of FIA_UAU.4 .....</b>	<b>74</b>
350	<b>11.6.13</b>	<b>Audit of FIA_UAU.5 .....</b>	<b>74</b>
351	<b>11.6.14</b>	<b>Audit of FIA_UAU.6 .....</b>	<b>74</b>
352	<b>11.6.15</b>	<b>Audit of FIA_UAU.7 .....</b>	<b>75</b>
353	<b>11.6.16</b>	<b>FIA_UAU.1 Timing of authentication .....</b>	<b>75</b>
354	<b>11.6.17</b>	<b>FIA_UAU.2 User authentication before any action.....</b>	<b>75</b>
355	<b>11.6.18</b>	<b>FIA_UAU.3 Unforgeable authentication .....</b>	<b>75</b>
356	<b>11.6.19</b>	<b>FIA_UAU.4 Single-use authentication mechanisms .....</b>	<b>76</b>
357	<b>11.6.20</b>	<b>FIA_UAU.5 Multiple authentication mechanisms.....</b>	<b>76</b>
358	<b>11.6.21</b>	<b>FIA_UAU.6 Re-authenticating .....</b>	<b>76</b>
359	<b>11.6.22</b>	<b>FIA_UAU.7 Protected authentication feedback .....</b>	<b>76</b>
360	<b>11.7</b>	<b>User identification (FIA_UID) .....</b>	<b>77</b>
361	<b>11.7.1</b>	<b>Family behaviour.....</b>	<b>77</b>
362	<b>11.7.2</b>	<b>Components leveling and description .....</b>	<b>77</b>
363	<b>11.7.3</b>	<b>Management of FIA_UID.1 .....</b>	<b>77</b>
364	<b>11.7.4</b>	<b>Management of FIA_UID.2 .....</b>	<b>77</b>
365	<b>11.7.5</b>	<b>Audit of FIA_UID.1, FIA_UID.2 .....</b>	<b>77</b>
366	<b>11.7.6</b>	<b>FIA_UID.1 Timing of identification.....</b>	<b>77</b>
367	<b>11.7.7</b>	<b>FIA_UID.2 User identification before any action.....</b>	<b>78</b>
368	<b>11.8</b>	<b>User-subject binding (FIA_USB).....</b>	<b>78</b>
369	<b>11.8.1</b>	<b>Family behaviour.....</b>	<b>78</b>
370	<b>11.8.2</b>	<b>Components leveling and description .....</b>	<b>78</b>
371	<b>11.8.3</b>	<b>Management of FIA_USB.1 .....</b>	<b>78</b>
372	<b>11.8.4</b>	<b>Audit of FIA_USB.1 .....</b>	<b>78</b>
373	<b>11.8.5</b>	<b>FIA_USB.1 User-subject binding.....</b>	<b>78</b>
374	<b>12</b>	<b>Class FMT: Security management .....</b>	<b>80</b>
375	<b>12.1</b>	<b>Class description.....</b>	<b>80</b>
376	<b>12.2</b>	<b>Limited capabilities and availability (FMT_LIM) .....</b>	<b>80</b>
377	<b>12.2.1</b>	<b>Family behaviour.....</b>	<b>80</b>
378	<b>12.2.2</b>	<b>Components leveling and description .....</b>	<b>81</b>
379	<b>12.2.3</b>	<b>Management of FMT_LIM.1, FMT_LIM.2.....</b>	<b>81</b>
380	<b>12.2.4</b>	<b>Audit of FMT_LIM.1.....</b>	<b>81</b>
381	<b>12.2.5</b>	<b>FMT_LIM.1 Limited capabilities.....</b>	<b>81</b>
382	<b>12.2.6</b>	<b>FMT_LIM.2 Limited availability .....</b>	<b>81</b>
383	<b>12.3</b>	<b>Management of functions in TSF (FMT_MOF) .....</b>	<b>82</b>
384	<b>12.3.1</b>	<b>Family behaviour.....</b>	<b>82</b>

385	<b>12.3.2 Components leveling and description.....</b>	<b>82</b>
386	<b>12.3.3 Management of FMT_MOF.1 .....</b>	<b>82</b>
387	<b>12.3.4 Audit of FMT_MOF.1.....</b>	<b>82</b>
388	<b>12.3.5 FMT_MOF.1 Management of security functions behaviour .....</b>	<b>82</b>
389	<b>12.4 Management of security attributes (FMT_MSA) .....</b>	<b>82</b>
390	<b>12.4.1 Family behaviour .....</b>	<b>82</b>
391	<b>12.4.2 Components leveling and description.....</b>	<b>83</b>
392	<b>12.4.3 Management of FMT_MSA.1.....</b>	<b>83</b>
393	<b>12.4.4 Management of FMT_MSA.2.....</b>	<b>83</b>
394	<b>12.4.5 Management of FMT_MSA.3.....</b>	<b>83</b>
395	<b>12.4.6 Management of FMT_MSA.4.....</b>	<b>83</b>
396	<b>12.4.7 Audit of FMT_MSA.1 .....</b>	<b>83</b>
397	<b>12.4.8 Audit of FMT_MSA.2 .....</b>	<b>84</b>
398	<b>12.4.9 Audit of FMT_MSA.3 .....</b>	<b>84</b>
399	<b>12.4.10 Audit of FMT_MSA.4.....</b>	<b>84</b>
400	<b>12.4.11 FMT_MSA.1 Management of security attributes .....</b>	<b>84</b>
401	<b>12.4.12 FMT_MSA.2 Secure security attributes.....</b>	<b>84</b>
402	<b>12.4.13 FMT_MSA.3 Static attribute initialization.....</b>	<b>85</b>
403	<b>12.4.14 FMT_MSA.4 Security attribute value inheritance.....</b>	<b>85</b>
404	<b>12.5 Management of TSF data (FMT_MTD).....</b>	<b>85</b>
405	<b>12.5.1 Family behaviour .....</b>	<b>85</b>
406	<b>12.5.2 Components leveling and description.....</b>	<b>85</b>
407	<b>12.5.3 Management of FMT_MTD.1 .....</b>	<b>86</b>
408	<b>12.5.4 Management of FMT_MTD.2 .....</b>	<b>86</b>
409	<b>12.5.5 Management of FMT_MTD.3 .....</b>	<b>86</b>
410	<b>12.5.6 Audit of FMT_MTD.1 .....</b>	<b>86</b>
411	<b>12.5.7 Audit of FMT_MTD.2 .....</b>	<b>86</b>
412	<b>12.5.8 Audit of FMT_MTD.3 .....</b>	<b>86</b>
413	<b>12.5.9 FMT_MTD.1 Management of TSF data.....</b>	<b>86</b>
414	<b>12.5.10 FMT_MTD.2 Management of limits on TSF data.....</b>	<b>86</b>
415	<b>12.5.11 FMT_MTD.3 Secure TSF data .....</b>	<b>87</b>
416	<b>12.6 Revocation (FMT_REV) .....</b>	<b>87</b>
417	<b>12.6.1 Family behaviour .....</b>	<b>87</b>
418	<b>12.6.2 Components leveling and description.....</b>	<b>87</b>
419	<b>12.6.3 Management of FMT_REV.1 .....</b>	<b>87</b>
420	<b>12.6.4 Audit of FMT_REV.1.....</b>	<b>87</b>
421	<b>12.6.5 FMT_REV.1 Revocation .....</b>	<b>88</b>
422	<b>12.7 Security attribute expiration (FMT_SAE).....</b>	<b>88</b>

423	<b>12.7.1 Family behaviour.....</b>	<b>88</b>
424	<b>12.7.2 Components leveling and description.....</b>	<b>88</b>
425	<b>12.7.3 Management of FMT_SAE.1 .....</b>	<b>88</b>
426	<b>12.7.4 Audit of FMT_SAE.1.....</b>	<b>88</b>
427	<b>12.7.5 FMT_SAE.1 Time-limited authorization .....</b>	<b>88</b>
428	<b>12.8 Specification of Management Functions (FMT_SMF) .....</b>	<b>89</b>
429	<b>12.8.1 Family behaviour.....</b>	<b>89</b>
430	<b>12.8.2 Components leveling and description.....</b>	<b>89</b>
431	<b>12.8.3 Management of FMT_SMF.1.....</b>	<b>89</b>
432	<b>12.8.4 Audit of FMT_SMF.1.....</b>	<b>89</b>
433	<b>12.8.5 FMT_SMF.1 Specification of Management Functions .....</b>	<b>89</b>
434	<b>12.9 Security management roles (FMT_SMR) .....</b>	<b>90</b>
435	<b>12.9.1 Family behaviour.....</b>	<b>90</b>
436	<b>12.9.2 Components leveling and description.....</b>	<b>90</b>
437	<b>12.9.3 Management of FMT_SMR.1 .....</b>	<b>90</b>
438	<b>12.9.4 Management of FMT_SMR.2 .....</b>	<b>90</b>
439	<b>12.9.5 Management of FMT_SMR.3 .....</b>	<b>90</b>
440	<b>12.9.6 Audit of FMT_SMR.1 .....</b>	<b>90</b>
441	<b>12.9.7 Audit of FMT_SMR.2 .....</b>	<b>90</b>
442	<b>12.9.8 Audit of FMT_SMR.3 .....</b>	<b>91</b>
443	<b>12.9.9 FMT_SMR.1 Security roles .....</b>	<b>91</b>
444	<b>12.9.10 FMT_SMR.2 Restrictions on security roles .....</b>	<b>91</b>
445	<b>12.9.11 FMT_SMR.3 Assuming roles .....</b>	<b>91</b>
446	<b>13 Class FPR: Privacy.....</b>	<b>92</b>
447	<b>13.1 Class description.....</b>	<b>92</b>
448	<b>13.2 Anonymity (FPR_ANO) .....</b>	<b>92</b>
449	<b>13.2.1 Family behaviour.....</b>	<b>92</b>
450	<b>13.2.2 Components leveling and description.....</b>	<b>93</b>
451	<b>13.2.3 Management of FPR_ANO.1, FPR_ANO.2 .....</b>	<b>93</b>
452	<b>13.2.4 Audit of FPR_ANO.1, FPR_ANO.2 .....</b>	<b>93</b>
453	<b>13.2.5 FPR_ANO.1 Anonymity .....</b>	<b>93</b>
454	<b>13.2.6 FPR_ANO.2 Anonymity without soliciting information.....</b>	<b>93</b>
455	<b>13.3 Pseudonymity (FPR_PSE) .....</b>	<b>94</b>
456	<b>13.3.1 Family behaviour.....</b>	<b>94</b>
457	<b>13.3.2 Components leveling and description.....</b>	<b>94</b>
458	<b>13.3.3 Management of FPR_PSE.1, FPR_PSE.2, FPR_PSE.3 .....</b>	<b>94</b>
459	<b>13.3.4 Audit of FPR_PSE.1, FPR_PSE.2, FPR_PSE.3 .....</b>	<b>94</b>
460	<b>13.3.5 FPR_PSE.1 Pseudonymity .....</b>	<b>94</b>

461	13.3.6 FPR_PSE.2 Reversible pseudonymity.....	95
462	13.3.7 FPR_PSE.3 Alias pseudonymity.....	95
463	13.4 Distribution of trust (FPR_TRD).....	96
464	13.4.1 Family behaviour .....	96
465	13.4.2 Components leveling and description.....	96
466	13.4.3 Management of FPR_TRD.1.....	96
467	13.4.4 Management of FPR_TRD.2.....	96
468	13.4.5 Management of FPR_TRD.3.....	96
469	13.4.6 Audit of FPR_TRD.1, FPR_TRD.2, FPR_TRD.3 .....	97
470	13.4.7 FPR_TRD.1 Administrative domains.....	97
471	13.4.8 FPR_TRD.2 Allocation of information assets .....	97
472	13.4.9 FPR_TRD.3 Allocation of processing activities.....	97
473	13.5 Unlinkability (FPR_UNL) .....	98
474	13.5.1 Family behaviour .....	98
475	13.5.2 Components leveling and description.....	98
476	13.5.3 Management of FPR_UNL.1, FPR_UNL.2, FPR_UNL.3 .....	98
477	13.5.4 Audit of FPR_UNL.1, FPR_UNL.2, FPR_UNL.3.....	99
478	13.5.5 FPR_UNL.1 Unlinkability of operations.....	99
479	13.6 Unobservability (FPR_UNO) .....	99
480	13.6.1 Family behaviour .....	99
481	13.6.2 Components leveling and description.....	100
482	13.6.3 Management of FPR_UNO.1, FPR_UNO.2 .....	100
483	13.6.4 Management of FPR_UNO.3 .....	100
484	13.6.5 Management of FPR_UNO.4 .....	100
485	13.6.6 Audit of FPR_UNO.1, FPR_UNO.2.....	100
486	13.6.7 Audit of FPR_UNO.3.....	100
487	13.6.8 Audit of FPR_UNO.4.....	100
488	13.6.9 FPR_UNO.1 Unobservability .....	101
489	13.6.10 FPR_UNO.2 Allocation of information impacting unobservability .....	101
490	13.6.11 FPR_UNO.3 Unobservability without soliciting information.....	101
491	13.6.12 FPR_UNO.4 Authorized user observability.....	101
492	14 Class FPT: Protection of the TSF.....	102
493	14.1 Class description .....	102
494	14.2 TOE emanation (FPT_EMS) .....	103
495	14.2.1 Family behaviour .....	103
496	14.2.2 Components leveling and description.....	104
497	14.2.3 Management of FPT_EMS.1 .....	104
498	14.2.4 Audit of FPT_EMS.1 .....	104



499	<b>14.2.5 FPT_EMS.1 Emanation of TSF and User data .....</b>	<b>104</b>
500	<b>14.3 Fail secure (FPT_FLS).....</b>	<b>104</b>
501	<b>14.3.1 Family behaviour.....</b>	<b>104</b>
502	<b>14.3.2 Components leveling and description .....</b>	<b>104</b>
503	<b>14.3.3 Management of FPT_FLS.1.....</b>	<b>105</b>
504	<b>14.3.4 Audit of FPT_FLS.1 .....</b>	<b>105</b>
505	<b>14.3.5 FPT_FLS.1 Failure with preservation of secure state.....</b>	<b>105</b>
506	<b>14.4 TSF initialization (FPT_INI) .....</b>	<b>105</b>
507	<b>14.4.1 Family behaviour.....</b>	<b>105</b>
508	<b>14.4.2 Components leveling and description .....</b>	<b>105</b>
509	<b>14.4.3 Management of FPT_INI.1.....</b>	<b>105</b>
510	<b>14.4.4 Audit of FPT_INI.1 .....</b>	<b>105</b>
511	<b>14.4.5 FPT_INI.1 TSF initialization .....</b>	<b>106</b>
512	<b>14.5 Availability of exported TSF data (FPT_ITA).....</b>	<b>106</b>
513	<b>14.5.1 Family behaviour.....</b>	<b>106</b>
514	<b>14.5.2 Components leveling and description .....</b>	<b>106</b>
515	<b>14.5.3 Management of FPT_ITA.1.....</b>	<b>106</b>
516	<b>14.5.4 Audit of FPT_ITA.1 .....</b>	<b>107</b>
517	<b>14.5.5 FPT_ITA.1 Inter-TSF availability within a defined availability metric .....</b>	<b>107</b>
518	<b>14.6 Confidentiality of exported TSF data (FPT_ITC).....</b>	<b>107</b>
519	<b>14.6.1 Family behaviour.....</b>	<b>107</b>
520	<b>14.6.2 Components leveling and description .....</b>	<b>107</b>
521	<b>14.6.3 Management of FPT_ITC.1 .....</b>	<b>107</b>
522	<b>14.6.4 Audit of FPT_ITC.1 .....</b>	<b>107</b>
523	<b>14.6.5 FPT_ITC.1 Inter-TSF confidentiality during transmission .....</b>	<b>107</b>
524	<b>14.7 Integrity of exported TSF data (FPT_ITI) .....</b>	<b>108</b>
525	<b>14.7.1 Family behaviour.....</b>	<b>108</b>
526	<b>14.7.2 Components leveling and description .....</b>	<b>108</b>
527	<b>14.7.3 Management of FPT_ITI.1 .....</b>	<b>108</b>
528	<b>14.7.4 Management of FPT_ITI.2 .....</b>	<b>108</b>
529	<b>14.7.5 Audit of FPT_ITI.1 .....</b>	<b>108</b>
530	<b>14.7.6 Audit of FPT_ITI.2 .....</b>	<b>108</b>
531	<b>14.7.7 FPT_ITI.1 Inter-TSF detection of modification .....</b>	<b>109</b>
532	<b>14.7.8 FPT_ITI.2 Inter-TSF detection and correction of modification .....</b>	<b>109</b>
533	<b>14.8 Internal TOE TSF data transfer (FPT_ITT).....</b>	<b>109</b>
534	<b>14.8.1 Family behaviour.....</b>	<b>109</b>
535	<b>14.8.2 Components leveling and description .....</b>	<b>109</b>
536	<b>14.8.3 Management of FPT_ITT.1.....</b>	<b>110</b>

537	<b>14.8.4 Management of FPT_ITT.2 .....</b>	<b>110</b>
538	<b>14.8.5 Management of FPT_ITT.3 .....</b>	<b>110</b>
539	<b>14.8.6 Audit of FPT_ITT.1, FPT_ITT.2 .....</b>	<b>110</b>
540	<b>14.8.7 Audit of FPT_ITT.3.....</b>	<b>110</b>
541	<b>14.8.8 FPT_ITT.1 Basic internal TSF data transfer protection .....</b>	<b>110</b>
542	<b>14.8.9 FPT_ITT.2 TSF data transfer separation .....</b>	<b>111</b>
543	<b>14.8.10 FPT_ITT.3 TSF data integrity monitoring .....</b>	<b>111</b>
544	<b>14.9 TSF physical protection (FPT_PHP).....</b>	<b>111</b>
545	<b>14.9.1 Family behaviour .....</b>	<b>111</b>
546	<b>14.9.2 Components leveling and description.....</b>	<b>112</b>
547	<b>14.9.3 Management of FPT_PHP.1 .....</b>	<b>112</b>
548	<b>14.9.4 Management of FPT_PHP.2 .....</b>	<b>112</b>
549	<b>14.9.5 Management of FPT_PHP.3 .....</b>	<b>112</b>
550	<b>14.9.6 Audit of FPT_PHP.1 .....</b>	<b>112</b>
551	<b>14.9.7 Audit of FPT_PHP.2 .....</b>	<b>112</b>
552	<b>14.9.8 Audit of FPT_PHP.3 .....</b>	<b>112</b>
553	<b>14.9.9 FPT_PHP.1 Passive detection of physical attack.....</b>	<b>113</b>
554	<b>14.9.10 FPT_PHP.2 Notification of physical attack.....</b>	<b>113</b>
555	<b>14.9.11 FPT_PHP.3 Resistance to physical attack .....</b>	<b>113</b>
556	<b>14.10 Trusted recovery (FPT_RCV).....</b>	<b>113</b>
557	<b>14.10.1 Family behaviour .....</b>	<b>113</b>
558	<b>14.10.2 Components leveling and description.....</b>	<b>114</b>
559	<b>14.10.3 Management of FPT_RCV.1 .....</b>	<b>114</b>
560	<b>14.10.4 Management of FPT_RCV.2, FPT_RCV.3 .....</b>	<b>114</b>
561	<b>14.10.5 Management of FPT_RCV.4 .....</b>	<b>114</b>
562	<b>14.10.6 Audit of FPT_RCV.1, FPT_RCV.2, FPT_RCV.3 .....</b>	<b>114</b>
563	<b>14.10.7 Audit of FPT_RCV.4.....</b>	<b>114</b>
564	<b>14.10.8 FPT_RCV.1 Manual recovery.....</b>	<b>115</b>
565	<b>14.10.9 FPT_RCV.3 Automated recovery without undue loss .....</b>	<b>115</b>
566	<b>14.10.10 FPT_RCV.4 Function recovery.....</b>	<b>116</b>
567	<b>14.11 Replay detection (FPT_RPL).....</b>	<b>116</b>
568	<b>14.11.1 Family behaviour .....</b>	<b>116</b>
569	<b>14.11.2 Components leveling and description.....</b>	<b>116</b>
570	<b>14.11.3 Management of FPT_RPL.1.....</b>	<b>116</b>
571	<b>14.11.4 Audit of FPT_RPL.1.....</b>	<b>116</b>
572	<b>14.11.5 FPT_RPL.1 Replay detection.....</b>	<b>116</b>
573	<b>14.12 State synchrony protocol (FPT_SSP).....</b>	<b>117</b>
574	<b>14.12.1 Family behaviour .....</b>	<b>117</b>

575	<b>14.12.2</b>	<b>Components leveling and description .....</b>	<b>117</b>
576	<b>14.12.3</b>	<b>Management of FPT_SSP.1, FPT_SSP.2 .....</b>	<b>117</b>
577	<b>14.12.4</b>	<b>Audit of FPT_SSP.1, FPT_SSP.2.....</b>	<b>117</b>
578	<b>14.12.5</b>	<b>FPT_SSP.1 Simple trusted acknowledgement.....</b>	<b>117</b>
579	<b>14.12.6</b>	<b>FPT_SSP.2 Mutual trusted acknowledgement.....</b>	<b>118</b>
580	<b>14.13</b>	<b>Time stamps (FPT_STM) .....</b>	<b>118</b>
581	<b>14.13.1</b>	<b>Family behaviour.....</b>	<b>118</b>
582	<b>14.13.2</b>	<b>Components leveling and description .....</b>	<b>118</b>
583	<b>14.13.3</b>	<b>Management of FPT_STM.1 .....</b>	<b>118</b>
584	<b>14.13.4</b>	<b>Management of FPT_STM.2 .....</b>	<b>118</b>
585	<b>14.13.5</b>	<b>Audit of FPT_STM.1 .....</b>	<b>118</b>
586	<b>14.13.6</b>	<b>Audit of FPT_STM.2 .....</b>	<b>119</b>
587	<b>14.13.7</b>	<b>FPT_STM.1 Reliable time stamps.....</b>	<b>119</b>
588	<b>14.13.8</b>	<b>FPT_STM.2 Time source.....</b>	<b>119</b>
589	<b>14.14</b>	<b>Inter-TSF TSF data consistency (FPT_TDC) .....</b>	<b>119</b>
590	<b>14.14.1</b>	<b>Family behaviour.....</b>	<b>119</b>
591	<b>14.14.2</b>	<b>Components leveling and description .....</b>	<b>119</b>
592	<b>14.14.3</b>	<b>Management of FPT_TDC.1 .....</b>	<b>119</b>
593	<b>14.14.4</b>	<b>Audit of FPT_TDC.1.....</b>	<b>120</b>
594	<b>14.14.5</b>	<b>FPT_TDC.1 Inter-TSF basic TSF data consistency .....</b>	<b>120</b>
595	<b>14.15</b>	<b>Testing of external entities (FPT_TEE) .....</b>	<b>120</b>
596	<b>14.15.1</b>	<b>Family behaviour.....</b>	<b>120</b>
597	<b>14.15.2</b>	<b>Components leveling and description .....</b>	<b>120</b>
598	<b>14.15.3</b>	<b>Management of FPT_TEE.1 .....</b>	<b>120</b>
599	<b>14.15.4</b>	<b>Audit of FPT_TEE.1 .....</b>	<b>121</b>
600	<b>14.15.5</b>	<b>FPT_TEE.1 Testing of external entities.....</b>	<b>121</b>
601	<b>14.16</b>	<b>Internal TOE TSF data replication consistency (FPT_TRC) .....</b>	<b>121</b>
602	<b>14.16.1</b>	<b>Family behaviour.....</b>	<b>121</b>
603	<b>14.16.2</b>	<b>Components leveling and description .....</b>	<b>121</b>
604	<b>14.16.3</b>	<b>Management of FPT_TRC.1.....</b>	<b>121</b>
605	<b>14.16.4</b>	<b>Audit of FPT_TRC.1.....</b>	<b>121</b>
606	<b>14.16.5</b>	<b>FPT_TRC.1 Internal TSF consistency .....</b>	<b>122</b>
607	<b>14.17</b>	<b>TSF self-test (FPT_TST) .....</b>	<b>122</b>
608	<b>14.17.1</b>	<b>Family behaviour.....</b>	<b>122</b>
609	<b>14.17.2</b>	<b>Components leveling and description .....</b>	<b>122</b>
610	<b>14.17.3</b>	<b>Management of FPT_TST.1 .....</b>	<b>122</b>
611	<b>14.17.4</b>	<b>Audit of FPT_TST.1 .....</b>	<b>123</b>
612	<b>14.17.5</b>	<b>FPT_TST.1 TSF self-testing .....</b>	<b>123</b>

613	<b>15</b>	<b>Class FRU: Resource utilization.....</b>	<b>124</b>
614	<b>15.1</b>	<b>Class description .....</b>	<b>124</b>
615	<b>15.2</b>	<b>Fault tolerance (FRU_FLT) .....</b>	<b>124</b>
616	<b>15.2.1</b>	<b>Family behaviour .....</b>	<b>124</b>
617	<b>15.2.2</b>	<b>Components leveling and description.....</b>	<b>124</b>
618	<b>15.2.3</b>	<b>Management of FRU_FLT.1, FRU_FLT.2.....</b>	<b>125</b>
619	<b>15.2.4</b>	<b>Audit of FRU_FLT.1 .....</b>	<b>125</b>
620	<b>15.2.5</b>	<b>Audit of FRU_FLT.2 .....</b>	<b>125</b>
621	<b>15.2.6</b>	<b>FRU_FLT.1 Degraded fault tolerance .....</b>	<b>125</b>
622	<b>15.2.7</b>	<b>FRU_FLT.2 Limited fault tolerance .....</b>	<b>125</b>
623	<b>15.3</b>	<b>Priority of service (FRU_PRS) .....</b>	<b>125</b>
624	<b>15.3.1</b>	<b>Family behaviour .....</b>	<b>125</b>
625	<b>15.3.2</b>	<b>Components leveling and description.....</b>	<b>125</b>
626	<b>15.3.3</b>	<b>Management of FRU_PRS.1, FRU_PRS.2 .....</b>	<b>126</b>
627	<b>15.3.4</b>	<b>Audit of FRU_PRS.1, FRU_PRS.2 .....</b>	<b>126</b>
628	<b>15.3.5</b>	<b>FRU_PRS.1 Limited priority of service.....</b>	<b>126</b>
629	<b>15.3.6</b>	<b>FRU_PRS.2 Full priority of service .....</b>	<b>126</b>
630	<b>15.4</b>	<b>Resource allocation (FRU_RSA).....</b>	<b>126</b>
631	<b>15.4.1</b>	<b>Family behaviour .....</b>	<b>126</b>
632	<b>15.4.2</b>	<b>Components leveling and description.....</b>	<b>127</b>
633	<b>15.4.3</b>	<b>Management of FRU_RSA.1 .....</b>	<b>127</b>
634	<b>15.4.4</b>	<b>Management of FRU_RSA.2 .....</b>	<b>127</b>
635	<b>15.4.5</b>	<b>Audit of FRU_RSA.1, FRU_RSA.2 .....</b>	<b>127</b>
636	<b>15.4.6</b>	<b>FRU_RSA.1 Maximum quotas.....</b>	<b>127</b>
637	<b>15.4.7</b>	<b>FRU_RSA.2 Minimum and maximum quotas.....</b>	<b>127</b>
638	<b>16</b>	<b>Class FTA: TOE access.....</b>	<b>129</b>
639	<b>16.1</b>	<b>Class description .....</b>	<b>129</b>
640	<b>16.2</b>	<b>Limitation on scope of selectable attributes (FTA_LSA).....</b>	<b>129</b>
641	<b>16.2.1</b>	<b>Family behaviour .....</b>	<b>129</b>
642	<b>16.2.2</b>	<b>Components leveling and description.....</b>	<b>129</b>
643	<b>16.2.3</b>	<b>Management of FTA_LSA.1.....</b>	<b>130</b>
644	<b>16.2.4</b>	<b>Audit of FTA_LSA.1 .....</b>	<b>130</b>
645	<b>16.2.5</b>	<b>FTA_LSA.1 Limitation on scope of selectable attributes.....</b>	<b>130</b>
646	<b>16.3</b>	<b>Limitation on multiple concurrent sessions (FTA_MCS) .....</b>	<b>130</b>
647	<b>16.3.1</b>	<b>Family behaviour .....</b>	<b>130</b>
648	<b>16.3.2</b>	<b>Components leveling and description.....</b>	<b>130</b>
649	<b>16.3.3</b>	<b>Management of FTA_MCS.1 .....</b>	<b>130</b>
650	<b>16.3.4</b>	<b>Management of FTA_MCS.2 .....</b>	<b>131</b>

651	<b>16.3.5 Audit of FTA_MCS.1, FTA_MCS.2 .....</b>	<b>131</b>
652	<b>16.3.6 FTA_MCS.1 Basic limitation on multiple concurrent sessions .....</b>	<b>131</b>
653	<b>16.3.7 FTA_MCS.2 Per user attribute limitation on multiple concurrent sessions ....</b>	<b>131</b>
654	<b>16.4 Session locking and termination (FTA_SSL).....</b>	<b>131</b>
655	<b>16.4.1 Family behaviour.....</b>	<b>131</b>
656	<b>16.4.2 Components leveling and description .....</b>	<b>132</b>
657	<b>16.4.3 Management of FTA_SSL.1.....</b>	<b>132</b>
658	<b>16.4.4 Management of FTA_SSL.2.....</b>	<b>132</b>
659	<b>16.4.5 Management of FTA_SSL.3.....</b>	<b>132</b>
660	<b>16.4.6 Management of FTA_SSL.4.....</b>	<b>132</b>
661	<b>16.4.7 Audit of FTA_SSL.1, FTA_SSL.2 .....</b>	<b>133</b>
662	<b>16.4.8 Audit of FTA_SSL.3 .....</b>	<b>133</b>
663	<b>16.4.9 Audit of FTA_SSL.4.....</b>	<b>133</b>
664	<b>16.4.10 FTA_SSL.1 TSF-initiated session locking .....</b>	<b>133</b>
665	<b>16.4.11 FTA_SSL.2 User-initiated locking.....</b>	<b>133</b>
666	<b>16.4.12 FTA_SSL.3 TSF-initiated termination .....</b>	<b>134</b>
667	<b>16.4.13 FTA_SSL.4 User-initiated termination .....</b>	<b>134</b>
668	<b>16.5 TOE access banners (FTA_TAB).....</b>	<b>134</b>
669	<b>16.5.1 Family behaviour.....</b>	<b>134</b>
670	<b>16.5.2 Components leveling and description .....</b>	<b>134</b>
671	<b>16.5.3 Management of FTA_TAB.1 .....</b>	<b>134</b>
672	<b>16.5.4 Audit of FTA_TAB.1.....</b>	<b>134</b>
673	<b>16.5.5 FTA_TAB.1 Default TOE access banners.....</b>	<b>135</b>
674	<b>16.6 TOE access history (FTA_TAH).....</b>	<b>135</b>
675	<b>16.6.1 Family behaviour.....</b>	<b>135</b>
676	<b>16.6.2 Components leveling and description .....</b>	<b>135</b>
677	<b>16.6.3 Management of FTA_TAH.1 .....</b>	<b>135</b>
678	<b>16.6.4 Audit of FTA_TAH.1 .....</b>	<b>135</b>
679	<b>16.6.5 FTA_TAH.1 TOE access history.....</b>	<b>135</b>
680	<b>16.7 TOE session establishment (FTA_TSE) .....</b>	<b>136</b>
681	<b>16.7.1 Family behaviour.....</b>	<b>136</b>
682	<b>16.7.2 Components leveling and description .....</b>	<b>136</b>
683	<b>16.7.3 Management of FTA_TSE.1 .....</b>	<b>136</b>
684	<b>16.7.4 Audit of FTA_TSE.1 .....</b>	<b>136</b>
685	<b>16.7.5 FTA_TSE.1 TOE session establishment .....</b>	<b>136</b>
686	<b>17 Class FTP: Trusted path/channels.....</b>	<b>137</b>
687	<b>17.1 Class description.....</b>	<b>137</b>
688	<b>17.2 Inter-TSF trusted channel (FTP_ITC) .....</b>	<b>137</b>

689	<b>17.2.1 Family behaviour .....</b>	<b>138</b>
690	<b>17.2.2 Components leveling and description.....</b>	<b>138</b>
691	<b>17.2.3 Management of FTP_ITC.1.....</b>	<b>138</b>
692	<b>17.2.4 Audit of FTP_ITC.1 .....</b>	<b>138</b>
693	<b>17.2.5 FTP_ITC.1 Inter-TSF trusted channel.....</b>	<b>138</b>
694	<b>17.3 Secure channel (FTP_PRO) .....</b>	<b>139</b>
695	<b>17.3.1 Components leveling and description.....</b>	<b>139</b>
696	<b>17.3.2 Management of FTP_PRO.1 .....</b>	<b>139</b>
697	<b>17.3.3 Audit of FTP_PRO.1 .....</b>	<b>139</b>
698	<b>17.3.4 FTP_PRO.1 Trusted channel protocol.....</b>	<b>140</b>
699	<b>17.3.5 FTP_PRO.2 Trusted channel key establishment.....</b>	<b>140</b>
700	<b>17.3.6 FTP_PRO.3 Trusted channel data protection.....</b>	<b>141</b>
701	<b>17.4 Trusted path (FTP_TRP).....</b>	<b>142</b>
702	<b>17.4.1 Family behaviour .....</b>	<b>142</b>
703	<b>17.4.2 Components leveling and description.....</b>	<b>142</b>
704	<b>17.4.3 Management of FTP_TRP.1 .....</b>	<b>142</b>
705	<b>17.4.4 Audit of FTP_TRP.1 .....</b>	<b>142</b>
706	<b>17.4.5 FTP_TRP.1 Trusted path.....</b>	<b>142</b>
707	<b>Annex A (normative) Security functional requirements structure of the application</b>	
708	<b>notes .....</b>	<b>144</b>
709	<b>Annex B (informative) Dependency tables for security functional components.....</b>	<b>147</b>
710	<b>Annex C (normative) Class FAU: Security audit - application notes .....</b>	<b>158</b>
711	<b>Annex D (normative) Class FCO: Communication- application notes .....</b>	<b>170</b>
712	<b>Annex E (normative) Class FCS: Cryptographic support- application notes .....</b>	<b>176</b>
713	<b>Annex F (normative) Class FDP: User data protection- application notes.....</b>	<b>185</b>
714	<b>Annex G (normative) Class FIA: Identification and authentication- application notes</b>	
715	<b>.....</b>	<b>209</b>
716	<b>Annex H (normative) Class FMT: Security management- application notes.....</b>	<b>218</b>
717	<b>Annex I (normative) Class FPR: Privacy- application notes.....</b>	<b>227</b>
718	<b>Annex J (normative) Class FPT: Protection of the TSF- application notes .....</b>	<b>239</b>
719	<b>Annex K (normative) Class FRU: Resource utilization- application notes .....</b>	<b>255</b>
720	<b>Annex L (normative) Class FTA: TOE access- application notes .....</b>	<b>260</b>
721	<b>Annex M (normative) Class FTP: Trusted path/channels- application notes .....</b>	<b>266</b>
722		

## 723 Foreword

724 ISO (the International Organization for Standardization) and IEC (the International  
725 Electrotechnical Commission) form the specialized system for worldwide standardization.  
726 National bodies that are members of ISO or IEC participate in the development of International  
727 Standards through technical committees established by the respective organization to deal with  
728 particular fields of technical activity. ISO and IEC technical committees collaborate in fields of  
729 mutual interest. Other international organizations, governmental and non-governmental, in  
730 liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and  
731 IEC have established a joint technical committee, ISO/IEC JTC 1.

732 The procedures used to develop this document and those intended for its further maintenance  
733 are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria  
734 needed for the different types of document should be noted. This document was drafted in  
735 accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see [www.iso](http://www.iso.org/directives)  
736 [.org/directives](http://www.iso.org/directives)).

737 Attention is drawn to the possibility that some of the elements of this document may be the  
738 subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such  
739 patent rights. Details of any patent rights identified during the development of the document will  
740 be in the Introduction and/or on the ISO list of patent declarations received (see [www.iso](http://www.iso.org/patents)  
741 [.org/patents](http://www.iso.org/patents)).

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

744 For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and  
745 expressions related to conformity assessment, as well as information about ISO's adherence to  
746 the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see  
747 [www.iso.org/iso/foreword.html](http://www.iso.org/iso/foreword.html).

748 This document was prepared by Technical Committee ISO/IEC JTC 1, Information technology,  
749 Subcommittee SC 27, IT Security techniques.

750 A list of all parts in the ISO/IEC 15408 series can be found on the ISO website.

751 Any feedback or questions on this document should be directed to the user's national standards  
752 body. A complete listing of these bodies can be found at [www.iso.org/members.html](http://www.iso.org/members.html).

753 This **fourth** edition cancels and replaces the **third** edition (ISO 15408-2:2008), which has been  
754 technically revised.

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

- 756 — The document has been revised to comply with ISO/IEC Directives
- 757 — Technical changes have been introduced:
  - 758 ○ New security functional components have been introduced

759

## Introduction

Security functional components, as defined in this document, are the basis for the security functional requirements expressed in a Protection Profile (PP) or a Security Target (ST). These requirements describe the desired security behaviour expected of a Target of Evaluation (TOE) and are intended to meet the security objectives as stated in a PP or an ST. These requirements describe security properties that users **can** detect by direct interaction (i.e. inputs, outputs) with the IT or by the IT response to stimulus.

Security functional components express security requirements intended to counter threats in the assumed operating environment of the TOE and/or cover any identified organizational security policies.

The audience for this document includes consumers, developers, and evaluators of secure IT products. ISO/IEC 15408-1:20XX, Clause 5.3 provides additional information on the target audience of the ISO/IEC 15408 series, and on the use of the ISO/IEC 15408 series by the groups that comprise the target audience. These groups **may should** use this document as follows:

- a) Consumers, who use this document when selecting components to express functional requirements which satisfy the security objectives expressed in a PP or ST. ISO/IEC 15408-1:20XX, Clause 6 provides more detailed information on the relationship between security objectives and security requirements.
- b) Developers, who respond to actual or perceived consumer security requirements in constructing a TOE, **may** find a standardized method to understand those requirements in this document. They **can** also use the contents of this document as a basis for further defining the TOE security functionality and mechanisms that comply with those requirements.
- c) Evaluators, who use the functional requirements defined in this document in verifying that the TOE functional requirements expressed in the PP or ST satisfy the IT security objectives and that all dependencies are accounted for and shown to be satisfied. Evaluators also **should** use this document to assist in determining whether a given TOE satisfies stated requirements.



# IT Security techniques — Evaluation criteria for IT security — Part 2: Security functional components

## 1 Scope

This document defines the required structure and content of security functional components for the purpose of security evaluation. It includes a catalogue of functional components that will meet the common security functionality requirements of many IT products.

## 2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

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

~~ISO/IEC 15408-3, *IT Security techniques — Evaluation criteria for IT security — Part 3: Security assurance requirements*~~

### Editors' Note

ISO/IEC 15408-3 is not normative to this document and will be removed in the next draft.

## 3 Terms and Definitions

For the purposes of this document, the terms, definitions, and abbreviated terms given in ISO/IEC 15408-1:20XX apply.

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

- ISO Online browsing platform: available at <http://www.iso.org/obp>
- IEC Electropedia: available at <http://www.electropedia.org/>

## 4 Overview

The ISO/IEC 15408 series and the associated security functional requirements described in this document are not intended to be a definitive answer to all the problems of IT security. This document offers a set of well understood security functional components that **can** be used to specify trusted products reflecting the needs of the market. These security functional components are presented as the current state of the art in security requirements specification and evaluation.

This document does not include all possible security functional components but contains those that are known and agreed to be of value by this the contributors to this document.

Since the understanding and needs of consumers **may** change, the functional components in this document will need to be maintained. It is envisioned that some PP/ST authors **may** have security needs not (yet) covered by the functional requirement components in this document. In those cases, the PP/ST author **may** choose to consider using functional components and requirements that are not given in this document. The concepts of extensibility are explained in Annex D of ISO/IEC 15408-1:20XX.

## 4.1 Organization of this document

Clause 5 describes the paradigm, explaining how security functional requirements **can** be derived from the security functional components given in this document.

Clause 6 introduces the catalogue of functional components while clauses 7 through 17 describe the functional classes.

Annex A provides explanatory information for potential users of the functional components.

Annex B provides a complete cross reference table of the functional component dependencies.

Annex C through Annex M provide the explanatory information for the functional classes. This material must be seen as normative instructions on how to apply relevant operations and select appropriate audit or documentation information; the use of the auxiliary verb “**should**” means that the instruction is strongly preferred, but others **may** be justifiable. Where different options are given, the choice is left to the PP/ST author.

Those who author PPs or STs **should** refer to Clause 8 of ISO/IEC 15408-1:20XX for relevant structures, rules, and guidance, in addition:

- a) ISO/IEC 15408-1:20XX, Clause 3 defines the terms and definitions used in ISO/IEC 15408.
- b) ISO/IEC 15408-1:20XX, Annex A defines the structure for STs.
- c) ISO/IEC 15408-1:20XX, Annex B defines the structure for PPs and modular PPs.
- d) ISO/IEC 15408-1:20XX, Annex B defines the structure for packages.

## 5 Functional requirements paradigm

### Editors' note

The editors have revised this clause making corrections for consistency with the revisions in ISO/IEC 15408-1

This clause describes the paradigm used in the security functional components and the derivation of security functional requirements. The key concepts discussed are highlighted in bold/italics. This subclause is not intended to replace or supersede any of the terms found in ISO/IEC 15408-1:20XX, Clause 3.

This document is a catalogue of security functional components that **can** be used to identify security functional requirements that **may** be specified for a **Target of Evaluation (TOE)**.

### Editors' Note

Editors suggest that the difference between a security functional component and a security functional requirement **should** be explained.

Editors propose the following text:

“Security functional components provide a template for security functional requirements. Security functional components **may** contain the operations **selection** and **assignment** which are explained in ISO/IEC 15408-1. Security functional requirements form part of the TOE security specification.”

If no comments are received on this proposal, the editor's proposal will be accepted and presented in the next draft.

TOE evaluation is concerned primarily with ensuring that a defined set of **security functional requirements (SFRs)** is enforced over the TOE resources. The SFRs define the rules by which the TOE governs access to and use of its resources, and thus information and services controlled by the TOE.

The SFRs **may** define multiple **Security Function Policies (SFPs)** to represent the rules that the TOE must enforce. Each SFP **must** specify its **scope of control**, by defining the subjects, objects, resources or information, and operations to which it applies. All SFPs are implemented by the

TSF (see below), whose mechanisms enforce the rules defined in the SFRs and provide necessary capabilities.

Those portions of a TOE that **must** be relied on for the correct enforcement of the SFRs are collectively referred to as the **TOE Security Functionality (TSF)**. The TSF consists of all hardware, software, and firmware of a TOE that is either directly or indirectly relied upon for security enforcement.

The TOE **may** be a monolithic product containing hardware, firmware, and software.

Alternatively, a TOE **may** be a distributed product that consists internally of multiple separated parts. Each of these parts of the TOE provides a particular service for the TOE and is connected to the other parts of the TOE through an **internal communication channel**. This channel **can** be as small as a processor bus or **may** encompass a network internal to the TOE.

When the TOE consists of multiple parts, each part of the TOE **may** have its own part of the TSF which exchanges user and TSF data over internal communication channels with other parts of the TSF. This interaction is called **internal TOE transfer**. In this case, the separate parts of the TSF abstractly form the composite TSF, which enforces the SFRs.

TOE interfaces **may** be localized to the particular TOE, or they **may** allow interaction with other IT products over **external communication channels**. These external interactions with other IT products **may** take two forms:

- a) The SFRs of the other “trusted IT product” and the SFRs of the TOE have been administratively coordinated and the other trusted IT product is assumed to enforce its SFRs correctly (e. g. by being separately evaluated). Exchanges of information in this situation are called **inter-TSF transfers**, as they are between the TSFs of distinct trusted products.
- b) The other IT product **may** not be trusted, it **may** be called an “untrusted IT product”. Therefore, its SFRs are either unknown or their implementation is not viewed as trustworthy. TSF mediated exchanges of information in this situation are called **transfers outside of the TOE**, as there is no TSF (or its policy characteristics are unknown) on the other IT product.

The set of interfaces, whether interactive (man-machine interface) or programmatic (application programming interface), through which resources are accessed that are mediated by the TSF, or information is obtained from the TSF, is referred to as the **TSF Interface (TSFI)**. The TSFI defines the boundaries of the TOE functionality that provide for the enforcement of the SFRs.

Users are outside of the TOE. However, in order to request that services be performed by the TOE that are subject to rules defined in the SFRs, users interact with the TOE through the TSFIs. There are two types of users of interest to this document: **human users** and **external IT entities**. Human users **may** further be differentiated as **local human users**, meaning they interact directly with the TOE via TOE devices or **remote human users**, meaning they interact indirectly with the TOE through another IT product.

#### EXAMPLE 1

An example of a TOE device is a workstation.

A period of interaction between users and the TSF is referred to as a user **session**. Establishment of user sessions **can** be controlled based on a variety of considerations.

#### EXAMPLE 2

user authentication, time of day, method of accessing the TOE, and number of allowed concurrent sessions (per user or in total).

This document uses the term **authorized** to signify a user who possesses the rights and/or privileges necessary to perform an operation. The term **authorized user**, therefore, indicates

that it is allowable for a user to perform a specific operation or a set of operations as defined by the SFRs.

To express requirements that call for the separation of administrator duties, the relevant security functional components (from family FMT\_SMR) explicitly state that administrative **roles** are required. A role is a pre-defined set of rules establishing the allowed interactions between a user operating in that role and the TOE. A TOE **may** support the definition of any number of roles.

EXAMPLE 3

Roles related to the secure operation of a TOE **may** include "Audit Administrator" and "User Accounts Administrator".

TOEs contain **resources** that **may** be used for the processing and storing of information. The primary goal of the TSF is the complete and correct enforcement of the SFRs over the resources and information that the TOE controls.

TOE resources **can** be structured and utilized in many different ways. However, this document makes a specific distinction that allows for the specification of desired security properties. All entities that **can** be created from resources **can** be characterized in one of two ways. The entities **may** be active, meaning that they are the cause of actions that occur internal to the TOE and cause operations to be performed on information. Alternatively, the entities **may** be passive, meaning that they are either the container from which information originates or to which information is stored.

Active entities in the TOE that perform operations on objects are referred to as **subjects**. Several types of subjects **may** exist within a TOE:

- a) those acting on behalf of an authorized user;

EXAMPLE 4

UNIX processes

- b) those acting as a specific functional process that **may** in turn act on behalf of multiple users;

EXAMPLE 5

functions as might be found in client/server architectures

- c) those acting as part of the TOE itself.

EXAMPLE 6

processes not acting on behalf of a user

This document addresses the enforcement of the SFRs over types of subjects as those listed above.

Passive entities in the TOE that contain or receive information and upon which subjects perform operations are called **objects**. In the case where a subject (an active entity) is the target of an operation, a subject **may** also be acted on as an object.

EXAMPLE 7

An example of a subject is an inter-process communication

Objects **can** contain **information**. This concept is required to specify information flow control policies as addressed in the FDP class.

Users, subjects, information, objects, sessions, and resources controlled by rules in the SFRs **may** possess certain **attributes** that contain information that is used by the TOE for its correct operation. Some attributes, such as file names, **may** be intended to be informational or **may** be used to identify individual resources while others, such as access control information, **may** exist specifically for the enforcement of the SFRs. These latter attributes are generally referred to as

“**security attributes**”. The word attribute will be used as a shorthand in some places in this document for the term “security attribute”. However, no matter what the intended purpose of the attribute information, it **may** be necessary to have controls on attributes as dictated by the SFRs.

Data in a TOE is categorized as either user data or TSF data. Figure 1 depicts this relationship. **User Data** is information stored in TOE resources that **can** be operated upon by users in accordance with the SFRs and upon which the TSF places no special meaning. **TSF Data** is information used by the TSF in making decisions as required by the SFRs. TSF Data may be influenced by users if allowed by the SFRs.

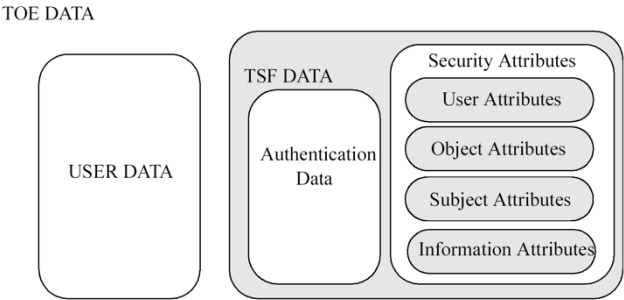
EXAMPLE 8

User data:  
The content of an electronic mail message is user data.

TSF data:  
Security attributes, authentication data, TSF internal status variables used by the rules defined in the SFRs or used for the protection of the TSF and access control list entries are examples of TSF data.

There are several SFPs that apply to data protection such as **access control SFPs** and **information flow control SFPs**. The mechanisms that implement access control SFPs base their policy decisions on attributes of the users, resources, subjects, objects, sessions, TSF status data and operations within the scope of control. These attributes are used in the set of rules that govern operations that subjects **may** perform on objects.

The mechanisms that implement information flow control SFPs base their policy decisions on the attributes of the subjects and information within the scope of control and the set of rules that govern the operations by subjects on information. The attributes of the information, which **may** be associated with the attributes of the container or **may** be derived from the data in the container, stay with the information as it is processed by the TSF.



**Figure 1 — Relationship between user data and TSF data**

Two specific types of TSF data addressed by this document **can** be, but are not necessarily, the same. These are **authentication data** and **secrets**.

Authentication data is used to verify the claimed identity of a user requesting services from a TOE. The most common form of authentication data is the password, which depends on being kept secret in order to be an effective security mechanism. However, not all forms of authentication data need to be kept secret. Biometric authentication devices do not rely on the fact that the data is kept secret, but rather that the data is something that only one user possesses and that cannot be forged.

EXAMPLE 9

Examples of biometric authentication devices include fingerprint readers and retinal scanners.

The term secrets, as used in this document, while applicable to authentication data, is intended to also be applicable to other types of data that must be kept secret in order to enforce a specific SFP.

EXAMPLE 10

a trusted channel mechanism that relies on cryptography to preserve the confidentiality of information being transmitted via the channel can only be as strong as the method used to keep the cryptographic keys secret from unauthorized disclosure

Therefore, some, but not all, authentication data needs to be kept secret and some, but not all, secrets are used as authentication data. Figure 2 shows this relationship between secrets and authentication data. In the Figure, the types of data typically encountered in the authentication data and the secrets subclauses are indicated.

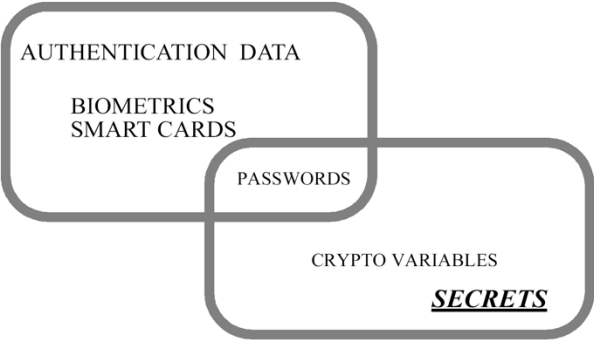


Figure 2 — Relationship between “authentication data” and “secrets”

6 Security functional components

**Editors' note**  
The editors have revised this clause making corrections for consistency with the revisions in ISO/IEC 15408-1.  
The Editors' have also attempted to correct inconsistencies noted in the use of the term security functional requirements where security functional components was meant.

6.1 Overview

This clause defines the content and presentation of the functional requirements of this document and provides guidance on the organization of the requirements for new, extended components that may be included in an ST, PP, PP-Module, or security functional package. The functional components and requirements are expressed in classes, families, and components.

6.1.1 Class structure

Figure 3 illustrates the functional class structure in diagrammatic form. Each functional class includes a class name, class introduction, and one or more functional families.

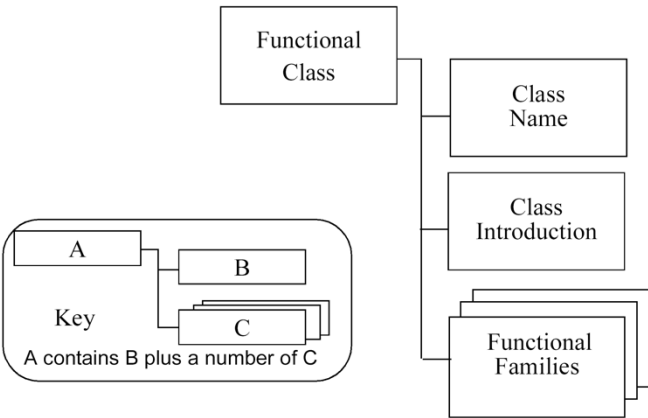


Figure 3 — Functional class structure

6.1.1.1 Class name

The class name subclause provides information necessary to identify and categorize a functional class. Every functional class has a unique name. The categorical information consists of a short name of three characters. The short name of the class is used in the specification of the short names of the families of that class.

6.1.1.2 Class introduction

The class introduction expresses the common intent or approach of those families to satisfy security objectives. The definition of functional classes does not reflect any formal taxonomy in the specification of the requirements.

The class introduction provides a figure describing the families in this class and the hierarchy of the components in each family, as explained in 6.2.

6.1.2 Family structure

Figure 4 illustrates the functional family structure in diagrammatic form.

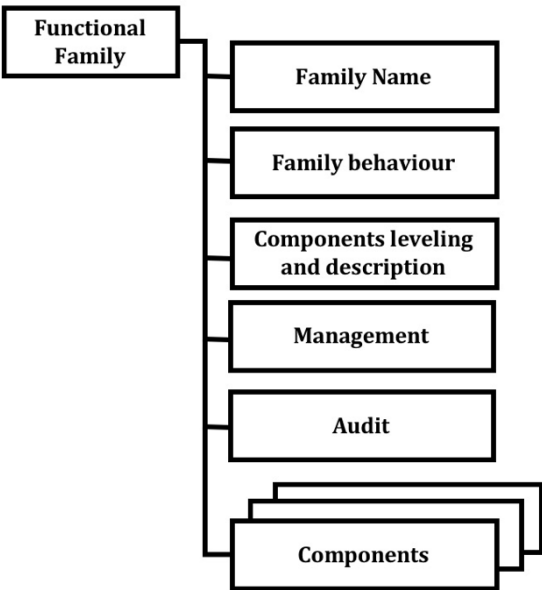


Figure 4 — Functional family structure



**6.1.2.1 Family name**

The family name subclause provides categorical and descriptive information necessary to identify and categorize a functional family. Every functional family has a unique name. The categorical information consists of a short name of seven characters, with the first three identical to the short name of the class followed by an underscore and the short name of the family as follows, XXX\_YYY. The unique short form of the family name provides the principal reference name for the security components.

**6.1.2.2 Family behaviour**

The family behaviour is the narrative description of the functional family stating its security objective and a general description of the functional requirements. These are described in greater detail below:

- a) The security objectives of the family address a security problem that **may** be solved with the help of a TOE that incorporates SFRs derived from a component of this family;
- b) The description of the *functional requirements* summarizes all the requirements that are included in the component(s). The description is aimed at authors of STs, PPs, PP-Modules or security functional packages who wish to assess whether the family is relevant to their specific requirements.

**6.1.2.3 Components leveling and description**

Functional families contain one or more components, any one of which **can** be selected for inclusion in STs, PPs, PP-Modules or security functional packages. The goal of this subclause is to provide information to users in selecting an appropriate functional component once the family has been identified as being a necessary or useful part of their security requirements.

This section of the functional family description describes the components available, and their rationale. The exact details of the components are contained within each component.

The relationships between components within a functional family **may** or **may** not be hierarchical. A component is hierarchical to another if it offers more security.

As explained in 6.2 the descriptions of the families provide a graphical overview of the hierarchy of the components in a family.

**6.1.2.4 Management**

The management clauses contain information for ST, PP, PP-Module, or security functional package authors to consider as management activities for a given component. The clauses reference components of the management class (FMT) and provide guidance regarding potential management activities that **may** be applied via operations to those components.

An author **may** select the indicated management components or **may** include other management requirements not listed to detail management activities. As such the information **should** be considered informative.

**6.1.2.5 Audit**

The *audit* requirements contain auditable events for the authors to select, if requirements from the class FAU, are included in the ST, PP, PP-Module, or security functional package. These requirements include security relevant events in terms of the various levels of detail supported by the components of the Security audit data generation (FAU\_GEN) family.

**EXAMPLE 1**

an audit note might include actions that are in terms of:

- Minimal - successful use of the security mechanism;



- Basic - any use of the security mechanism as well as relevant information regarding the security attributes involved;
- Detailed - any configuration changes made to the mechanism, including the actual configuration values before and after the change.

1067

1068 It **can** be observed that the categorization of auditable events is hierarchical.

EXAMPLE 2

For example, when Basic Audit Generation is desired, all auditable events identified as being both Minimal and Basic **should** be included in the PP/ST through the use of the appropriate assignment operation, except when the higher-level event simply provides more detail than the lower level event. When Detailed Audit Generation is desired, all identified auditable events (Minimal, Basic and Detailed) **should** be included in the PP/ST.

1069

1070 **Editors' Note**

1071 **Examples cannot contain requirements/recommendations.**

1072 **Is this intended to be a formal recommendation?**

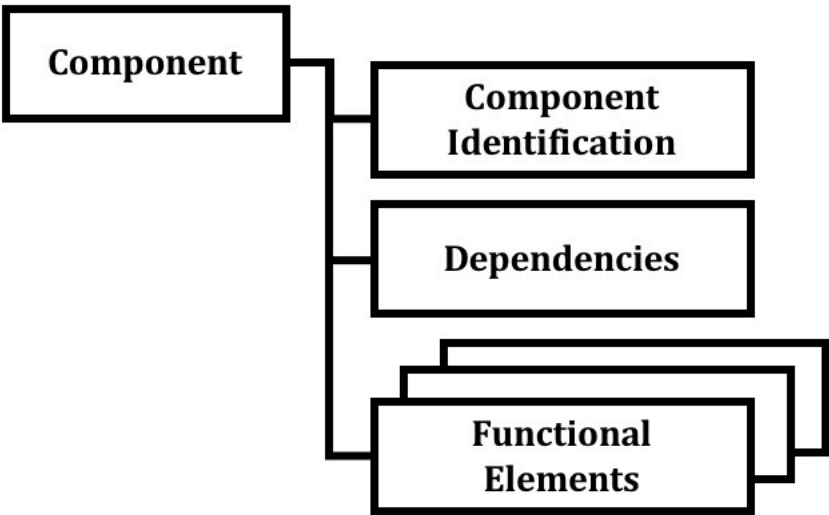
1073

1074 In the class FAU the rules governing the audit are explained in more detail.

1075 **6.1.3 Component structure**

1076 Figure 5 illustrates the functional component structure.

1077



1078 **Figure 5 — Functional component structure**

1079 **6.1.3.1 Component identification**

1080 The component identification subclause provides descriptive information necessary to identify,  
1081 categorize, register, and cross-reference a component. The following is provided as part of  
1082 every functional component:

1083 *A unique name.* The name reflects the purpose of the component.

1084 *A unique short name.* A unique short form of the functional component name. This short name  
1085 serves as the principal reference name for the categorization, registration, and cross-  
1086 referencing of the component. This short name reflects the class and family to which the  
1087 component belongs and the component number within the family.

1088 *A hierarchical-to list. A list of other components that this component is hierarchical to and for*  
 1089 *which this component can be used to satisfy dependencies to the listed components.*

#### 1090 **6.1.3.2 Functional elements**

1091 A set of elements is provided for each component. Each element is individually defined and is  
 1092 self-contained.

1093 A functional element is a part of a security functional component that if further divided would  
 1094 not yield a meaningful SFR. It is the smallest part of the taxonomy that is identified and  
 1095 recognized in the ISO/IEC 15408 series.

1096 When building packages, PPs and/or STs, it is not permitted to select only one or more  
 1097 elements from a component. The complete set of elements of a component must be selected for  
 1098 inclusion in a PP, PP-Module, security functional package or an ST.

1099 A unique short form of the functional element name is provided.

##### EXAMPLE

The component name FDP\_IFF.4.2 reads as follows:

- F - functional requirement,
- DP - class "User data protection",
- \_IFF - family "Information flow control functions",
- .4 - 4th component named "Partial elimination of illicit information flows",
- .2 - 2nd element of the component.

1100

#### 1101 **6.1.3.3 Dependencies**

1102 Dependencies among functional components arise when a component is not self-sufficient and  
 1103 relies upon the functionality of, or interaction with, another component for its own proper  
 1104 functioning.

1105 Each functional component provides a complete list of dependencies to other functional and  
 1106 assurance components. Some components may list "No dependencies". The components  
 1107 depended upon may in turn have dependencies on other components. The list provided in the  
 1108 components will be the direct dependencies. That is only references to the other functional  
 1109 components that are required for this component to perform its job properly. The indirect  
 1110 dependencies, that is the dependencies that result from the depended upon components can be  
 1111 found in Annex A of this document. It is noted that in some cases the dependency is optional in  
 1112 that a number of functional components are provided, where each one of them would be  
 1113 sufficient to satisfy the dependency.

##### EXAMPLE

FDP\_UIT.1 Data exchange integrity

1114 The dependency list identifies the minimum functional or assurance components needed to  
 1115 satisfy the security requirements associated with an identified component. Components that  
 1116 are hierarchical to the identified component may also be used to satisfy the dependency.

1117 The dependencies indicated in this document are normative and they shall be satisfied within a  
 1118 package, PP or ST. In situations where the indicated dependencies are not applicable, the author  
 1119 shall satisfy the dependency by providing a rationale why it is not applicable and may leave the  
 1120 depended upon component from the package, PP or ST.

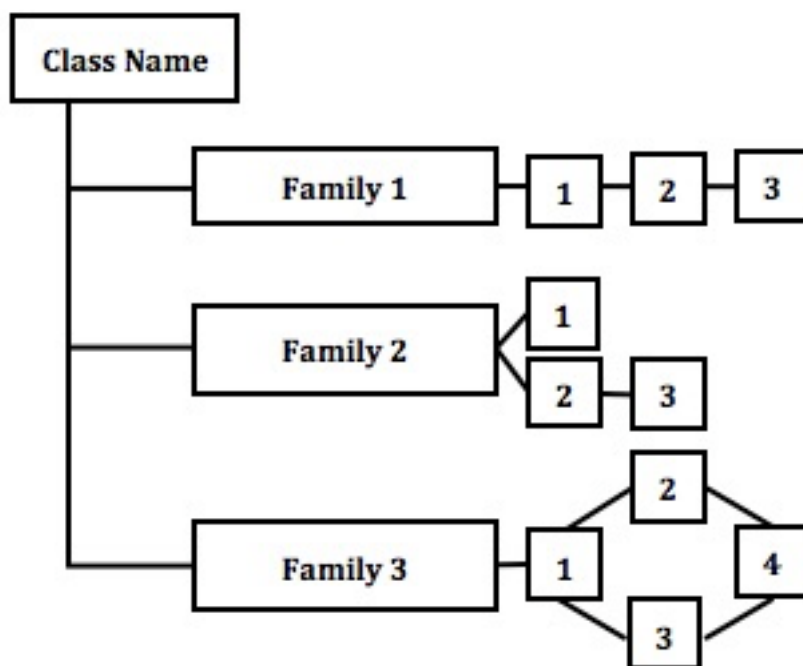
#### 1121 **6.2 Component catalogue**

1122 The grouping of the components in this document does not reflect any formal taxonomy.

This document contains classes of families and components, which are rough groupings on the basis of related function or purpose, presented in alphabetic order. At the start of each class is an informative diagram that indicates the taxonomy of each class, indicating the families in each class and the components in each family. Figure 6 is a useful indicator of the hierarchical relationship that **may** exist between components.

In the description of the functional components, a subclause identifies the dependencies between the component and any other components.

In each class, a figure describing the family hierarchy similar to Figure 6 is provided. In Figure 6 the first family, Family 1, contains three hierarchical components, where component 2 and component 3 **can** both be used to satisfy dependencies on component 1. Component 3 is hierarchical to component 2 and **can** also be used to satisfy dependencies on component 2.



**Figure 6 — Sample class decomposition diagram**

In Family 2 there are three components not all of which are hierarchical. Components 1 and 2 are hierarchical to no other components. Component 3 is hierarchical to component 2 and **can** be used to satisfy dependencies on component 2, but not to satisfy dependencies on component 1.

In Family 3, components 2, 3, and 4 are hierarchical to component 1. Components 2 and 3 are both hierarchical to component 1, but non-comparable. Component 4 is hierarchical to both component 2 and component 3.

These diagrams are meant to complement the text of the families and make identification of the relationships easier. They do not replace the “Hierarchical to:” note in each component that is the mandatory claim of hierarchy for each component.

### 6.2.1 Component changes highlighting

The relationship between components within a family is highlighted using a **bolding** convention. This bolding convention calls for the bolding of all new requirements. For hierarchical components, requirements are bolded when they are enhanced or modified beyond the requirements of the previous component. In addition, any new or enhanced permitted operations beyond the previous component are also highlighted using **bold** type.

7 Class FAU: Security audit

Editors' Note

The Editors' have removed examples given in clauses 7 – 17 since these should be places in the informative Annexes.

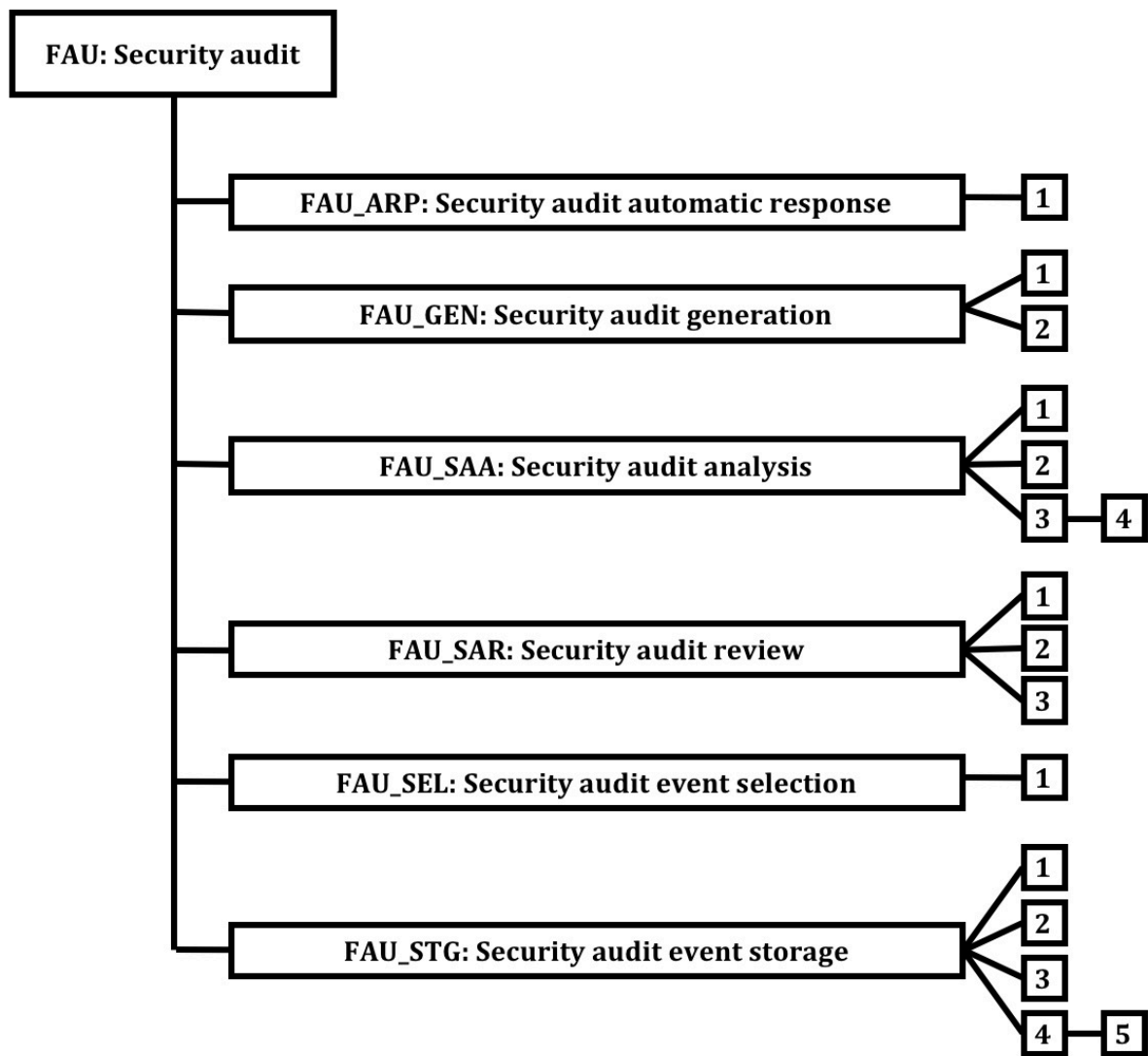
7.1 Class description

Security auditing involves recognizing, recording, storing, and analyzing information related to security relevant activities (i.e. activities controlled by the TSF). The resulting audit records can be examined to determine which security relevant activities took place and whom (which user) is responsible for them.

Figure 7 shows the decomposition of this class, it's families and components. Elements are not shown in the figure.

Annex C provides explanatory information for this class and should be consulted when using the components identified in this class.

Figure 7 — FAU: Security audit class decomposition



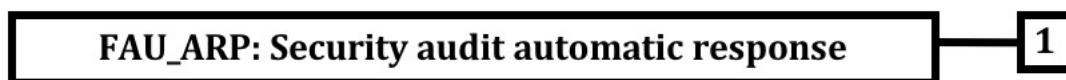
## 1168 7.2 Security audit automatic response (FAU\_ARP)

### 1169 7.2.1 Family behaviour

1170 This family defines the response to be taken in case of detected events indicative of a potential  
1171 security violation.

### 1172 7.2.2 Components leveling and description

1173 Figure 8 shows the component leveling for this family.



1174 **Figure 8 — FAU\_ARP: Component leveling**

1175 At FAU\_ARP.1 Security alarms, the TSF **shall** take actions in case a potential security violation is  
1176 detected.

### 1177 7.2.3 Management of FAU\_ARP.1

1178 The following actions **could** be considered for the management functions in FMT:

- 1179 a) the management (addition, removal, or modification) of actions.

### 1180 7.2.4 Audit of FAU\_ARP.1

1181 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1182 in the PP/ST:

- 1183 a) Minimal: Actions taken due to potential security violations.

### 1184 7.2.5 FAU\_ARP.1 Security alarms

#### 1185 7.2.5.1 Component relationships

1186 Hierarchical to: No other components.

1187 Dependencies: FAU\_SAA.1 Potential violation analysis

#### 1188 7.2.5.2 FAU\_ARP.1.1

1189 **The TSF **shall** take [assignment: *list of actions*] upon detection of a potential security**  
1190 **violation.**

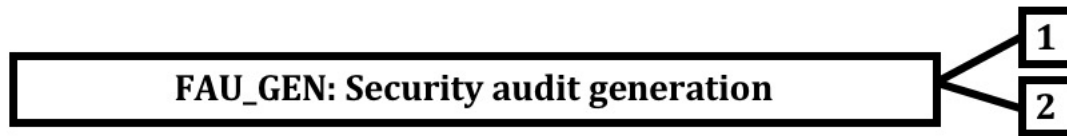
## 1191 7.3 Security audit data generation (FAU\_GEN)

### 1192 7.3.1 Family behaviour

1193 This family defines requirements for recording the occurrence of security relevant events that  
1194 take place under TSF control. This family identifies the level of auditing, enumerates the types  
1195 of events that **shall** be auditable by the TSF, and identifies the minimum set of audit-related  
1196 information that **should** be provided within various audit record types.

1197 **7.3.2 Components leveling and description**

1198 Figure 9 shows the component leveling for this family.



1199 **Figure 9 — FAU\_GEN: Component leveling**

1200 FAU\_GEN.1 Audit data generation defines the level of auditable events and specifies the list of  
 1201 data that **shall** be recorded in each record.

1202 At FAU\_GEN.2 User identity association, the TSF **shall** associate auditable events to individual  
 1203 user identities.

1204 **7.3.3 Management of FAU\_GEN.1, FAU\_GEN.2**

1205 The following actions **could** be considered for the management functions in FMT:

- 1206 a) There are no management activities foreseen.

1207 **7.3.4 Audit of FAU\_GEN.1, FAU\_GEN.2**

1208 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 1209 in the PP/ST:

- 1210 a) There are no auditable events foreseen.

1211 **7.3.5 FAU\_GEN.1 Audit data generation**

1212 **7.3.5.1 Component relationships**

1213 Hierarchical to: No other components.

1214 Dependencies: FPT\_STM.1 Reliable time stamps

1215 **7.3.5.2 FAU\_GEN.1.1**

1216 The TSF **shall** be able to generate audit data of the following auditable events:

- 1217 a) Start-up and shutdown of the audit functions;  
 1218 b) All auditable events for the [selection, choose one of: *minimum, basic,*  
 1219 *detailed, not specified*] level of audit; and  
 1220 c) [assignment: other specifically defined auditable events].

1221 **7.3.5.3 FAU\_GEN.1.2**

1222 The TSF **shall** record within the audit data at least the following information:

- 1223 a) Date and time of the auditable event, type of event, subject identity (if  
 1224 applicable), and the outcome (success or failure) of the event; and  
 1225 b) For each auditable event type, based on the auditable event definitions of the  
 1226 functional components included in the PP/ST, [assignment: *other audit*  
 1227 *relevant information*].

## 1228 7.3.6 FAU\_GEN.2 User identity association

### 1229 7.3.6.1 Component relationships

1230	Hierarchical to:	No other components.
1231	Dependencies:	FAU_GEN.1 Audit data generation
1232		FIA_UID.1 Timing of identification

### 1233 7.3.6.2 FAU\_GEN.2.1

1234 For audit events resulting from actions of identified users, the TSF **shall** be able to  
1235 associate each auditable event with the identity of the user that caused the event.

## 1236 7.4 Security audit analysis (FAU\_SAA)

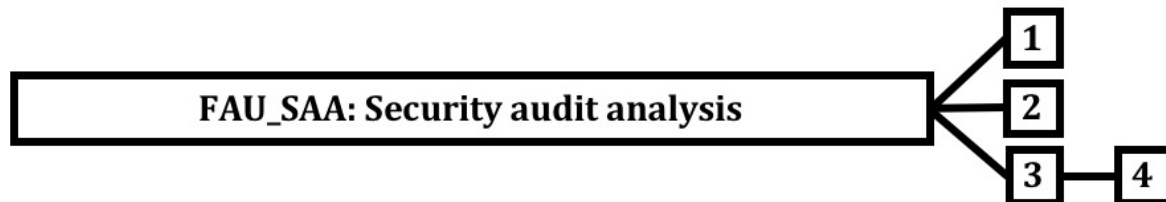
### 1237 7.4.1 Family behaviour

1238 This family defines requirements for automated means that analyze system activity and audit  
1239 data looking for possible or real security violations. This analysis **may** work in support of  
1240 intrusion detection, or automatic response to a potential security violation.

1241 The actions to be taken based on the detection **can** be specified using the Security audit  
1242 automatic response (FAU\_ARP) family as desired.

### 1243 7.4.2 Components leveling and description

1244 Figure 10 shows the component leveling for this family.



1245 **Figure 10 — FAU\_SAA: Component leveling**

1246 In FAU\_SAA.1 Potential violation analysis, basic threshold detection on the basis of a fixed rule  
1247 set is required.

1248 In FAU\_SAA.2 Profile based anomaly detection, the TSF maintains individual profiles of system  
1249 usage, where a profile represents the historical patterns of usage performed by members of the  
1250 profile target group. A profile target group refers to a group of one or more individuals who  
1251 interact with the TSF. Each member of a profile target group is assigned an individual suspicion  
1252 rating that represents how well that member's current activity corresponds to the established  
1253 patterns of usage represented in the profile. This analysis **can** be performed at runtime or  
1254 during a post-collection batch-mode analysis.

1255 In FAU\_SAA.3 Simple attack heuristics, the TSF **shall** be able to detect the occurrence of  
1256 signature events that represent a significant threat to enforcement of the SFRs. This search for  
1257 signature events **may** occur in real-time or during a post-collection batch-mode analysis.

1258 In FAU\_SAA.4 Complex attack heuristics, the TSF **shall** be able to represent and detect multi-  
1259 step intrusion scenarios. The TSF is able to compare system events (possibly performed by  
1260 multiple individuals) against event sequences known to represent entire intrusion scenarios.  
1261 The TSF **shall** be able to indicate when a signature event or event sequence is found that  
1262 indicates a potential violation of the enforcement of the SFRs.

1263 **7.4.3 Management of FAU\_SAA.1**

1264 The following actions **could** be considered for the management functions in FMT:

- 1265 a) Maintenance of the rules by (adding, modifying, deletion) of rules from the set of  
1266 rules.

1267 **7.4.4 Management of FAU\_SAA.2**

1268 The following actions **could** be considered for the management functions in FMT:

- 1269 a) Maintenance (deletion, modification, addition) of the group of users in the profile  
1270 target group.

1271 **7.4.5 Management of FAU\_SAA.3**

1272 The following actions **could** be considered for the management functions in FMT:

- 1273 a) Maintenance (deletion, modification, addition) of the subset of system events.

1274 **7.4.6 Management of FAU\_SAA.4**

1275 The following actions **could** be considered for the management functions in FMT:

- 1276 a) Maintenance (deletion, modification, addition) of the subset of system events;  
1277 b) Maintenance (deletion, modification, addition) of the set of sequences of system  
1278 events.

1279 **7.4.7 Audit of FAU\_SAA.1, FAU\_SAA.2, FAU\_SAA.3, FAU\_SAA.4**

1280 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1281 in the PP/ST:

- 1282 a) Minimal: Enabling and disabling of any of the analysis mechanisms;  
1283 b) Minimal: Automated responses performed by the tool.

1284 **7.4.8 FAU\_SAA.1 Potential violation analysis**

1285 **7.4.8.1 Component relationships**

1286	Hierarchical to:	No other components.
1287	Dependencies:	FAU_GEN.1 Audit data generation

1288 **7.4.8.2 FAU\_SAA.1.1**

1289 **The TSF **shall** be able to apply a set of rules in monitoring the audited events and based**  
1290 **upon these rules indicate a potential violation of the enforcement of the SFRs.**

1291 **7.4.8.3 FAU\_SAA.1.2**

1292 **The TSF **shall** enforce the following rules for monitoring audited events:**

- 1293 a) **Accumulation or combination of [assignment: *subset of defined auditable***  
1294 ***events*] known to indicate a potential security violation;**  
1295 b) **[assignment: *any other rules*].**

1296 **7.4.9 FAU\_SAA.2 Profile based anomaly detection**

1297 **7.4.9.1 Component relationships**

1298	Hierarchical to:	No other components.
1299	Dependencies:	FIA_UID.1 Timing of identification



1300 7.4.9.2 FAU\_SAA.2.1

1301 The TSF **shall** be able to maintain profiles of system usage, where an individual profile  
 1302 represents the historical patterns of usage performed by the member(s) of [assignment:  
 1303 *the profile target group*].

1304 7.4.9.3 FAU\_SAA.2.2

1305 The TSF **shall** be able to maintain a suspicion rating associated with each user whose  
 1306 activity is recorded in a profile, where the suspicion rating represents the degree to  
 1307 which the user's current activity is found inconsistent with the established patterns of  
 1308 usage represented in the profile.

1309 7.4.9.4 FAU\_SAA.2.3

1310 The TSF **shall** be able to indicate a possible violation of the enforcement of the SFRs when  
 1311 a user's suspicion rating exceeds the following threshold conditions [assignment:  
 1312 *conditions under which anomalous activity is reported by the TSF*].

1313 7.4.10 FAU\_SAA.3 Simple attack heuristics

1314 7.4.10.1 Component relationships

1315	Hierarchical to:	No other components.
1316	Dependencies:	No dependencies.

1317 7.4.10.2 FAU\_SAA.3.1

1318 The TSF **shall** be able to maintain an internal representation of the following signature  
 1319 events [assignment: *a subset of system events*] that **may** indicate a violation of the  
 1320 enforcement of the SFRs.

1321 7.4.10.3 FAU\_SAA.3.2

1322 The TSF **shall** be able to compare the signature events against the record of system  
 1323 activity discernible from an examination of [assignment: *the information to be used to*  
 1324 *determine system activity*].

1325 7.4.10.4 FAU\_SAA.3.3

1326 The TSF **shall** be able to indicate a potential violation of the enforcement of the SFRs  
 1327 when a system event is found to match a signature event that indicates a potential  
 1328 violation of the enforcement of the SFRs.

1329 7.4.11 FAU\_SAA.4 Complex attack heuristics

1330 7.4.11.1 Component relationships

1331	Hierarchical to:	FAU_SAA.3 Simple attack heuristics
1332	Dependencies:	No dependencies.

1333 7.4.11.2 FAU\_SAA.4.1

1334 The TSF **shall** be able to maintain an internal representation of the following **event sequences**  
 1335 **of known intrusion scenarios** [assignment: *list of sequences of system events whose*  
 1336 *occurrence are representative of known penetration scenarios*] and the following signature  
 1337 events [assignment: *a subset of system events*] that **may** indicate a **potential** violation of the  
 1338 enforcement of the SFRs.

1339 **7.4.11.3 FAU\_SAA.4.2**

1340 The TSF **shall** be able to compare the signature events **and event sequences** against the record  
 1341 of system activity discernible from an examination of [assignment: *the information to be used to*  
 1342 *determine system activity*].

1343 **7.4.11.4 FAU\_SAA.4.3**

1344 The TSF **shall** be able to indicate a potential violation of the enforcement of the SFRs when  
 1345 system **activity** is found to match a signature event **or event sequence** that indicates a  
 1346 potential violation of the enforcement of the SFRs.

1347 **7.5 Security audit review (FAU\_SAR)**1348 **7.5.1 Family behaviour**

1349 This family defines the requirements for audit tools, made available by the TOE to authorized  
 1350 users, in order to assist in the review of audit data.

1351 **Editors' Note**

1352 Editor suggests "This family defines the requirements for tools that are made available to authorized  
 1353 users to assist in the review of audit data."

1354 **7.5.2 Components leveling and description**

1355 Figure 11 shows the component leveling for this family.



1356 **Figure 11 — FAU\_SAR: Component leveling**

1357

1358 FAU\_SAR.1 Audit review, provides the capability to read information from the audit data.

1359 FAU\_SAR.2 Restricted audit review, requires that there are no other users except those that  
 1360 have been identified in FAU\_SAR.1 Audit review that **can** read the information.

1361 FAU\_SAR.3 Selectable audit review, requires audit review tools to select the audit data to be  
 1362 reviewed based on criteria.

1363 **7.5.3 Management of FAU\_SAR.1**

1364 The following actions **could** be considered for the management functions in FMT:

- 1365 a) Maintenance (deletion, modification, addition) of the group of users with read  
 1366 access right to the audit records.

1367 **7.5.4 Management of FAU\_SAR.2, FAU\_SAR.3**

1368 The following actions **could** be considered for the management functions in FMT:

- 1369 a) There are no management activities foreseen.

1370 **7.5.5 Audit of FAU\_SAR.1**

1371 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 1372 in the PP/ST:

1373 a) Basic: Reading of information from the audit records.

#### 1374 7.5.6 Audit of FAU\_SAR.2

1375 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1376 in the PP/ST:

1377 a) Basic: Unsuccessful attempts to read information from the audit records.

#### 1378 7.5.7 Audit of FAU\_SAR.3

1379 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1380 in the PP/ST:

1381 a) Detailed: the parameters used for the viewing.

#### 1382 7.5.8 FAU\_SAR.1 Audit review

##### 1383 7.5.8.1 Component relationships

1384 Hierarchical to: No other components.

1385 Dependencies: FAU\_GEN.1 Audit data generation

##### 1386 7.5.8.2 FAU\_SAR.1.1

1387 The TSF **shall** provide [assignment: *authorized users*] with the capability to read  
1388 [assignment: *list of audit information*] from the audit data.

##### 1389 7.5.8.3 FAU\_SAR.1.2

1390 The TSF **shall** provide the audit data in a manner suitable for the user to interpret the  
1391 information.

#### 1392 7.5.9 FAU\_SAR.2 Restricted audit review

##### 1393 7.5.9.1 Component relationships

1394 Hierarchical to: No other components.

1395 Dependencies: FAU\_SAR.1 Audit review

##### 1396 7.5.9.2 FAU\_SAR.2.1

1397 The TSF **shall** prohibit all users read access to the audit data, except those users that  
1398 have been granted explicit read-access.

#### 1399 7.5.10 FAU\_SAR.3 Selectable audit review

1400 Hierarchical to: No other components.

1401 Dependencies: FAU\_SAR.1 Audit review

##### 1402 7.5.10.1 FAU\_SAR.3.1

1403 The TSF **shall** provide the ability to apply [assignment: *methods of selection and/or*  
1404 *ordering*] of audit data based on [assignment: *criteria with logical relations*].

### 1405 7.6 Security audit event selection (FAU\_SEL)

#### 1406 7.6.1 Family behaviour

1407 This family defines requirements to select the set of events to be audited during TOE operation  
1408 from the set of all auditable events.

1409 **7.6.2 Components leveling and description**

1410 Figure 12 shows the component leveling for this family.

1411 **Figure 12 — FAU\_SEL: Component leveling**

1412

1413 FAU\_SEL.1 Selective audit, requires the ability to select the set of events to be audited from the  
 1414 set of all auditable events, identified in FAU\_GEN.1 Audit data generation, based upon attributes  
 1415 to be specified by the PP/ST author.

1416 **7.6.3 Management of FAU\_SEL.1**1417 The following actions **could** be considered for the management functions in FMT:

1418 a) Maintenance of the rights to view/modify the audit data.

1419 **7.6.4 Audit of FAU\_SEL.1**

1420 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 1421 in the PP/ST:

1422 a) Minimal: All modifications to the audit configuration that occur while the audit  
 1423 collection functions are operating.

1424 **7.6.5 FAU\_SEL.1 Selective audit**1425 **7.6.5.1 Component relationships**

1426 Hierarchical to: No other components.

1427 Dependencies: FAU\_GEN.1 Audit data generation  
 1428 FMT\_MTD.1 Management of TSF data

1429 **7.6.5.2 FAU\_SEL.1.1**

1430 **The TSF **shall** be able to select the set of events to be audited from the set of all auditable**  
 1431 **events based on the following attributes:**

1432 a) [selection: *object identity, user identity, subject identity, host identity, event*  
 1433 *type*]

1434 b) [assignment: *list of additional attributes that audit selectivity is based upon*]

1435

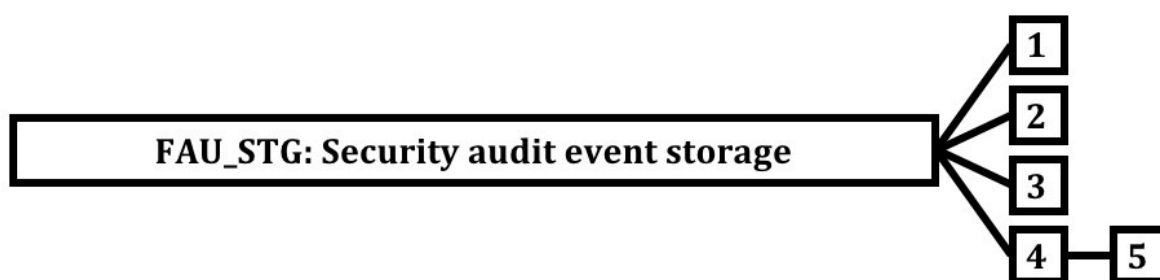
## 7.7 Security audit data storage (FAU\_STG)

### 7.7.1 Family behaviour

This family defines the requirements for the TSF to be able to create and maintain a secure audit trail. Stored audit data refers to those data stored within an audit trail, and not to any audit data that has been retrieved (to temporary storage) through selection.

### 7.7.2 Components leveling and description

Figure 13 shows the component leveling for this family.



**Figure 13 — FAU\_STG: Component leveling**

FAU\_STG.1 Audit data storage location, requires that the storage location(s) for audit data be specified

FAU\_STG.2 Protected audit data storage, requires that protections are placed on the audit data. It will be protected from unauthorized deletion and/or modification.

FAU\_STG.3 Guarantees of audit data availability, specifies the guarantees that the TSF maintains over the audit data given the occurrence of an undesired condition.

FAU\_STG.4 Prevention of audit data loss, specifies actions in case the storage for audit data is full.

FAU\_STG.5 Action in case of possible audit data loss, specifies actions to be taken if a threshold on the stored audit data is exceeded.

### 7.7.3 Management of FAU\_STG.1

The following actions **could** be considered for the management functions in FMT:

- a) Maintenance of remote audit storage locations

### 7.7.4 Management of FAU\_STG.2

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

### 7.7.5 Management of FAU\_STG.3

The following actions **could** be considered for the management functions in FMT:

- a) Maintenance of the parameters that control the audit data storage capability.

### 7.7.6 Management of FAU\_STG.4

The following actions **could** be considered for the management functions in FMT:

- 1466 a) Maintenance (deletion, modification, addition) of actions to be taken in case of  
1467 imminent audit data storage failure.
- 1468 **7.7.7 Management of FAU\_STG.5**
- 1469 The following actions **could** be considered for the management functions in FMT:
- 1470 a) Maintenance (deletion, modification, addition) of actions to be taken in case of  
1471 audit data storage failure.
- 1472 **7.7.8 Audit of FAU\_STG.1**
- 1473 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1474 in the PP/ST:
- 1475 a) Basic: Changes in the location of remote audit data storage.
- 1476 **7.7.9 Audit of FAU\_STG.2, FAU\_STG.4**
- 1477 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1478 in the PP/ST:
- 1479 a) There are no auditable events foreseen.
- 1480 **7.7.10 Audit of FAU\_STG.3**
- 1481 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1482 in the PP/ST:
- 1483 a) Basic: Actions taken due to exceeding of a threshold.
- 1484 **7.7.11 Audit of FAU\_STG.5**
- 1485 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1486 in the PP/ST:
- 1487 a) Basic: Actions taken due to the audit data storage failure.
- 1488 **7.7.12 FAU\_STG.1 Audit data storage location**
- 1489 **7.7.12.1 Component relationships**
- |      |                  |                                   |
|------|------------------|-----------------------------------|
| 1490 | Hierarchical to: | No other components               |
| 1491 | Dependencies:    | FAU_GEN.1 Audit data generation   |
| 1492 |                  | FTP_ITC Inter-TSF trusted channel |
- 1493 **7.7.12.2 FAU\_STG.1.1**
- 1494 **The TSF **shall** be able to store generated audit data on the [selection: *TOE itself*, transmit**  
1495 ***the generated audit data to an external IT entity using a trusted channel according to***  
1496 ***FTP\_ITC, [assignment: other storage location(s)].]***
- 1497 **7.7.13 FAU\_STG.2 Protected audit data storage**
- 1498 **7.7.13.1 Component relationships**
- |      |                  |                                 |
|------|------------------|---------------------------------|
| 1499 | Hierarchical to: | No other components             |
| 1500 | Dependencies:    | FAU_GEN.1 Audit data generation |
- 1501 **7.7.13.2 FAU\_STG.2.1**
- 1502 **The TSF **shall** protect the stored audit data in the audit trail from unauthorized deletion.**

## 1503 7.7.13.3 FAU\_STG.2.2

1504 The TSF **shall** be able to [selection, choose one of: *prevent, detect*] unauthorized  
 1505 modifications to the stored audit data in the audit trail.

## 1506 7.7.14 FAU\_STG.3 Guarantees of audit data availability

## 1507 7.7.14.1 Component relationships

1508 Hierarchical to: No other components

1509 Dependencies: FAU\_GEN.1 Audit data generation

## 1510 7.7.14.2 FAU\_STG.3.1

1511 The TSF **shall** ensure that [assignment: *metric for saving audit data*] stored audit data  
 1512 will be maintained when the following conditions occur: [selection: *audit data storage*  
 1513 *exhaustion, failure, attack*].

## 1514 7.7.15 FAU\_STG.4 Prevention of audit data loss

## 1515 7.7.15.1 Component relationships

1516 Hierarchical to: No other components

1517 Dependencies: FAU\_STG.2 Protected audit data storage

1518 FAU\_GEN.1 Audit data generation

## 1519 7.7.15.2 FAU\_STG.4.1

1520 The TSF **shall** [selection: *ignore audited events, "prevent audited events, except those*  
 1521 *taken by the authorized user with special rights", overwrite the oldest stored audit*  
 1522 *records*], [assignment: *other actions to be taken in case of audit storage failure and*  
 1523 *conditions for the actions*] if the audit data storage is full.

## 1524 7.7.16 FAU\_STG.5 Action in case of possible audit data loss

## 1525 7.7.16.1 Component relationships

1526 Hierarchical to: FAU\_STG.4 Prevention of audit data loss

1527 Dependencies: FAU\_STG.2 Protected audit data storage

## 1528 7.7.16.2 FAU\_STG.5.1

1529 The TSF **shall** [assignment: *actions to be taken in case of possible audit data storage failure*]  
 1530 if the audit data storage **exceeds** [assignment: *pre-defined limit*].

1531

8 Class FCO: Communication

8.1 Class description

This class provides two families specifically concerned with assuring the identity of a party participating in a data exchange. These families are related to assuring the identity of the originator of transmitted information (proof of origin) and assuring the identity of the recipient of transmitted information (proof of receipt). These families ensure that an originator cannot deny having sent the message, nor can the recipient deny having received it. Figure 14 shows the decomposition of the class.

Figure 14 shows the decomposition of this class, it's families and components. Elements are not shown in the figure.

Annex D provides explanatory information for this class and should be consulted when using the components identified in this class.

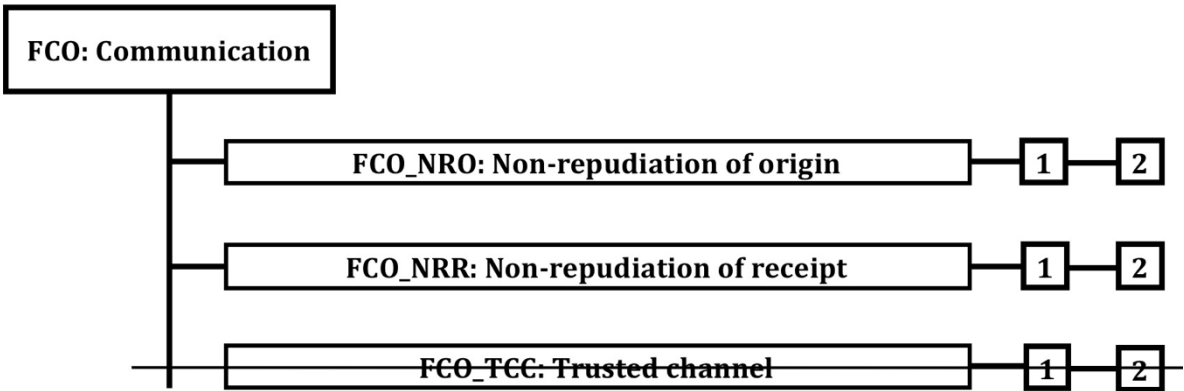


Figure 14 — FCO: Communication class decomposition

8.2 Non-repudiation of origin (FCO\_NRO)

8.2.1 Family behaviour

Non-repudiation of origin ensures that the originator of information cannot successfully deny having sent the information. This family requires that the TSF provide a method to ensure that a subject that receives information during a data exchange is provided with evidence of the origin of the information. This evidence can then be verified by either this subject or other subjects.

8.2.2 Components leveling and description

Figure 15 shows the component leveling for this family.



Figure 15 — FCO\_NRO: Component leveling

FCO\_NRO.1 Selective proof of origin, requires the TSF to provide subjects with the capability to request evidence of the origin of information.

FCO\_NRO.2 Enforced proof of origin, requires that the TSF always generate evidence of origin for transmitted information.

8.2.3 Management of FCO\_NRO.1, FCO\_NRO.2

The following actions could be considered for the management functions in FMT:



- 1560 a) The management of changes to information types, fields, originator attributes and  
1561 recipients of evidence.

#### 1562 8.2.4 Audit of FCO\_NRO.1

1563 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1564 in the PP/ST:

- 1565 a) Minimal: The identity of the user who requested that evidence of origin would be  
1566 generated.
- 1567 b) Minimal: The invocation of the non-repudiation service.
- 1568 c) Basic: Identification of the information, the destination, and a copy of the evidence  
1569 provided.
- 1570 d) Detailed: The identity of the user who requested a verification of the evidence.

#### 1571 8.2.5 Audit of FCO\_NRO.2

1572 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1573 in the PP/ST:

- 1574 a) Minimal: The invocation of the non-repudiation service.
- 1575 b) Basic: Identification of the information, the destination, and a copy of the evidence  
1576 provided.
- 1577 c) Detailed: The identity of the user who requested a verification of the evidence.

#### 1578 8.2.6 FCO\_NRO.1 Selective proof of origin

##### 1579 8.2.6.1 Component relationships

- |      |                  |                                    |
|------|------------------|------------------------------------|
| 1580 | Hierarchical to: | No other components.               |
| 1581 | Dependencies:    | FIA_UID.1 Timing of identification |

##### 1582 8.2.6.2 FCO\_NRO.1.1

1583 The TSF **shall** be able to generate evidence of origin for transmitted [assignment: *list of*  
1584 *information types*] at the request of the [selection: *originator, recipient, [assignment: list*  
1585 *of third parties*]].

##### 1586 8.2.6.3 FCO\_NRO.1.2

1587 The TSF **shall** be able to relate the [assignment: *list of attributes*] of the originator of the  
1588 information, and the [assignment: *list of information fields*] of the information to which  
1589 the evidence applies.

##### 1590 8.2.6.4 FCO\_NRO.1.3

1591 The TSF **shall** provide a capability to verify the evidence of origin of information to  
1592 [selection: *originator, recipient, [assignment: list of third parties]*] given [assignment:  
1593 *limitations on the evidence of origin*].

#### 1594 8.2.7 FCO\_NRO.2 Enforced proof of origin

##### 1595 8.2.7.1 Component relationships

- |      |                  |                                     |
|------|------------------|-------------------------------------|
| 1596 | Hierarchical to: | FCO_NRO.1 Selective proof of origin |
| 1597 | Dependencies:    | FIA_UID.1 Timing of identification  |

1598 **8.2.7.2 FCO\_NRO.2.1**

1599 The TSF **shall enforce the generation of** evidence of origin for transmitted [assignment: *list of*  
1600 *information types*] at **all times**.

1601 **8.2.7.3 FCO\_NRO.2.2**

1602 The TSF **shall** be able to relate the [assignment: *list of attributes*] of the originator of the  
1603 information, and the [assignment: *list of information fields*] of the information to which the  
1604 evidence applies.

1605 **8.2.7.4 FCO\_NRO.2.3**

1606 The TSF **shall** provide a capability to verify the evidence of origin of information to [selection:  
1607 *originator, recipient, [assignment: list of third parties]*] given [assignment: *limitations on the*  
1608 *evidence of origin*].

1609 **8.3 Non-repudiation of receipt (FCO\_NRR)**1610 **8.3.1 Family behaviour**

1611 Non-repudiation of receipt ensures that the recipient of information cannot successfully deny  
1612 receiving the information. This family requires that the TSF provide a method to ensure that a  
1613 subject that transmits information during a data exchange is provided with evidence of receipt  
1614 of the information. This evidence **can** then be verified by either this subject or other subjects.

1615 **8.3.2 Components leveling and description**

1616 Figure 16 shows the component leveling for this family.



1617 **Figure 16 — FCO\_NRR: Component leveling**

1618 FCO\_NRR.1 Selective proof of receipt, requires the TSF to provide subjects with a capability to  
1619 request evidence of the receipt of information.

1620 FCO\_NRR.2 Enforced proof of receipt, requires that the TSF always generate evidence of receipt  
1621 for received information.

1622 **8.3.3 Management of FCO\_NRR.1, FCO\_NRR.2**

1623 The following actions **could** be considered for the management functions in FMT:

- 1624 a) The management of changes to information types, fields, originator attributes and  
1625 third-party recipients of evidence.

1626 **8.3.4 Audit of FCO\_NRR.1**

1627 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1628 in the PP/ST:

- 1629 a) Minimal: The identity of the user who requested that evidence of receipt would be  
1630 generated.
- 1631 b) Minimal: The invocation of the non-repudiation service.
- 1632 c) Basic: Identification of the information, the destination, and a copy of the evidence  
1633 provided.
- 1634 d) Detailed: The identity of the user who requested a verification of the evidence.

### 1635 8.3.5 Audit of FCO\_NRR.2

1636 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1637 in the PP/ST:

- 1638 a) Minimal: The invocation of the non-repudiation service.
- 1639 b) Basic: Identification of the information, the destination, and a copy of the evidence  
1640 provided.
- 1641 c) Detailed: The identity of the user who requested a verification of the evidence.

### 1642 8.3.6 FCO\_NRR.1 Selective proof of receipt

#### 1643 8.3.6.1 Component relationships

1644	Hierarchical to:	No other components.
1645	Dependencies:	FIA_UID.1 Timing of identification

#### 1646 8.3.6.2 FCO\_NRR.1.1

1647 The TSF **shall** be able to generate evidence of receipt for received [assignment: *list of*  
1648 *information types*] at the request of the [selection: *originator, recipient, [assignment: list*  
1649 *of third parties]*].

#### 1650 8.3.6.3 FCO\_NRR.1.2

1651 The TSF **shall** be able to relate the [assignment: *list of attributes*] of the recipient of the  
1652 information, and the [assignment: *list of information fields*] of the information to which  
1653 the evidence applies.

#### 1654 8.3.6.4 FCO\_NRR.1.3

1655 The TSF **shall** provide a capability to verify the evidence of receipt of information to  
1656 [selection: *originator, recipient, [assignment: list of third parties]*] given [assignment:  
1657 *limitations on the evidence of receipt*].

### 1658 8.3.7 FCO\_NRR.2 Enforced proof of receipt

#### 1659 8.3.7.1 Component relationships

1660	Hierarchical to:	FCO_NRR.1 Selective proof of receipt
1661	Dependencies:	FIA_UID.1 Timing of identification

#### 1662 8.3.7.2 FCO\_NRR.2.1

1663 The TSF **shall enforce the generation of** evidence of receipt for received [assignment: *list of*  
1664 *information types*] at **all times**.

#### 1665 8.3.7.3 FCO\_NRR.2.2

1666 The TSF **shall** be able to relate the [assignment: *list of attributes*] of the recipient of the  
1667 information, and the [assignment: *list of information fields*] of the information to which the  
1668 evidence applies.

#### 1669 8.3.7.4 FCO\_NRR.2.3

1670 The TSF **shall** provide a capability to verify the evidence of receipt of information to [selection:  
1671 *originator, recipient, [assignment: list of third parties]*] given [assignment: *limitations on the*  
1672 *evidence of receipt*].

**8.4 Trusted channel (FCO\_TCC)****Editors' note**

This family is based on N5087, which has also been sent to the CCDB for comment and suggested text.

Editors' note that WD2 DE/JM2 request removing FCO\_TCC in favor of FTP\_PRO, however this was deferred pending receipt of input from the CCDB.

This family will be removed in the next draft unless contributions are received to the contrary.

**8.4.1 Family behaviour**

A trusted channel is a bidirectional communication channel between the TOE and a user. The TSF mediate the initialization of the trusted channel (including the definition of the security properties of the trusted channel) and control the security functions provided by the trusted channel. After setting up such a trusted channel, communication between the TSF and the other trusted IT product will be protected against one or more security threats. The type of threats the channel protects against need to be defined in the security functional requirement.

A trusted communication channel **can** be initiated upon request of a subject within the TOE, upon request of an external user, or by request of either entity. The TSF **may** limit the initialization of a trusted channel to the external user, to the subject requesting the initialization, allow for both to request the initialization of the trusted channel or **may** itself decide to initiate a trusted channel based on defined criteria that require such a trusted channel to be used.

The security properties of a trusted communication channel **may** be static or dynamic. In the case of dynamic security properties, the management of those security properties (their initialization, the authorization required to modify those properties, the conditions that need to be satisfied before the properties **can** be modified) needs to be defined.

**8.4.2 Components leveling and description**

Figure 17 shows the component leveling for this family.



**Figure 17 — FCO\_TCC: Component leveling**

FCO\_TCC.1 Trusted Communication Channel with fixed security properties, requires the TSF to provide users and/or subjects with a trusted communication channels with fixed security properties.

FCO\_TCC.2 Trusted Communication Channel with selectable security properties, requires the TSF to provide users and/or subjects with a trusted communication channels with selectable security properties.

**8.4.3 Management of FCO\_TCC.1, FCO\_TCC.2**

The following actions **could** be considered for the management functions in FMT:

a) ????

**8.4.4 Audit of FCO\_TCC.1, FCO\_TCC.2**

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

a) Successful/unsuccessful attempts to initiate the trusted channel

b) Violations of a security property of the trusted channel

1713 **8.4.5 — FCO\_TCC.1 Trusted Communication Channel with fixed security properties**

1714 **8.4.5.1 — Component relationships**

1715 Hierarchical to: \_\_\_\_\_ No other components.

1716 Dependencies: \_\_\_\_\_ None

1717 **8.4.5.2 — FCO\_TCC.1.1**

1718 The TSF **shall** provide the capability to set up a trusted communication channel between  
1719 ~~{assignment: type of subject}~~ and ~~{assignment: type of user}~~ that is logically distinct from  
1720 other communication channels.

1721 **8.4.5.3 — FCO\_TCC.1.2**

1722 The TSF **shall** permit ~~{selection: user, subject, user, or subject}~~ to request the  
1723 establishment of a trusted channel.

1724 **8.4.5.4 — FCO\_TCC.1.3**

1725 The TSF **shall** support the following security properties for the trusted channel:  
1726 ~~{selection: confidentiality protection, integrity protection, replay protection, user~~  
1727 ~~authentication, TSF authentication to the user, non-repudiation of origin, nonrepudiation of~~  
1728 ~~receipt, {assignment: other security properties}]~~.

1729 **8.4.5.5 — FCO\_TCC.1.4**

1730 The TSF **shall** implement the trusted channel in compliance with the following security  
1731 standards ~~{assignment: list of security standards or none}~~ using the following options  
1732 ~~{selection: {assignment: list of options}, none}~~.

1733 **8.4.6 — FCO\_TCC.2 Trusted Communication Channel with selectable security properties**

1734 **8.4.6.1 — Component relationships**

1735 Hierarchical to: \_\_\_\_\_ FCO\_TCC.1 Trusted Communication Channel with  
1736 fixed security properties

1737 Dependencies: \_\_\_\_\_ None

1738 **8.4.6.2 — FCO\_TCC.2.1**

1739 The TSF **shall** provide the capability to set up a trusted communication channel between  
1740 ~~{assignment: type of subject}~~ and ~~{assignment: type of user}~~ that is logically distinct from other  
1741 communication channels.

1742 **8.4.6.3 — FCO\_TCC.2.2**

1743 The TSF **shall** permit ~~{selection: user, subject, user, or subject}~~ to request the establishment of a  
1744 trusted channel.

1745 **8.4.6.4 — FCO\_TCC.2.3**

1746 The TSF **shall** support the following security properties for the trusted channel: ~~{selection:~~  
1747 ~~confidentiality protection, integrity protection, replay protection, user authentication, TSF~~  
1748 ~~authentication to the user, non-repudiation of origin, nonrepudiation of receipt, {assignment:~~  
1749 ~~other security properties}]~~.

1750 **8.4.6.5 — FCO\_TCC.2.4**

1751 The TSF ~~shall~~ implement the trusted channel in compliance with the following security  
 1752 standards [assignment: *list of security standards or none*] using the following options [selection:  
 1753 [assignment: *list of options*], *none*].

1754 **8.4.6.6 — FCO\_TCC.2.5**

1755 The TSF ~~shall~~ allow the following security properties to be selectable when the trusted  
 1756 channel is established: [selection: *confidentiality protection, integrity protection, replay*  
 1757 *protection, user authentication, TSF authentication to the user, nonrepudiation of origin,*  
 1758 *non-repudiation of receipt, [assignment: other security properties]]].*

1759 **8.4.6.7 — FCO\_TCC.2.6**

1760 The TSF ~~shall~~ allow [selection: *type of subject, type of user*] to select the security  
 1761 properties of the trusted channel.

1762 **8.4.6.8 — FCO\_TCC.2.7**

1763 The TSF ~~shall~~ allow [selection: *type of subject, type of user*] to modify the security  
 1764 properties of the trusted channel using the following rules [assignment: *rules that define*  
 1765 *the restrictions for the modification of the security properties of a trusted channel*]  
 1766

1767    **9    Class FCS: Cryptographic support**

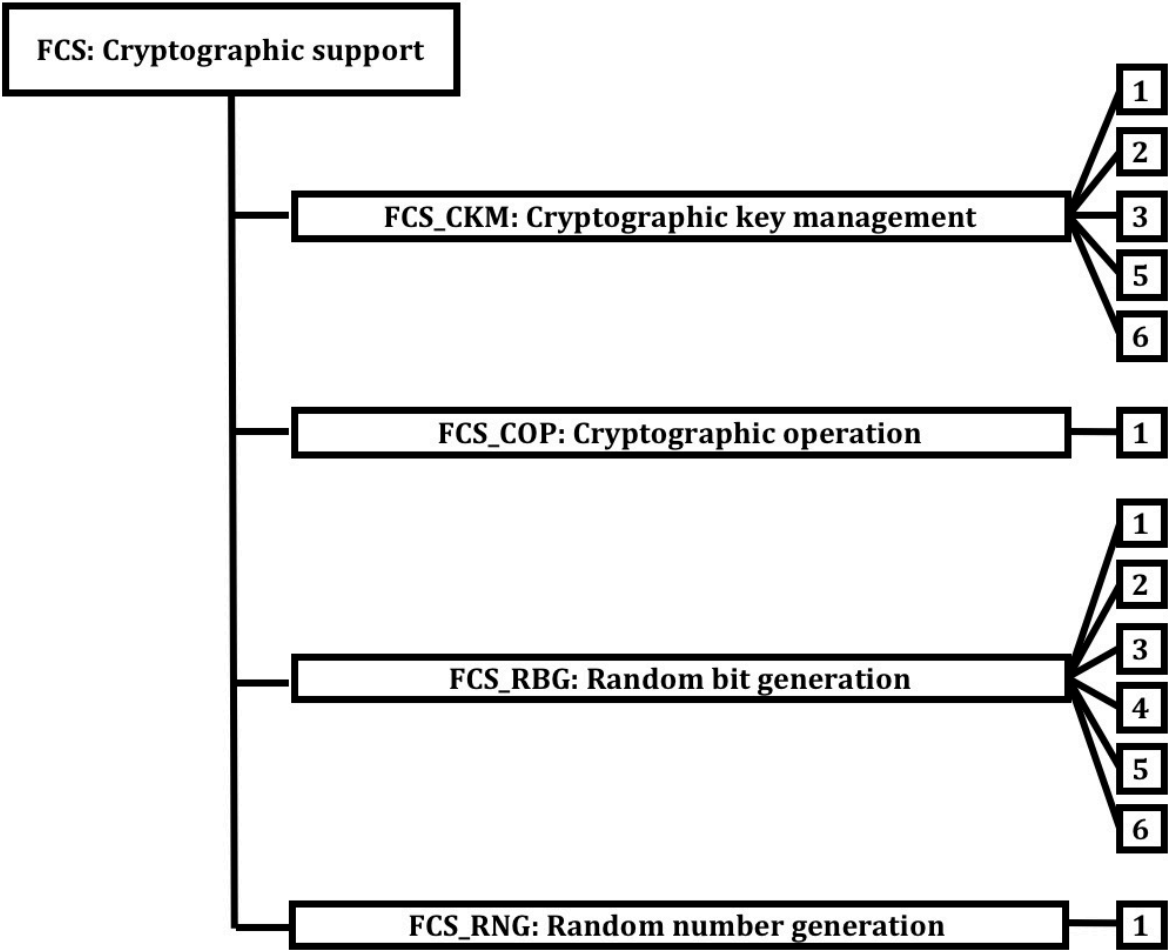
1768    **9.1 Class description**

1769    The TSF **may** employ cryptographic functionality to help satisfy several high-level security  
1770    objectives. These include (but are not limited to): identification and authentication, non-  
1771    repudiation, trusted path, trusted channel, and data separation. This class is used when the TOE  
1772    implements cryptographic functions, the implementation of which **could** be in hardware,  
1773    firmware and/or software.

1774    The FCS: Cryptographic support class is composed of four families.

1775    Figure 18 shows the decomposition of this class, it's families and components. Elements are not  
1776    shown in the figure.

1777    Annex E provides explanatory information for this class and **should** be consulted when using  
1778    the components identified in this class.



1779                   **Figure 18 — FCS: Cryptographic support class decomposition**

1780    **9.2 Cryptographic key management (FCS\_CKM)**

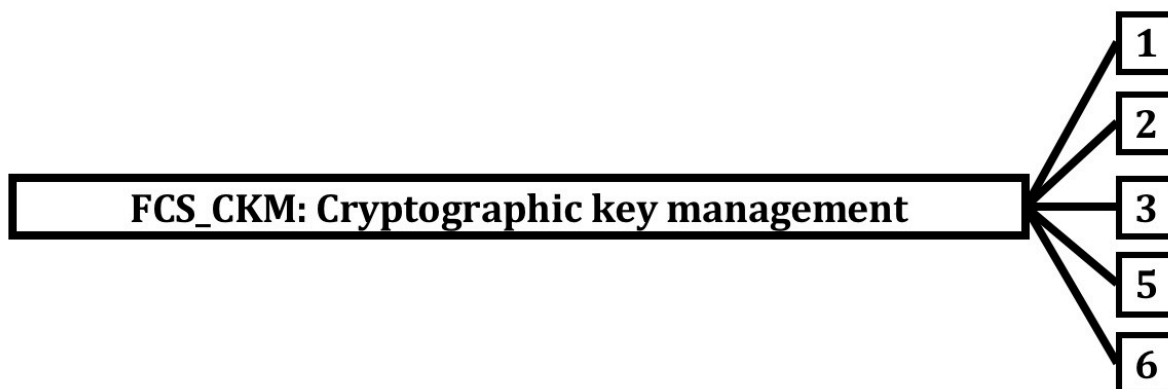
1781    **9.2.1 Family behaviour**

1782    Cryptographic keys must be managed throughout their life cycle. This family is intended to  
1783    support that lifecycle and consequently defines requirements for the following activities:  
1784    cryptographic key generation, cryptographic key derivation, cryptographic key distribution,  
1785    cryptographic key access and timing and event of cryptographic key destruction. This family

1786 **should** be included whenever there are functional requirements for the management of  
 1787 cryptographic keys.

## 1788 9.2.2 Components leveling and description

1789 Figure 19 shows the component leveling for this family.



1790 **Figure 19 — FCS\_CKM: Component leveling**

1791 FCS\_CKM.1 Cryptographic key generation, requires cryptographic keys to be generated in  
 1792 accordance with a specified algorithm and key sizes which **can** be based on an assigned  
 1793 standard.

1794 FCS\_CKM.2 Cryptographic key distribution, requires cryptographic keys to be distributed in  
 1795 accordance with a specified distribution method which **can** be based on an assigned standard.

1796 FCS\_CKM.3 Cryptographic key access requires access to cryptographic keys to be performed in  
 1797 accordance with a specified access method which **can** be based on an assigned standard.

1798 FCS\_CKM.5 Cryptographic key derivation, requires that the methods, standards, and parameters  
 1799 for key-derivation are specified.

1800 FCS\_CKM.6 Timing and event of cryptographic key destruction, requires cryptographic keys to  
 1801 be destroyed in accordance with specified destruction methods which **can** be based on an  
 1802 assigned standard.

1803 NOTE Previous editions of this standard specified FCS\_CKM.4 which has been deprecated in this edition. In  
 1804 order to preserve consistency between editions of this standard the component number has not been re-used.

## 1805 9.2.3 Management of FCS\_CKM.1, FCS\_CKM.2, FCS\_CKM.3, FCS\_CKM.5, CKM.6

1806 The following actions **could** be considered for the management functions in FMT:

1807 a) There are no management activities foreseen.

## 1808 9.2.4 Audit of FCS\_CKM.1, FCS\_CKM.2, FCS\_CKM.3, FCS\_CKM.5, CKM.6

1809 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 1810 in the PP/ST:

1811 a) Minimal: Success and failure of the activity.

1812 b) Basic: The object attribute(s), and object value(s) excluding any sensitive  
 1813 information

## 1814 9.2.5 FCS\_CKM.1 Cryptographic key generation

### 1815 9.2.5.1 Component relationships

1816 Hierarchical to: No other components.



1817 Dependencies: [FCS\_CKM.5 Cryptographic key derivation, or  
 1818 FCS\_COP.1 Cryptographic operation]  
 1819 FCS\_CKM.3 Cryptographic key access

#### 1820 9.2.5.2 FCS\_CKM.1.1

1821 The TSF **shall** generate cryptographic keys in accordance with a specified cryptographic  
 1822 key generation algorithm [assignment: *cryptographic key generation algorithm*] and  
 1823 specified cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the  
 1824 following: [assignment: *list of standards*].

### 1825 9.2.6 FCS\_CKM.2 Cryptographic key distribution

#### 1826 9.2.6.1 Component relationships

1827 Hierarchical to: No other components.  
 1828 Dependencies: [FDP\_ITC.1 Import of user data without security  
 1829 attributes, or  
 1830 FDP\_ITC.2 Import of user data with security  
 1831 attributes, or  
 1832 FCS\_CKM.1 Cryptographic key generation or  
 1833 FCS\_CKM.5 Cryptographic key derivation]  
 1834 FCS\_CKM.3 Cryptographic key access

#### 1835 9.2.6.2 FCS\_CKM.2.1

1836 The TSF **shall** distribute cryptographic keys in accordance with a specified cryptographic  
 1837 key distribution method [assignment: *cryptographic key distribution method*] that meets  
 1838 the following: [assignment: *list of standards*].

### 1839 9.2.7 FCS\_CKM.3 Cryptographic key access

#### 1840 9.2.7.1 Component relationships

1841 Hierarchical to: No other components.  
 1842 Dependencies: [FDP\_ITC.1 Import of user data without security  
 1843 attributes, or  
 1844 FDP\_ITC.2 Import of user data with security  
 1845 attributes, or  
 1846 FCS\_CKM.1 Cryptographic key generation or  
 1847 FCS\_CKM.5 Cryptographic key derivation]

#### 1848 9.2.7.2 FCS\_CKM.3.1

1849 The TSF **shall** perform [assignment: *type of cryptographic key access*] in accordance with  
 1850 a specified cryptographic key access method [assignment: *cryptographic key access*  
 1851 *method*] that meets the following: [assignment: *list of standards*].

### 1852 9.2.8 FCS\_CKM.4 Cryptographic key destruction

1853 The component has been deprecated. See FCS\_CKM.6 Timing and event of cryptographic key  
 1854 destruction instead.

1855 **Editors' Note**

The Editors' have taken the approach of deprecation in order to avoid conflicts and difficulties in the migration of one edition of the standard to the next. Taking this approach this could help reduce confusion during the transition.

## 9.2.9 FCS\_CKM.5 Cryptographic key derivation

### 9.2.9.1 Component relationships

Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.6 Timing and event of cryptographic key destruction

### 9.2.9.2 FCS\_CKM.5.1

The TSF **shall** derive cryptographic keys [assignment: *key type*] from [selection: *input parameters*] in accordance with a specified key derivation algorithm [selection: *key derivation algorithm*] and specified cryptographic key sizes [selection: *list of key sizes*] that meet the following [selection: *list of standards*].

NOTE See E.2.5.1. for information on using this component

## 9.2.10 FCS\_CKM.6 Timing and event of cryptographic key destruction

### 9.2.10.1 Component relationships

Hierarchical to:	No other components
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.5 Cryptographic key derivation

### 9.2.10.2 FCS\_CKM.6.1

The TSF **shall** destroy [assignment: *list of cryptographic keys (including keying material)*] when [selection: *no longer needed, [assignment: other circumstances for key or key material destruction]*].

### 9.2.10.3 FCS\_CKM.6.2

The TSF **shall** destroy cryptographic keys and keying material specified by FCS\_CKM.6.1 in accordance with a specified cryptographic key destruction method [assignment: *cryptographic key destruction method*] that meets the following: [assignment: *list of standards*].

## 9.3 Cryptographic operation (FCS\_COP)

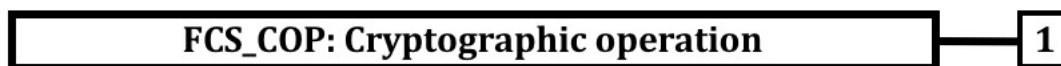
### 9.3.1 Family behaviour

In order for a cryptographic operation to function correctly, the operation must be performed in accordance with a specified algorithm and with a cryptographic key of a specified size. This family **should** be included whenever there are requirements for cryptographic operations to be performed.

1896 Typical cryptographic operations include data encryption and/or decryption, digital signature  
 1897 generation and/or verification, cryptographic checksum generation for integrity and/or  
 1898 verification of checksum, secure hash (message digest), cryptographic key encryption and/or  
 1899 decryption, and cryptographic key agreement.

### 1900 9.3.2 Components leveling and description

1901 Figure 20 shows the component leveling for this family.



1902 **Figure 20 — FCS\_COP: Component leveling**

1903 FCS\_COP.1 Cryptographic operation, requires a cryptographic operation to be performed in  
 1904 accordance with a specified algorithm and with a cryptographic key of specified sizes. The  
 1905 specified algorithm and cryptographic key sizes **can** be based on an assigned standard.

### 1906 9.3.3 Management of FCS\_COP.1

1907 The following actions **could** be considered for the management functions in FCS:

- 1908 a) There are no management activities foreseen.

### 1909 9.3.4 Audit of FCS\_COP.1

1910 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 1911 in the PP/ST:

- 1912 a) Minimal: Success and failure, and the type of cryptographic operation.  
 1913 b) Basic: Any applicable cryptographic mode(s) of operation, subject attributes and  
 1914 object attributes.

### 1915 9.3.5 FCS\_COP.1 Cryptographic operation

#### 1916 9.3.5.1 Component relationships

1917	Hierarchical to:	No other components.
1918	Dependencies:	[FDP_ITC.1 Import of user data without security
1919		attributes, or
1920		FDP_ITC.2 Import of user data with security
1921		attributes, or
1922		FCS_CKM.1 Cryptographic key generation]
1923		FCS_CKM.3 Cryptographic key access
1924		FCS_RBG.1 Random bit generation

#### 1925 9.3.5.2 FCS\_COP.1.1

1926 The TSF **shall** perform [assignment: *list of cryptographic operations*] in accordance with a  
 1927 specified cryptographic algorithm [assignment: *cryptographic algorithm*] and  
 1928 cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following:  
 1929 [assignment: *list of standards*].

## 1930 9.4 Random bit generation (FCS\_RBG)

### 1931 9.4.1 Family behaviour

1932 Components in this family address the requirements for random bit/number generation.

1933 **9.4.2 Components leveling and description**

1934 Figure 21 shows the component leveling for this family.



1935 **Figure 21 — FCS\_RBG: Component leveling**

1936 FCS\_RBG.1 Random bit generation (RBG) requires random bit generation to be performed in  
1937 accordance with selected standards.

1938 FCS\_RBG.2 Random bit generation (external seeding) gives requirements for seeding by an  
1939 external (outside the TOE) entropy source.

1940 FCS\_RBG.3 Random bit generation (internal seeding – single source) gives requirements for  
1941 seeding using a TSF entropy source.

1942 FCS\_RBG.4 Random bit generation (internal seeding – multiple sources) gives requirements for  
1943 seeding using multiple TSF entropy sources.

1944 FCS\_RBG.5 Random bit generation (combining entropy sources) gives requirements for  
1945 combining multiple entropy sources (multiple internal sources, internal and external).

1946 FCS\_RBG.6 Random bit generation service requires random numbers to be supplied over an  
1947 external interface as a service to other entities.

1948 **9.4.3 Management of FCS\_RBG.1, FCS\_RBG.2, FCS\_RBG.3, FCS\_RBG.4, FCS\_RBG.5,**  
1949 **FCS\_RBG.6**

1950 The following actions **could** be considered for the management functions in FMT:

1951 a) There are no management activities foreseen.

1952 **9.4.4 Audit of FCS\_RBG.1, FCS\_RBG.2**

1953 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1954 in the PP/ST:

1955 a) Minimal: failure of the randomization process, failure to initialize or reseed (as  
1956 supported by the technology)

1957 **9.4.5 Audit of FCS\_RBG.3, FCS\_RBG.4, FCS\_RBG.6, FCS\_RBG.6**

1958 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
1959 in the PP/ST:

1960 a) There are no auditable events foreseen.

1961 **9.4.6 FCS\_RBG.1 Random bit generation (RBG)**

1962 **9.4.6.1 Component relationships**

1963	Hierarchical to:	No other components
1964	Dependencies:	[FCS_RBG.2 Random bit generation (external seeding), FCS_RBG.3 Random bit generation (internal seeding – single source)]
1965		
1966		FPT_FLS.1 Failure with preservation of secure state
1967		FPT_TST.1 TSF self-testing
1968		

1969 **9.4.6.2 FCS\_RBG.1.1**

1970 The TSF **shall** perform deterministic random bit generation services using [assignment: *RBG algorithm*] in accordance with [assignment: *list of standards*] after initialization with  
 1971 a seed.  
 1972

1973 **9.4.6.3 FCS\_RBG.1.2**

1974 The TSF **shall** initialize and update the RBG state using a noise source as shown in the  
 1975 RBG State Update Table.

1976 **Table 1 – RBG State Update Table**

Identifier	Noise source	Update type	Condition	list of standards
Source1	[selection: TOE internal, external]	initialize	initialization	[assignment: list of standards]
[assignment: identifier]	[selection: TOE internal, external]	[selection: reseed, unstantiate+stantiate]	[selection: on demand; on the condition: [assignment: condition]; after [assignment: time]]	[assignment: list of standards]

1977

1978 **9.4.7 FCS\_RBG.2 Random bit generation (external seeding)**

1979 **9.4.7.1 Component relationships**

1980	Hierarchical to:	No other components.
1981	Dependencies:	FCS_RBG.1 Random bit generation (RBG)

1982 **9.4.7.2 FCS\_RBG.2.1**

1983 The TSF **shall** be able to accept a minimum input of [assignment: *minimum input length greater than zero*] from an external interface for the purpose of seed generation.  
 1984

1985 **9.4.8 FCS\_RBG.3 Random bit generation (internal seeding – single source)**

1986 **9.4.8.1 Component relationships**

1987	Hierarchical to:	No other components
1988	Dependencies:	FCS_RBG.1 Random bit generation (RBG)

1989	<b>9.4.8.2 FCS_RBG.3.1</b>	
1990	<b>The TSF shall be able to seed the RBG using a single [selection: <i>TSF software-based noise source, TSF hardware-based noise source</i>] with a minimum of [assignment: <i>number of bits</i>] bits of min-entropy.</b>	
1991		
1992		
1993	<b>9.4.9 FCS_RBG.4 Random bit generation (internal seeding – multiple sources)</b>	
1994	<b>9.4.9.1 Component relationships</b>	
1995	Hierarchical to:	No other components
1996	Dependencies:	FCS_RBG.1 Random bit generation (RBG)
1997		FCS_RBG.3 Random bit generation (internal seeding
1998		– single source)
1999	<b>9.4.9.2 FCS_RBG.4.1</b>	
2000	<b>The TSF shall be able to seed the RBG using [selection: [assignment: <i>number</i>] <i>TSF software-based noise source(s)</i>, [assignment: <i>number</i>] <i>TSF hardware-based noise source(s)</i>].</b>	
2001		
2002	<b>9.4.10 FCS_RBG.5 Random bit generation (combining entropy sources)</b>	
2003	<b>9.4.10.1 Component relationships</b>	
2004	Hierarchical to:	No other components.
2005	Dependencies:	FCS_RBG.1 Random bit generation (RBG)
2006		[FCS_RBG.2 Random bit generation (external
2007		seeding), or
2008		FCS_RBG.3 Random bit generation (internal seeding
2009		– single source)]
2010	<b>9.4.10.2 FCS_RBG.5.1 Combining entropy sources</b>	
2011	<b>The TSF shall [assignment: <i>combining operation</i>] [selection: <i>TSF entropy source(s)</i>, <i>TOE external entropy source(s)</i>] to create the entropy input into the derivation function as</b>	
2012	<b>defined in [assignment: <i>list of standards</i>], resulting in a minimum of [assignment:</b>	
2013	<b><i>number of bits</i>] bits of min-entropy.</b>	
2014		
2015	<b>9.4.11 FCS_RBG.6 Random bit generation service</b>	
2016	<b>9.4.11.1 Component relationships</b>	
2017	Hierarchical to:	No other components.
2018	Dependencies:	FCS_RBG.1 Random bit generation (RBG)
2019		[FCS_RBG.2 Random bit generation (external
2020		seeding), or
2021		FCS_RBG.3 Random bit generation (internal seeding
2022		– single source)]
2023	<b>9.4.11.2 FCS_RBG.6.1</b>	
2024	<b>The TSF shall provide a [selection: <i>hardware, software, [assignment: <i>other interface type</i>]]</i></b>	
2025	<b>interface to make the RBG output, as specified in FCS_RBG.1 Random bit generation</b>	
2026	<b>(RBG), available as a service to entities outside of the TOE.</b>	
2027	<b>9.5 Generation of random numbers (FCS_RNG)</b>	
2028	<b>Editors' Notes</b>	

2029 This SFR was proposed by WD 1 DE/DB17 (N1462).

### 2030 9.5.1 Family behaviour

2031 This family defines quality requirements for the generation of random numbers which are  
2032 intended to be use for cryptographic purposes.

### 2033 9.5.2 Components leveling and description

2034 Figure 22 shows the component leveling for this family.



2035 Figure 22 — FCS\_RNG: Component leveling

2036 FCS\_RNG.1 Random number generation requires that random numbers meet a defined quality  
2037 metric.

### 2038 9.5.3 Management of FCS\_RNG.1

2039 There are no management activities foreseen.

### 2040 9.5.4 Audit of FCS\_RNG.1

2041 There are no actions defined to be auditable.

### 2042 9.5.5 FCS\_RNG.1 Random number generation

#### 2043 9.5.5.1 Component relationships

2044 Hierarchical to: No other components.

2045 Dependencies: No dependencies.

#### 2046 9.5.5.2 FCS\_RNG.1.1

2047 The TSF **shall** provide a [selection: *physical, non-physical true, deterministic, hybrid*  
2048 *physical, hybrid deterministic*] random number generator that implements: [assignment:  
2049 *list of security capabilities*].

#### 2050 9.5.5.3 FCS\_RNG.1.2

2051 The TSF **shall** provide [selection: *bits, octets of bits, numbers* [assignment: *format of the*  
2052 *numbers*]] that meet [assignment: *a defined quality metric*].

2053

## 2054 **10 Class FDP: User data protection**

### 2055 **10.1 Class description**

2056 This class contains families specifying requirements related to protecting user data. FDP: User  
 2057 data protection is split into four groups of families (listed below) that address user data within  
 2058 a TOE, during import, export, and storage as well as security attributes directly related to user  
 2059 data.

2060 The families in this class are organized into four groups:

2061 a) User data protection security function policies:

2062 — Access control policy (FDP\_ACC); and

2063 — Information flow control policy (FDP\_IFC).

2064 Components in these families permit the PP/ST author to name the user data  
 2065 protection security function policies and define the scope of control of the policy,  
 2066 necessary to address the security objectives. The names of these policies are meant  
 2067 to be used throughout the remainder of the functional components that have an  
 2068 operation that calls for an assignment or selection of an "access control SFP" or an  
 2069 "information flow control SFP". The rules that define the functionality of the named  
 2070 access control and information flow control SFPs will be defined in the Access  
 2071 control functions (FDP\_ACF) and Information flow control functions (FDP\_IFF)  
 2072 families (respectively).

2073 b) Forms of user data protection:

2074 — Access control functions (FDP\_ACF);

2075 — Information flow control functions (FDP\_IFF);

2076 — Internal TOE transfer (FDP\_ITT);

2077 — Information Retention Control (FDP\_IRC)

2078 — Residual information protection (FDP\_RIP);

2079 — Rollback (FDP\_ROL);

2080 — Stored data confidentiality (FDP\_SDC); and

2081 — Stored data integrity (FDP\_SDI).

2082 c) Off-line storage, import and export:

2083 — Data authentication (FDP\_DAU);

2084 — Export from the TOE (FDP\_ETC);

2085 — Import from outside of the TOE (FDP\_ITC).

2086 Components in these families address the trustworthy transfer into or out of the  
 2087 TOE.

2088 d) Inter-TSF communication:

2089 — Inter-TSF user data confidentiality transfer protection (FDP\_UCT); and

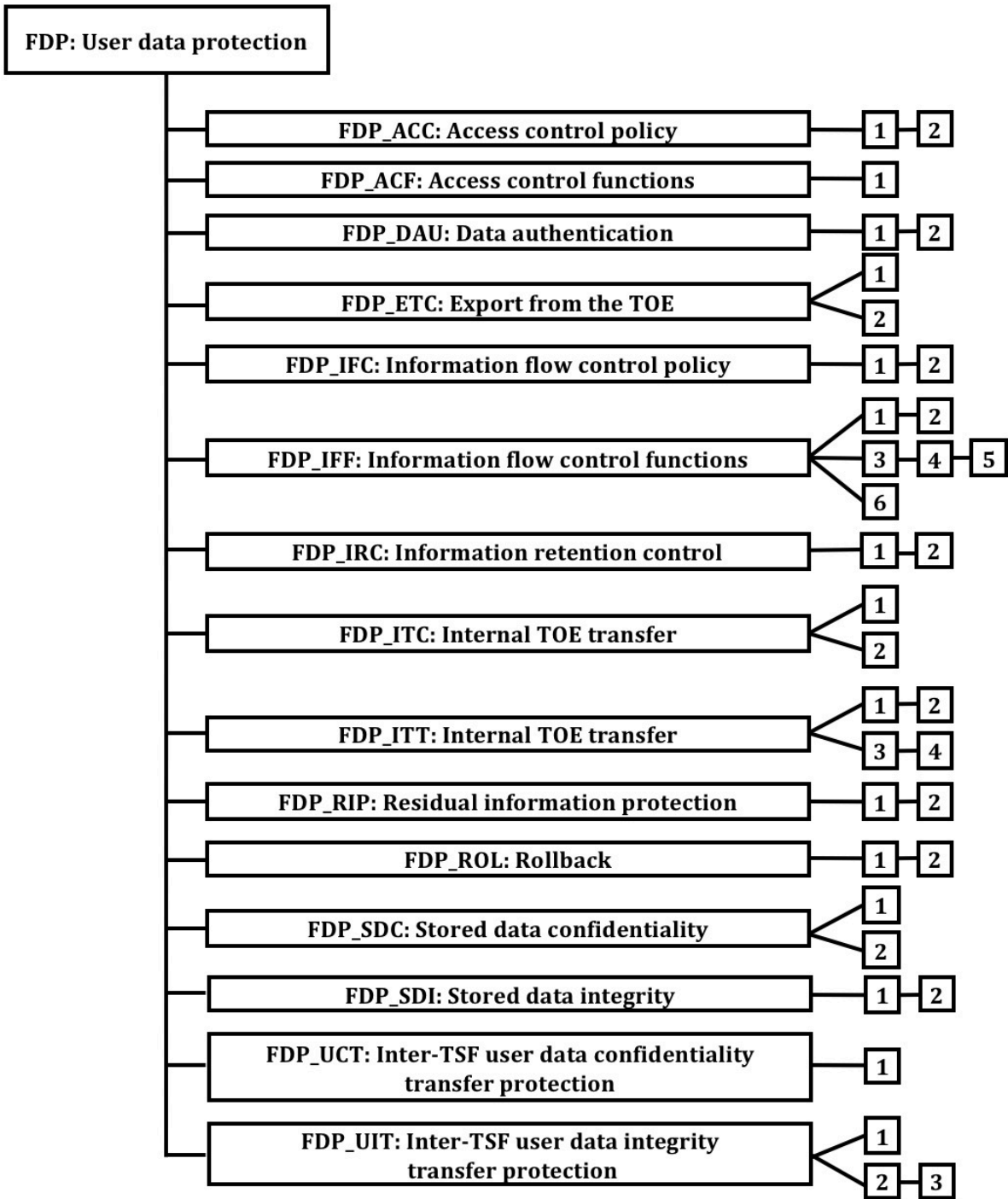
2090 — Inter-TSF user data integrity transfer protection (FDP\_UIT).

2091 — Components in these families address communication between the TSF of the  
 2092 TOE and another trusted IT product.

2093 Figure 23 shows the decomposition of this class, it's families and components. Elements are not  
 2094 shown in the figure.



2095 Annex F provides explanatory information for this class and **should** be consulted when using  
2096 the components identified in this class.



2097 **Figure 23 — FDP: User data protection class decomposition**

2098 **10.2 Access control policy (FDP\_ACC)**

2099 **10.2.1 Family behaviour**

2100 This family identifies the access control SFPs (by name) and defines the scope of control of the  
2101 policies that form the identified access control portion of the SFRs related to the SFP. This scope  
2102 of control is characterized by three sets: the subjects under control of the policy, the objects  
2103 under control of the policy, and the operations among controlled subjects and controlled

2104 objects that are covered by the policy. The criteria allow multiple policies to exist, each having a  
 2105 unique name. This is accomplished by iterating components from this family once for each  
 2106 named access control policy. The rules that define the functionality of an access control SFP will  
 2107 be defined by other families such as Access control functions (FDP\_ACF) and Export from the  
 2108 TOE (FDP\_ETC). The names of the access control SFPs identified here in Access control policy  
 2109 (FDP\_ACC) are meant to be used throughout the remainder of the functional components that  
 2110 have an operation that calls for an assignment or selection of an “access control SFP.”

## 2111 10.2.2 Components leveling and description

2112 Figure 24 shows the component leveling for this family.



2113 **Figure 24 — FDP\_ACC: Component leveling**

2114 FDP\_ACC.1 Subset access control, requires that each identified access control SFP be in place for  
 2115 a subset of the possible operations on a subset of the objects in the TOE.

2116 FDP\_ACC.2 Complete access control, requires that each identified access control SFP cover all  
 2117 operations on subjects and objects covered by that SFP. It further requires that all objects and  
 2118 operations protected by the TSF are covered by at least one identified access control SFP.

## 2119 10.2.3 Management of FDP\_ACC.1, FDP\_ACC.2

2120 The following actions **could** be considered for the management functions in FMT:

- 2121 a) There are no management activities foreseen.

## 2122 10.2.4 Audit of FDP\_ACC.1, FDP\_ACC.2

2123 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 2124 in the PP/ST:

- 2125 a) There are no auditable events foreseen.

## 2126 10.2.5 FDP\_ACC.1 Subset access control

### 2127 10.2.5.1 Component relationships

2128 Hierarchical to: No other components.

2129 Dependencies: FDP\_ACF.1 Security attribute-based access control

### 2130 10.2.5.2 FDP\_ACC.1.1

2131 The TSF **shall** enforce the [assignment: *access control SFP*] on [assignment: *list of subjects,*  
 2132 *objects, and operations among subjects and objects covered by the SFP*].

## 2133 10.2.6 FDP\_ACC.2 Complete access control

### 2134 10.2.6.1 Component relationships

2135 Hierarchical to: FDP\_ACC.1 Subset access control

2136 Dependencies: FDP\_ACF.1 Security attribute-based access control

### 2137 10.2.6.2 FDP\_ACC.2.1

2138 The TSF **shall** enforce the [assignment: *access control SFP*] on [assignment: *list of subjects and*  
 2139 *objects*] and all operations among subjects and objects covered by the **SFP**.

### 10.2.6.3 FDP\_ACC.2.2

The TSF **shall** ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.

## 10.3 Access control functions (FDP\_ACF)

### 10.3.1 Family behaviour

This family describes the rules for the specific functions that **can** implement an access control policy named in Access control policy (FDP\_ACC). Access control policy (FDP\_ACC) specifies the scope of control of the policy.

### 10.3.2 Components leveling and description

Figure 25 shows the component leveling for this family.



**Figure 25 — FDP\_ACF: Component leveling**

This family addresses security attribute usage and characteristics of policies. The component within this family is meant to be used to describe the rules for the function that implements the SFP as identified in Access control policy (FDP\_ACC). The PP/ST author **may** also iterate this component to address multiple policies in the TOE.

FDP\_ACF.1 Security attribute-based access control Security attribute-based access control allows the TSF to enforce access based upon security attributes and named groups of attributes. Furthermore, the TSF **may** have the ability to explicitly authorize or deny access to an object based upon security attributes.

### 10.3.3 Management of FDP\_ACF.1

The following actions **could** be considered for the management functions in FMT:

- a) Managing the attributes used to make explicit access or denial-based decisions.

### 10.3.4 Audit of FDP\_ACF.1

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: Successful requests to perform an operation on an object covered by the SFP.
- b) Basic: All requests to perform an operation on an object covered by the SFP.
- c) Detailed: The specific security attributes used in making an access check.

## 10.3.5 FDP\_ACF.1 Security attribute-based access control

### 10.3.5.1 Component relationships

Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control
	FMT_MSA.3 Static attribute

### 10.3.5.2 FDP\_ACF.1.1

The TSF **shall** enforce the [assignment: *access control SFP*] to objects based on the following: [assignment: *list of subjects and objects controlled under the indicated SFP, and*

2177 *for each, the SFP-relevant security attributes, or named groups of SFP-relevant security*  
 2178 *attributes].*

#### 2179 10.3.5.3 FDP\_ACF.1.2

2180 The TSF **shall** enforce the following rules to determine if an operation among controlled  
 2181 subjects and controlled objects is allowed: [assignment: *rules governing access among*  
 2182 *controlled subjects and controlled objects using controlled operations on controlled*  
 2183 *objects].*

#### 2184 10.3.5.4 FDP\_ACF.1.3

2185 The TSF **shall** explicitly authorize access of subjects to objects based on the following  
 2186 additional rules: [assignment: *rules, based on security attributes, that explicitly authorize*  
 2187 *access of subjects to objects].*

#### 2188 10.3.5.5 FDP\_ACF.1.4

2189 The TSF **shall** explicitly deny access of subjects to objects based on the following  
 2190 additional rules: [assignment: *rules, based on security attributes, that explicitly deny*  
 2191 *access of subjects to objects].*

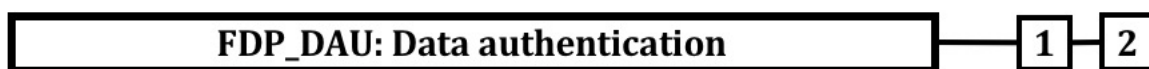
### 2192 10.4 Data authentication (FDP\_DAU)

#### 2193 10.4.1 Family behaviour

2194 Data authentication permits an entity to accept responsibility for the authenticity of  
 2195 information. This family provides a method of providing a guarantee of the validity of a specific  
 2196 unit of data that **can** be subsequently used to verify that the information content has not been  
 2197 forged or fraudulently modified. In contrast to FAU: Security audit, this family is intended to be  
 2198 applied to "static" data rather than data that is being transferred.

#### 2199 10.4.2 Components leveling and description

2200 Figure 26 shows the component leveling for this family.



2201 **Figure 26 — FDP\_DAU: Component leveling**

2202 FDP\_DAU.1 Basic Data Authentication, requires that the TSF is capable of generating a  
 2203 guarantee of authenticity of the information content of objects.

2204 FDP\_DAU.2 Data Authentication with Identity of Guarantor additionally requires that the TSF is  
 2205 capable of establishing the identity of the subject who provided the guarantee of authenticity.

#### 2206 10.4.3 Management of FDP\_DAU.1, FDP\_DAU.2

2207 The following actions **could** be considered for the management functions in FMT:

- 2208 a) The assignment or modification of the objects for which data authentication **may**  
 2209 apply **could** be configurable.

#### 2210 10.4.4 Audit of FDP\_DAU.1

2211 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 2212 in the PP/ST:

- 2213 a) Minimal: Successful generation of validity evidence.
- 2214 b) Basic: Unsuccessful generation of validity evidence.

2215 c) Detailed: The identity of the subject that requested the evidence.

#### 2216 10.4.5 Audit of FDP\_DAU.2

2217 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
2218 in the PP/ST:

2219 a) Minimal: Successful generation of validity evidence.

2220 b) Basic: Unsuccessful generation of validity evidence.

2221 c) Detailed: The identity of the subject that requested the evidence.

2222 d) Detailed: The identity of the subject that generated the evidence.

#### 2223 10.4.6 FDP\_DAU.1 Basic Data Authentication

##### 2224 10.4.6.1 Component relationships

2225 Hierarchical to: No other components.

2226 Dependencies: No dependencies.

##### 2227 10.4.6.2 FDP\_DAU.1.1

2228 The TSF **shall** provide a capability to generate evidence that **can** be used as a guarantee of  
2229 the validity of [assignment: *list of objects or information types*].

##### 2230 10.4.6.3 FDP\_DAU.1.2

2231 The TSF **shall** provide [assignment: *list of subjects*] with the ability to verify evidence of  
2232 the validity of the indicated information.

#### 2233 10.4.7 FDP\_DAU.2 Data Authentication with Identity of Guarantor

##### 2234 10.4.7.1 Component relationships

2235 Hierarchical to: FDP\_DAU.1 Basic Data Authentication

2236 Dependencies: FIA\_UID.1 Timing of identification

##### 2237 10.4.7.2 FDP\_DAU.2.1

2238 The TSF **shall** provide a capability to generate evidence that **can** be used as a guarantee of the  
2239 validity of [assignment: *list of objects or information types*].

##### 2240 10.4.7.3 FDP\_DAU.2.2

2241 The TSF **shall** provide [assignment: *list of subjects*] with the ability to verify evidence of the  
2242 validity of the indicated information **and the identity of the user that generated the**  
2243 **evidence.**

#### 2244 10.5 Export from the TOE (FDP\_ETC)

##### 2245 10.5.1 Family behaviour

2246 This family defines functions for TSF-mediated exporting of user data from the TOE such that its  
2247 security attributes and protection either **can** be explicitly preserved or **can** be ignored once it  
2248 has been exported. It is concerned with limitations on export and with the association of  
2249 security attributes with the exported user data.

2250 **10.5.2 Components leveling and description**

2251 Figure 27 shows the component leveling for this family.

2252 **Figure 27 — FDP\_ETC: Component leveling**

2253 FDP\_ETC.1 Export of user data without security attributes, requires that the TSF enforces the  
 2254 appropriate SFPs when exporting user data outside the TSF. User data that is exported by this  
 2255 function is exported without its associated security attributes.

2256 FDP\_ETC.2 Export of user data with security attributes, requires that the TSF enforces the  
 2257 appropriate SFPs using a function that accurately and unambiguously associates security  
 2258 attributes with the user data that is exported.

2259 **10.5.3 Management of FDP\_ETC.1**2260 The following actions **could** be considered for the management functions in FMT:

- 2261 a) There are no management activities foreseen.

2262 **10.5.4 Management of FDP\_ETC.2**2263 The following actions **could** be considered for the management functions in FMT:

- 2264 a) The additional exportation control rules **could** be configurable by a user in a  
 2265 defined role.

2266 **10.5.5 Audit of FDP\_ETC.1, FDP\_ETC.2**

2267 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 2268 in the PP/ST:

- 2269 a) Minimal: Successful export of information.  
 2270 b) Basic: All attempts to export information.

2271 **10.5.6 FDP\_ETC.1 Export of user data without security attributes**2272 **10.5.6.1 Component relationships**

2273	Hierarchical to:	No other components.
2274	Dependencies:	[FDP_ACC.1 Subset access control, or
2275		FDP_IFC.1 Subset information flow control]

2276 **10.5.6.2 FDP\_ETC.1.1**

2277 The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow*  
 2278 *control SFP(s)*] when exporting user data, controlled under the SFP(s), outside of the TOE.

2279 **10.5.6.3 FDP\_ETC.1.2**2280 The TSF **shall** export the user data without the user data's associated security attributes.2281 **10.5.7 FDP\_ETC.2 Export of user data with security attributes**2282 **10.5.7.1 Component relationships**

2283	Hierarchical to:	No other components.
------	------------------	----------------------



- 2284 Dependencies: [FDP\_ACC.1 Subset access control, or  
2285 FDP\_IFC.1 Subset information flow control]
- 2286 **10.5.7.2 FDP\_ETC.2.1**
- 2287 The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow*  
2288 *control SFP(s)*] when exporting user data, controlled under the SFP(s), outside of the TOE.
- 2289 **10.5.7.3 FDP\_ETC.2.2**
- 2290 The TSF **shall** export the user data with the user data's associated security attributes.
- 2291 **10.5.7.4 FDP\_ETC.2.3**
- 2292 The TSF **shall** ensure that the security attributes, when exported outside the TOE, are  
2293 unambiguously associated with the exported user data.
- 2294 **10.5.7.5 FDP\_ETC.2.4**
- 2295 The TSF **shall** enforce the following rules when user data is exported from the TOE:  
2296 [assignment: *additional exportation control rules*].
- 2297 **10.6 Information flow control policy (FDP\_IFC)**
- 2298 **10.6.1 Family behaviour**
- 2299 This family identifies the information flow control SFPs (by name) and defines the scope of  
2300 control for each named information flow control SFP. This scope of control is characterized by  
2301 three sets: the subjects under control of the policy, the information under control of the policy,  
2302 and operations which cause controlled information to flow to and from controlled subjects  
2303 covered by the policy. The criteria allow multiple policies to exist, each having a unique name.  
2304 This is accomplished by iterating components from this family once for each named information  
2305 flow control policy. The rules that define the functionality of an information flow control SFP  
2306 will be defined by other families such as Information flow control functions (FDP\_IFF) and  
2307 Export from the TOE (FDP\_ETC). The names of the information flow control SFPs identified here  
2308 in Information flow control policy (FDP\_IFC) are meant to be used throughout the remainder of  
2309 the functional components that have an operation that calls for an assignment or selection of an  
2310 "information flow control SFP."
- 2311 The TSF mechanism controls the flow of information in accordance with the information flow  
2312 control SFP. Operations that would change the security attributes of information are not  
2313 generally permitted as this would be in violation of an information flow control SFP. However,  
2314 such operations **may** be permitted as exceptions to the information flow control SFP if explicitly  
2315 specified.
- 2316 **10.6.2 Components leveling and description**
- 2317 Figure 28 shows the component leveling for this family.



2318 **Figure 28 — FDP\_IFC: Component leveling**

- 2319 FDP\_IFC.1 Subset information flow control, requires that each identified information flow  
2320 control SFPs be in place for a subset of the possible operations on a subset of information flows  
2321 in the TOE.
- 2322 FDP\_IFC.2 Complete information flow control, requires that each identified information flow  
2323 control SFP cover all operations on subjects and information covered by that SFP. It further

2324 requires that all information flows and operations controlled by the TSF are covered by at least  
2325 one identified information flow control SFP.

### 2326 **10.6.3 Management of FDP\_IFC.1, FDP\_IFC.2**

2327 The following actions **could** be considered for the management functions in FMT:

2328 a) There are no management activities foreseen.

### 2329 **10.6.4 Audit of FDP\_IFC.1, FDP\_IFC.2**

2330 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
2331 in the PP/ST:

2332 a) There are no auditable events foreseen.

### 2333 **10.6.5 FDP\_IFC.1 Subset information flow control**

#### 2334 **10.6.5.1 Component relationships**

2335 Hierarchical to: No other components.

2336 Dependencies: FDP\_IFF.1 Simple security attributes

#### 2337 **10.6.5.2 FDP\_IFC.1.1**

2338 The TSF **shall** enforce the [assignment: *information flow control SFP*] on [assignment: *list*  
2339 *of subjects, information, and operations that cause controlled information to flow to and*  
2340 *from controlled subjects covered by the SFP*].

### 2341 **10.6.6 FDP\_IFC.2 Complete information flow control**

#### 2342 **10.6.6.1 Component relationships**

2343 Hierarchical to: FDP\_IFC.1 Subset information flow control

2344 Dependencies: FDP\_IFF.1 Simple security attributes

#### 2345 **10.6.6.2 FDP\_IFC.2.1**

2346 The TSF **shall** enforce the [assignment: *information flow control SFP*] on [assignment: *list of*  
2347 *subjects and information*] **and all** operations that cause **that** information to flow to and from  
2348 subjects covered by the **SFP**.

#### 2349 **10.6.6.3 FDP\_IFC.2.2**

2350 The TSF **shall** ensure that all operations that cause any information in the TOE to flow to  
2351 and from any subject in the TOE are covered by an information flow control SFP.

## 2352 **10.7 Information flow control functions (FDP\_IFF)**

### 2353 **10.7.1 Family behaviour**

2354 This family describes the rules for the specific functions that **can** implement the information  
2355 flow control SFPs named in Information flow control policy (FDP\_IFC), which also specifies the  
2356 scope of control of the policy. It consists of two kinds of requirements: one addressing the  
2357 common information flow function issues, and a second addressing illicit information flows (i.e.  
2358 covert channels). This division arises because the issues concerning illicit information flows are,  
2359 in some sense, orthogonal to the rest of an information flow control SFP. By their nature, they  
2360 circumvent the information flow control SFP resulting in a violation of the policy. As such, they  
2361 require special functions to either limit or prevent their occurrence.



## 10.7.2 Components leveling and description

Figure 29 shows the component leveling for this family.



**Figure 29 — FDP\_IFF: Component leveling**

FDP\_IFF.1 Simple security attributes, requires security attributes on information, and on subjects that cause that information to flow and on subjects that act as recipients of that information. It specifies the rules that must be enforced by the function and describes how security attributes are derived by the function.

FDP\_IFF.2 Hierarchical security attributes expands on the requirements of FDP\_IFF.1 Simple security attributes by requiring that all information flow control SFPs in the set of SFRs use hierarchical security attributes that form a lattice (as defined in mathematics). FDP\_IFF.2.6 is derived from the mathematical properties of a lattice. A lattice consists of a set of elements with an ordering relationship with the property defined in the first bullet, a least upper bound which is the unique element in the set that is greater or equal (in the ordering relationship) than any other element of the lattice, and a greatest lower bound, which is the unique element in the set that is smaller or equal than any other element of the lattice.

FDP\_IFF.3 Limited illicit information flows, requires the SFP to cover illicit information flows, but not necessarily eliminate them.

FDP\_IFF.4 Partial elimination of illicit information flows, requires the SFP to cover the elimination of some (but not necessarily all) illicit information flows.

FDP\_IFF.5 No illicit information flows, requires SFP to cover the elimination of all illicit information flows.

FDP\_IFF.6 Illicit information flow monitoring, requires the SFP to monitor illicit information flows for specified and maximum capacities.

## 10.7.3 Management of FDP\_IFF.1, FDP\_IFF.2

The following actions **could** be considered for the management functions in FMT:

- a) Managing the attributes used to make explicit access-based decisions.

## 10.7.4 Management of FDP\_IFF.3, FDP\_IFF.4, FDP\_IFF.5

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

## 10.7.5 Management of FDP\_IFF.6

The following actions **could** be considered for the management functions in FMT:

- a) The enabling or disabling of the monitoring function.
- b) Modification of the maximum capacity at which the monitoring occurs.

## 10.7.6 Audit of FDP\_IFF.1, FDP\_IFF.2, FDP\_IFF.5

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: Decisions to permit requested information flows.

- 2399           b) Basic: All decisions on requests for information flow.
- 2400           c) Detailed: The specific security attributes used in making an information flow
- 2401           enforcement decision.
- 2402           d) Detailed: Some specific subsets of the information that has flowed based upon
- 2403           policy goals.
- 2404   **10.7.7 Audit of FDP\_IFF.3, FDP\_IFF.4, FDP\_IFF.6**
- 2405   The following actions **should** be auditable if FAU\_GEN Security audit data generation is included
- 2406   in the PP/ST:
- 2407           a) Minimal: Decisions to permit requested information flows;
- 2408           b) Basic: All decisions on requests for information flow;
- 2409           c) Basic: The use of identified illicit information flow channels;
- 2410           d) Detailed: The specific security attributes used in making an information flow
- 2411           enforcement decision;
- 2412           e) Detailed: Some specific subsets of the information that has flowed based upon
- 2413           policy goals;
- 2414           f) Detailed: The use of identified illicit information flow channels with estimated
- 2415           maximum capacity exceeding a specified value.
- 2416   **10.7.8 FDP\_IFF.1 Simple security attributes**
- 2417   **10.7.8.1 Component relationships**
- 2418           Hierarchical to:                   No other components.
- 2419           Dependencies:                   FDP\_IFC.1 Subset information flow control
- 2420   FMT\_MSA.3 Static attribute
- 2421   **10.7.8.2 FDP\_IFF.1.1**
- 2422   The TSF **shall** enforce the [assignment: *information flow control SFP*] based on the
- 2423   following types of subject and information security attributes: [assignment: *list of*
- 2424   *subjects and information controlled under the indicated SFP, and for each, the security*
- 2425   *attributes*].
- 2426   **10.7.8.3 FDP\_IFF.1.2**
- 2427   The TSF **shall** permit an information flow between a controlled subject and controlled
- 2428   information via a controlled operation if the following rules hold: [assignment: *for each*
- 2429   *operation, the security attribute-based relationship that must hold between subject and*
- 2430   *information security attributes*].
- 2431   **10.7.8.4 FDP\_IFF.1.3**
- 2432   The TSF **shall** enforce the [assignment: *additional information flow control SFP rules*].
- 2433   **10.7.8.5 FDP\_IFF.1.4**
- 2434   The TSF **shall** explicitly authorize an information flow based on the following rules:
- 2435   [assignment: *rules, based on security attributes, that explicitly authorize information*
- 2436   *flows*].

2437 **10.7.8.6 FDP\_IFF.1.5**

2438 The TSF **shall** explicitly deny an information flow based on the following rules:  
 2439 [assignment: *rules, based on security attributes, that explicitly deny information flows*].

2440 **10.7.9 FDP\_IFF.2 Hierarchical security attributes**2441 **10.7.9.1 Component relationships**

2442 Hierarchical to: FDP\_IFF.1 Simple security attributes  
 2443 Dependencies: FDP\_IFC.1 Subset information flow control  
 2444 FMT\_MSA.3 Static attribute

2445 **10.7.9.2 FDP\_IFF.2.1**

2446 The TSF **shall** enforce the [assignment: *information flow control SFP*] based on the following  
 2447 types of subject and information security attributes: [assignment: *list of subjects and*  
 2448 *information controlled under the indicated SFP, and for each, the security attributes*].

2449 **10.7.9.3 FDP\_IFF.2.2**

2450 The TSF **shall** permit an information flow between a controlled subject and controlled  
 2451 information via a controlled operation if the following rules, **based on the ordering**  
 2452 **relationships between security attributes** hold: [assignment: *for each operation, the security*  
 2453 *attribute-based relationship that must hold between subject and information security attributes*].

2454 **10.7.9.4 FDP\_IFF.2.3**

2455 The TSF **shall** enforce the [assignment: *additional information flow control SFP rules*].

2456 **10.7.9.5 FDP\_IFF.2.4**

2457 The TSF **shall** explicitly authorize an information flow based on the following rules:  
 2458 [assignment: *rules, based on security attributes, that explicitly authorize information flows*].

2459 **10.7.9.6 FDP\_IFF.2.5**

2460 The TSF **shall** explicitly deny an information flow based on the following rules: [assignment:  
 2461 *rules, based on security attributes, that explicitly deny information flows*].

2462 **10.7.9.7 FDP\_IFF.2.6**

2463 The TSF **shall** enforce the following relationships for any two valid information flow  
 2464 control security attributes:

- 2465 a) There exists an ordering function that, given two valid security attributes,  
 2466 determines if the security attributes are equal, if one security attribute is  
 2467 greater than the other, or if the security attributes are incomparable; and
- 2468 b) There exists a “least upper bound” in the set of security attributes, such that,  
 2469 given any two valid security attributes, there is a valid security attribute that  
 2470 is greater than or equal to the two valid security attributes; and
- 2471 c) There exists a “greatest lower bound” in the set of security attributes, such  
 2472 that, given any two valid security attributes, there is a valid security attribute  
 2473 that is not greater than the two valid security attributes.

2474 **10.7.10 FDP\_IFF.3 Limited illicit information flows**2475 **10.7.10.1 Component relationships**

2476 Hierarchical to: No other components.

2477 Dependencies: FDP\_IFC.1 Subset information flow control

2478 **10.7.10.2 FDP\_IFF.3.1**

2479 The TSF **shall** enforce the [assignment: *information flow control SFP*] to limit the capacity  
 2480 of [assignment: *types of illicit information flows*] to a [assignment: *maximum capacity*].

2481 **10.7.11 FDP\_IFF.4 Partial elimination of illicit information flows**2482 **10.7.11.1 Component relationships**

2483 Hierarchical to: FDP\_IFF.3 Limited illicit information flows

2484 Dependencies: FDP\_IFC.1 Subset information flow control

2485 **10.7.11.2 FDP\_IFF.4.1**

2486 The TSF **shall** enforce the [assignment: *information flow control SFP*] to limit the capacity of  
 2487 [assignment: *types of illicit information flows*] to a [assignment: *maximum capacity*].

2488 **10.7.11.3 FDP\_IFF.4.2**2489 The TSF **shall** prevent [assignment: *types of illicit information flows*].2490 **10.7.12 FDP\_IFF.5 No illicit information flows**2491 **10.7.12.1 Component relationships**

2492 Hierarchical to: FDP\_IFF.4 Partial elimination of illicit information  
 2493 flows

2494 Dependencies: FDP\_IFC.1 Subset information flow control

2495 **10.7.12.2 FDP\_IFF.5.1**

2496 The TSF **shall** ensure that **no illicit information flows exist to circumvent** [assignment:  
 2497 *name of information flow control SFP*].

2498 **10.7.13 FDP\_IFF.6 Illicit information flow monitoring**2499 **10.7.13.1 Component relationships**

2500 Hierarchical to: No other components.

2501 Dependencies: FDP\_IFC.1 Subset information flow control

2502 **10.7.13.2 FDP\_IFF.6.1**

2503 The TSF **shall** enforce the [assignment: *information flow control SFP*] to monitor  
 2504 [assignment: *types of illicit information flows*] when it exceeds the [assignment: *maximum*  
 2505 *capacity*].

## 10.8 Information Retention Control (FDP\_IRC)

### 10.8.1 Family behaviour

The “Information retention control” family addresses a basic need in secure information processing and storage applications for secure management of data no more needed by the TOE to perform its operation, but still stored in the TOE.,~~which however appears not to be covered by ISO/IEC 15408(all parts).~~

Editors’ note

“appears not to be covered”? Either it is or it isn’t covered.

Editors propose to delete this statement.

If no comments are received on this, the editor’s proposal will be accepted and presented in the next draft.

The traditional view of IT systems as data storage systems induced naturally into thinking that once entered, data would be seldom deleted from the system, and if so, mainly because of storage exhaustion problems.

But in a multilateral or high security environment it is important to minimize the replication, and temporal time frame in which information is contained in the system. Also, users might want their IT products to avoid retaining data that they consider exploitable by third parties or threatening their privacy. In this case, such a requirement **can** help users to gain confidence that the product is secure, as far as it deletes every copy of the data when not needed anymore.

The FDP\_RIP “Residual information protection” family addresses one side of this problem, but an explicit requirement on the management of no longer needed data is missing.

Of course, competing requirements **may** arise, as data **may** be needed by the system for more activities over a long period of time. Possible solutions to this problem are:

- Better protecting the information objects stored in the TOE from access,
- Re-requesting the protected information from the user each time it is needed.

Information retention control ensures, that information no longer necessary for the operation of the TOE is deleted by the TOE. Components of this family require the PP author to identify TOE activities and objects required for those activities, and not to be kept in the TOE, and the TOE to keep track of such stored objects, and to delete on-line and off-line copies of unnecessary information objects.

This family sets only requirements on information objects requested for specific activities in the TOE operation, and not on general data gathering. The policies which control the collection, storage, processing, disclosure, and elimination of general user data stored on the TOE must be detailed elsewhere, and are domain of the environmental objectives and organizational policies, not of the PP.

When more than one activity requires the presence of a protected object, all activities, which refer to the required object **shall** end before deleting it.

### 10.8.2 Components leveling and description

Figure 30 shows the component leveling for this family.

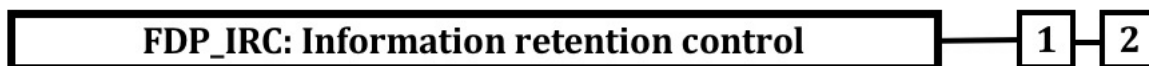


Figure 30 — FDP\_IRC: Component leveling

2546 FDP\_IRC.1 Subset information control requires that the TSF ensure that any copy of a defined  
 2547 subset of objects in the TSC is deleted when no longer strictly necessary for the operation of the  
 2548 TOE, and to identify and define the activities for which the object is required.

2549 FDP\_IRC.2 Complete information control requires them same but regarding to all objects in the  
 2550 TSC.

2551 **Editors' Note:**

2552 **Do we need the term "TSC" here? It seems this abbreviation has not been used since CC3.1 R3**

### 2553 **10.8.3 Management of FDP\_IRC.1**

2554 The following actions **could** be considered for the management functions in FMT:

2555 a) There are no management actions foreseen.

### 2556 **10.8.4 Audit of FDP\_IRC.1**

2557 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 2558 in the PP/ST:

2559 a) There are no auditable events foreseen.

### 2560 **10.8.5 FDP\_IRC.1 Subset information control**

#### 2561 **10.8.5.1 Component relationships**

2562 Hierarchical to: No other components.

2563 Dependencies: TBD.

#### 2564 **10.8.5.2 FDP\_IRC.1.1**

2565 The TSF **shall** ensure that [assignment: *list of objects*] required for [assignment: *list of*  
 2566 *activities*] **shall** be eliminated immediately from the TOE upon termination of the  
 2567 activities for which they are required.

### 2568 **10.8.6 FDP\_IRC.2 Complete information control**

#### 2569 **10.8.6.1 Component relationships**

2570 Hierarchical to: FDP\_IRC.1 Subset information control.

2571 Dependencies: TBD.

#### 2572 **10.8.6.2 FDP\_IRC.2.1**

2573 The TSF **shall** ensure that **all** objects required for [assignment: *list of activities*] **shall** be  
 2574 eliminated immediately from the TOE upon termination of the activities for which they are  
 2575 required.

## 2576 **10.9 Import from outside of the TOE (FDP\_ITC)**

### 2577 **10.9.1 Family behaviour**

2578 This family defines the mechanisms for TSF-mediated importing of user data into the TOE such  
 2579 that it has appropriate security attributes and is appropriately protected. It is concerned with  
 2580 limitations on importation, determination of desired security attributes, and interpretation of  
 2581 security attributes associated with the user data.

## 10.9.2 Components leveling and description

Figure 31 shows the component leveling for this family.



**Figure 31 — FDP\_ITC: Component leveling**

FDP\_ITC.1 Import of user data without security attributes, requires that the security attributes correctly represent the user data and are supplied separately from the object.

FDP\_ITC.2 Import of user data with security attributes, requires that security attributes correctly represent the user data and are accurately and unambiguously associated with the user data imported from outside the TOE.

## 10.9.3 Management of FDP\_ITC.1, FDP\_ITC.2

The following actions **could** be considered for the management functions in FMT:

- a) The modification of the additional control rules used for import.

## 10.9.4 Audit of FDP\_ITC.1, FDP\_ITC.2

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: Successful import of user data, including any security attributes.
- b) Basic: All attempts to import user data, including any security attributes.
- c) Detailed: The specification of security attributes for imported user data supplied by an authorized user.

## 10.9.5 FDP\_ITC.1 Import of user data without security attributes

### 10.9.5.1 Component relationships

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_MSA.3 Static attribute initialization

### 10.9.5.2 FDP\_ITC.1.1

The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow control SFP(s)*] when importing user data, controlled under the SFP, from outside of the TOE.

### 10.9.5.3 FDP\_ITC.1.2

The TSF **shall** ignore any security attributes associated with the user data when imported from outside the TOE.

### 10.9.5.4 FDP\_ITC.1.3

The TSF **shall** enforce the following rules when importing user data controlled under the SFP from outside the TOE: [assignment: *additional importation control rules*].



2616 **10.9.6 FDP\_ITC.2 Import of user data with security attributes**2617 **10.9.6.1 Component relationships**

2618	Hierarchical to:	No other components.
2619	Dependencies:	[FDP_ACC.1 Subset access control, or
2620		FDP_IFC.1 Subset information flow control]
2621		[FTP_ITC.1 Inter-TSF trusted channel, or
2622		FTP_TRP.1 Trusted path]
2623		FPT_TDC.1 Inter-TSF basic TSF data consistency

2624 **10.9.6.2 FDP\_ITC.2.1**

2625 The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow*  
 2626 *control SFP(s)*] when importing user data, controlled under the SFP, from outside of the  
 2627 TOE.

2628 **10.9.6.3 FDP\_ITC.2.2**

2629 The TSF **shall** use the security attributes associated with the imported user data.

2630 **10.9.6.4 FDP\_ITC.2.3**

2631 The TSF **shall** ensure that the protocol used provides for the unambiguous association  
 2632 between the security attributes and the user data received.

2633 **10.9.6.5 FDP\_ITC.2.4**

2634 The TSF **shall** ensure that interpretation of the security attributes of the imported user  
 2635 data is as intended by the source of the user data.

2636 **10.9.6.6 FDP\_ITC.2.5**

2637 The TSF **shall** enforce the following rules when importing user data controlled under the  
 2638 SFP from outside the TOE: [assignment: *additional importation control rules*].

2639 **10.10 Internal TOE transfer (FDP\_ITT)**2640 **10.10.1 Family behaviour**

2641 This family provides requirements that address protection of user data when it is transferred  
 2642 between separated parts of a TOE across an internal channel. This **may** be contrasted with the  
 2643 Inter-TSF user data confidentiality transfer protection (FDP\_UCT) and Inter-TSF user data  
 2644 integrity transfer protection (FDP\_UIT) families, which provide protection for user data when it  
 2645 is transferred between distinct TSFs across an external channel, and Export from the TOE  
 2646 (FDP\_ETC) and Import from outside of the TOE (FDP\_ITC), which address TSF-mediated  
 2647 transfer of data to or from outside the TOE.

2648 **10.10.2 Components leveling and description**

2649 Figure 32 shows the component leveling for this family.





**Figure 32 — FDP\_ITT: Component leveling**

- 2651 FDP\_ITT.1 Basic internal transfer protection, requires that user data be protected when  
 2652 transmitted between parts of the TOE.
- 2653 FDP\_ITT.2 Transmission separation by attribute, requires separation of data based on the value  
 2654 of SFP-relevant attributes in addition to the first component.
- 2655 FDP\_ITT.3 Integrity monitoring, requires that the TSF monitor user data transmitted between  
 2656 parts of the TOE for identified integrity errors.
- 2657 FDP\_ITT.4 Attribute-based integrity monitoring expands on the third component by allowing  
 2658 the form of integrity monitoring to differ by SFP-relevant attribute.

**10.10.3 Management of FDP\_ITT.1, FDP\_ITT.2**

2660 The following actions **could** be considered for the management functions in FMT:

- 2661 a) If the TSF provides multiple methods to protect user data during transmission  
 2662 between physically separated parts of the TOE, the TSF **could** provide a pre-defined  
 2663 role with the ability to select the method that will be used.

**10.10.4 Management of FDP\_ITT.3, FDP\_ITT.4**

2665 The following actions **could** be considered for the management functions in FMT:

- 2666 a) The specification of the actions to be taken upon detection of an integrity error  
 2667 **could** be configurable.

**10.10.5 Audit of FDP\_ITT.1, FDP\_ITT.2**

2669 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 2670 in the PP/ST:

- 2671 a) Minimal: Successful transfers of user data, including identification of the protection  
 2672 method used.
- 2673 b) Basic: All attempts to transfer user data, including the protection method used and  
 2674 any errors that occurred.

**10.10.6 Audit of FDP\_ITT.3, FDP\_ITT.4**

2676 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 2677 in the PP/ST:

- 2678 a) Minimal: Successful transfers of user data, including identification of the integrity  
 2679 protection method used.
- 2680 b) Basic: All attempts to transfer user data, including the integrity protection method  
 2681 used and any errors that occurred.
- 2682 c) Basic: Unauthorized attempts to change the integrity protection method.
- 2683 d) Detailed: The action taken upon detection of an integrity error.

**10.10.7 FDP\_ITT.1 Basic internal transfer protection****10.10.7.1 Component relationships**

- 2686 Hierarchical to: No other components.
- 2687 Dependencies: [FDP\_ACC.1 Subset access control, or  
 2688 FDP\_IFC.1 Subset information flow control]

2689 **10.10.7.2 FDP\_ITT.1.1**

2690 The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow*  
 2691 *control SFP(s)*] to prevent the [selection: *disclosure, modification, loss of use*] of user data  
 2692 when it is transmitted between physically-separated parts of the TOE.

2693 **10.10.8 FDP\_ITT.2 Transmission separation by attribute**2694 **10.10.8.1 Component relationships**

2695 Hierarchical to: FDP\_ITT.1 Basic internal transfer protection  
 2696 Dependencies: [FDP\_ACC.1 Subset access control, or  
 2697 FDP\_IFC.1 Subset information flow control]

2698 **10.10.8.2 FDP\_ITT.2.1**

2699 The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow control*  
 2700 *SFP(s)*] to prevent the [selection: *disclosure, modification, loss of use*] of user data when it is  
 2701 transmitted between physically-separated parts of the TOE.

2702 **10.10.8.3 FDP\_ITT.2.2**

2703 The TSF **shall** separate data controlled by the SFP(s) when transmitted between  
 2704 physically-separated parts of the TOE, based on the values of the following: [assignment:  
 2705 *security attributes that require separation*].

2706 **10.10.9 FDP\_ITT.3 Integrity monitoring**2707 **10.10.9.1 Component relationships**

2708 Hierarchical to: No other components.  
 2709 Dependencies: [FDP\_ACC.1 Subset access control, or  
 2710 FDP\_IFC.1 Subset information flow control]  
 2711 FDP\_ITT.1 Basic internal transfer protection

2712 **10.10.9.2 FDP\_ITT.3.1**

2713 The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow*  
 2714 *control SFP(s)*] to monitor user data transmitted between physically-separated parts of  
 2715 the TOE for the following errors: [assignment: *integrity errors*].

2716 **10.10.9.3 FDP\_ITT.3.2**

2717 Upon detection of a data integrity error, the TSF **shall** [assignment: *specify the action to*  
 2718 *be taken upon integrity error*].

2719 **10.10.10 FDP\_ITT.4 Attribute-based integrity monitoring**2720 **10.10.10.1 Component relationships**

2721 Hierarchical to: FDP\_ITT.3 Integrity monitoring  
 2722 Dependencies: [FDP\_ACC.1 Subset access control, or  
 2723 FDP\_IFC.1 Subset information flow control]  
 2724 FDP\_ITT.2 Transmission separation by attribute

#### 10.10.10.2 FDP\_ITT.4.1

The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow control SFP(s)*] to monitor user data transmitted between physically-separated parts of the TOE for the following errors: [assignment: *integrity errors*], **based on the following attributes**:  
[assignment: *security attributes that require separate transmission channels*].

#### 10.10.10.3 FDP\_ITT.4.2

Upon detection of a data integrity error, the TSF **shall** [assignment: *specify the action to be taken upon integrity error*].

### 10.11 Residual information protection (FDP\_RIP)

#### 10.11.1 Family behaviour

This family addresses the need to ensure that any data contained in a resource is not available when the resource is de-allocated from one object and reallocated to a different object. This family requires protection for any data contained in a resource that has been logically deleted or released but **may** still be present within the TSF-controlled resource which in turn **may** be re-allocated to another object.

#### 10.11.2 Components leveling and description

Figure 33 shows the component leveling for this family.

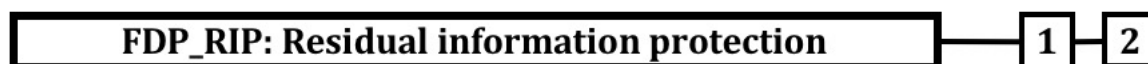


Figure 33 — FDP\_RIP: Component leveling

FDP\_RIP.1 Subset residual information protection, requires that the TSF ensure that any residual information content of any resources is unavailable to a defined subset of the objects controlled by the TSF upon the resource's allocation or deallocation.

FDP\_RIP.2 Full residual information protection, requires that the TSF ensure that any residual information content of any resources is unavailable to all objects upon the resource's allocation or deallocation.

#### 10.11.3 Management of FDP\_RIP.1, FDP\_RIP.2

The following actions **could** be considered for the management functions in FMT:

- a) The choice of when to perform residual information protection (i.e. upon allocation or deallocation) **could** be made configurable within the TOE.

#### 10.11.4 Audit of FDP\_RIP.1, FDP\_RIP.2

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) There are no auditable events foreseen.

#### 10.11.5 FDP\_RIP.1 Subset residual information protection

##### 10.11.5.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

2761 **10.11.5.2 FDP\_RIP.1.1**

2762 The TSF **shall** ensure that any previous information content of a resource is made  
 2763 unavailable upon the [selection: *allocation of the resource to, deallocation of the resource*  
 2764 *from*] the following objects: [assignment: *list of objects*].

2765 **10.11.6 FDP\_RIP.2 Full residual information protection**2766 **10.11.6.1 Component relationships**

2767 Hierarchical to: FDP\_RIP.1 Subset residual information protection

2768 Dependencies: No dependencies.

2769 **10.11.6.2 FDP\_RIP.2.1**

2770 The TSF **shall** ensure that any previous information content of a resource is made unavailable  
 2771 upon the [selection: *allocation of the resource to, deallocation of the resource from*] **all** objects.

2772 **10.12 Rollback (FDP\_ROL)**2773 **10.12.1 Family behaviour**

2774 The rollback operation involves undoing the last operation or a series of operations, bounded  
 2775 by some limit, such as a period of time, and return to a previous known state. Rollback provides  
 2776 the ability to undo the effects of an operation or series of operations to preserve the integrity of  
 2777 the user data.

2778 **10.12.2 Components leveling and description**

2779 Figure 34 shows the component leveling for this family.



2780 **Figure 34 — FDP\_ROL: Component leveling**

2781 FDP\_ROL.1 Basic rollback addresses a need to roll back or undo a limited number of operations  
 2782 within the defined bounds.

2783 FDP\_ROL.2 Advanced rollback addresses the need to roll back or undo all operations within the  
 2784 defined bounds.

2785 **10.12.3 Management of FDP\_ROL.1, FDP\_ROL.2**

2786 The following actions **could** be considered for the management functions in FMT:

2787 a) The boundary limit to which rollback **may** be performed **could** be a configurable  
 2788 item within the TOE.

2789 b) Permission to perform a rollback operation **could** be restricted to a well-defined  
 2790 role.

2791 **10.12.4 Audit of FDP\_ROL.1, FDP\_ROL.2**

2792 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 2793 in the PP/ST:

2794 a) Minimal: All successful rollback operations.

2795 b) Basic: All attempts to perform rollback operations.

2796 c) Detailed: All attempts to perform rollback operations, including identification of the  
 2797 types of operations rolled back.

2798 **10.12.5 FDP\_ROL.1 Basic rollback**2799 **10.12.5.1 Component relationships**

2800 Hierarchical to: No other components.  
 2801 Dependencies: [FDP\_ACC.1 Subset access control, or  
 2802 FDP\_IFC.1 Subset information flow control]

2803 **10.12.5.2 FDP\_ROL.1.1**

2804 The TSF **shall** enforce [assignment: *access control SFP(s) and/or information flow control*  
 2805 *SFP(s)*] to permit the rollback of the [assignment: *list of operations*] on the [assignment:  
 2806 *information and/or list of objects*].

2807 **10.12.5.3 FDP\_ROL.1.2**

2808 The TSF **shall** permit operations to be rolled back within the [assignment: *boundary limit*  
 2809 *to which rollback may be performed*].

2810 **10.12.6 FDP\_ROL.2 Advanced rollback**2811 **10.12.6.1 Component relationships**

2812 Hierarchical to: FDP\_ROL.1 Basic rollback  
 2813 Dependencies: [FDP\_ACC.1 Subset access control, or  
 2814 FDP\_IFC.1 Subset information flow control]

2815 **10.12.6.2 FDP\_ROL.2.1**

2816 The TSF **shall** enforce [assignment: *access control SFP(s) and/or information flow control SFP(s)*]  
 2817 to permit the rollback of **all** the **operations** on the [assignment: *list of objects*].

2818 **10.12.6.3 FDP\_ROL.2.2**

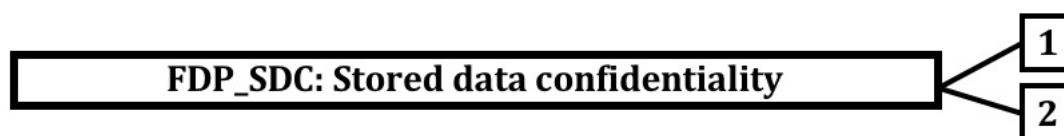
2819 The TSF **shall** permit operations to be rolled back within the [assignment: *boundary limit to*  
 2820 *which rollback may be performed*].

2821 **10.13 Stored data confidentiality (FDP\_SDC)**2822 **10.13.1 Family behaviour**

2823 This family provides requirements that address protection of user data confidentiality while  
 2824 these data are stored within memory areas protected by the TSF. The TSF provides access to the  
 2825 data in the memory through the specified interfaces only and prevents compromise of their  
 2826 information bypassing these interfaces. It complements the family Stored data integrity  
 2827 (FDP\_SDI) which protects the user data from integrity errors while being stored in the memory.

2828 **10.13.2 Components leveling and description**

2829 Figure 35 shows the component leveling for this family.



2830 **Figure 35 — FDP\_SDC: Component leveling**

2831 FDP\_SDC.1 Stored data confidentiality, requires the TSF to protect the confidentiality of  
2832 information of the user data in specified memory areas.

2833 FDP\_SDC.2 Stored data confidentiality with dedicated method, requires the TSF to protect the  
2834 confidentiality of the user data according to data characteristics leading to specify a dedicated  
2835 method of protection of confidentiality.

### 2836 **10.13.3 Management of FDP\_SDC.1, FDP\_SDC.2**

2837 The following actions **could** be considered for the management functions in FMT:

2838 a) No specific management functions are identified

### 2839 **10.13.4 Audit of FDP\_SDC.1, FDP\_SDC.2**

2840 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
2841 in the PP/ST:

2842 a) There are no auditable events foreseen.

### 2843 **10.13.5 FDP\_SDC.1 Stored data confidentiality**

#### 2844 **10.13.5.1 Component relationships**

2845 Hierarchical to: No other components.

2846 Dependencies: No dependencies.

#### 2847 **10.13.5.2 FDP\_SDC.1.1**

2848 The TSF **shall** ensure the confidentiality of user data while it is stored in the [selection:  
2849 *temporary memory, persistent memory, any memory*].

### 2850 **10.13.6 FDP\_SDC.2 Stored data confidentiality with dedicated method**

#### 2851 **10.13.6.1 Component relationships**

2852 Hierarchical to: No other components.

2853 Dependencies: FCS\_COP.1.

#### 2854 **10.13.6.2 FDP\_SDC.2.1**

2855 The TSF **shall** ensure the confidentiality of the user data according to [assignment: data  
2856 characteristics] while it is stored in the TSF.

#### 2857 **10.13.6.3 FDP\_SDC.2.2**

2858 The TSF **shall** ensure the confidentiality of user data without user intervention.

### 2859 **10.13.7 FDP\_SDC.3 Stored data confidentiality with user credential**

2860 **Editors' Note:**

2861 **WD2 NIAP / 16 proposed:**

2862 "What about another component including a request of a user credential as element of protection  
2863 method?"

2864 Editors request comments on this proposal, and in case of agreement, contributions of text, leveling and  
2865 the application notes in response to CD1.

#### 2866 **10.13.7.1 Component relationships**

2867 Hierarchical to:

2868 Dependencies:

2869 **10.13.7.2 FDP\_SDC.3.1**

2870 <TBD>

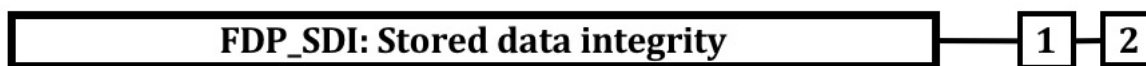
## 2871 **10.14 Stored data integrity (FDP\_SDI)**

### 2872 **10.14.1 Family behaviour**

2873 This family provides requirements that address protection of user data while it is stored within  
2874 containers controlled by the TSF. Integrity errors **may** affect user data stored in memory, or in a  
2875 storage device. This family differs from Internal TOE transfer (FDP\_ITT) which protects the user  
2876 data from integrity errors while being transferred within the TOE.

### 2877 **10.14.2 Components leveling and description**

2878 Figure 36 shows the component leveling for this family.



2879 **Figure 36 — FDP\_SDI: Component leveling**

2880 FDP\_SDI.1 Stored data integrity monitoring, requires that the TSF monitor user data stored  
2881 within containers controlled by the TSF for identified integrity errors.

2882 FDP\_SDI.2 Stored data integrity monitoring and action adds the additional capability to the first  
2883 component by allowing for actions to be taken as a result of an error detection.

### 2884 **10.14.3 Management of FDP\_SDI.1**

2885 The following actions **could** be considered for the management functions in FMT:

2886 a) There are no management activities foreseen.

### 2887 **10.14.4 Management of FDP\_SDI.2**

2888 The following actions **could** be considered for the management functions in FMT:

2889 a) The actions to be taken upon the detection of an integrity error **could** be  
2890 configurable.

### 2891 **10.14.5 Audit of FDP\_SDI.1**

2892 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
2893 in the PP/ST:

2894 a) Minimal: Successful attempts to check the integrity of user data, including an  
2895 indication of the results of the check.

2896 b) Basic: All attempts to check the integrity of user data, including an indication of the  
2897 results of the check, if performed.

2898 c) Detailed: The type of integrity error that occurred.

### 2899 **10.14.6 Audit of FDP\_SDI.2**

2900 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
2901 in the PP/ST:

2902 a) Minimal: Successful attempts to check the integrity of user data, including an  
2903 indication of the results of the check.



- 2904            b) Basic: All attempts to check the integrity of user data, including an indication of the  
2905            results of the check, if performed.
- 2906            c) Detailed: The type of integrity error that occurred.
- 2907            d) Detailed: The action taken upon detection of an integrity error.

#### 2908    **10.14.7 FDP\_SDI.1 Stored data integrity monitoring**

##### 2909    **10.14.7.1 Component relationships**

- 2910            Hierarchical to:                      No other components.
- 2911            Dependencies:                              No dependencies.

##### 2912    **10.14.7.2 FDP\_SDI.1.1**

2913    The TSF **shall** monitor user data stored in containers controlled by the TSF for  
2914    [assignment: *integrity errors*] on all objects, based on the following attributes:  
2915    [assignment: *user data attributes*].

#### 2916    **10.14.8 FDP\_SDI.2 Stored data integrity monitoring and action**

- 2917            Hierarchical to:                      FDP\_SDI.1 Stored data integrity monitoring
- 2918            Dependencies:                              No dependencies.

##### 2919    **10.14.8.1 FDP\_SDI.2.1**

2920    The TSF **shall** monitor user data stored in containers controlled by the TSF for [assignment:  
2921    *integrity errors*] on all objects, based on the following attributes: [assignment: *user data*  
2922    *attributes*].

##### 2923    **10.14.8.2 FDP\_SDI.2.2**

2924    Upon detection of a data integrity error, the TSF **shall** [assignment: *action to be taken*].

### 2925    **10.15 Inter-TSF user data confidentiality transfer protection (FDP\_UCT)**

#### 2926    **10.15.1 Family behaviour**

2927    This family defines the requirements for ensuring the confidentiality of user data when it is  
2928    transferred using an external channel between the TOE and another trusted IT product.

#### 2929    **10.15.2 Components leveling and description**

2930    Figure 37 shows the component leveling for this family.

2931

### FDP\_UCT: Inter-TSF user data confidentiality transfer protection

1

2932                      **Figure 37 — FDP\_UCT: Component leveling**

2933    In FDP\_UCT.1 Basic data exchange confidentiality, the goal is to provide protection from  
2934    disclosure of user data while in transit.

#### 2935    **10.15.3 Management of FDP\_UCT.1**

2936    The following actions **could** be considered for the management functions in FMT:



2937 a) There are no management activities foreseen.

#### 2938 **10.15.4 Audit of FDP\_UCT.1**

2939 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
2940 in the PP/ST:

2941 a) Minimal: The identity of any user or subject using the data exchange mechanisms.

2942 b) Basic: The identity of any unauthorized user or subject attempting to use the data  
2943 exchange mechanisms.

2944 c) Basic: A reference to the names or other indexing information useful in identifying  
2945 the user data that was transmitted or received. This **could** include security  
2946 attributes associated with the information.

#### 2947 **10.15.5 FDP\_UCT.1 Basic data exchange confidentiality**

##### 2948 **10.15.5.1 Component relationships**

2949 Hierarchical to: No other components.

2950 Dependencies: [FTP\_ITC.1 Inter-TSF trusted channel, or  
2951 FTP\_TRP.1 Trusted path]  
2952 [FDP\_ACC.1 Subset access control, or  
2953 FDP\_IFC.1 Subset information flow control]

##### 2954 **10.15.5.2 FDP\_UCT.1.1**

2955 The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow*  
2956 *control SFP(s)*] to [selection: *transmit, receive*] user data in a manner protected from  
2957 unauthorized disclosure.

#### 2958 **10.16 Inter-TSF user data integrity transfer protection (FDP\_UIT)**

##### 2959 **10.16.1 Family behaviour**

2960 This family defines the requirements for providing integrity for user data in transit between the  
2961 TOE and another trusted IT product and recovering from detectable errors. At a minimum, this  
2962 family monitors the integrity of user data for modifications. Furthermore, this family supports  
2963 different ways of correcting detected integrity errors.

##### 2964 **10.16.2 Components leveling and description**

2965 Figure 38 shows the component leveling for this family.

2966



2967 **Figure 38 — FDP\_UIT: Component leveling**

2968 FDP\_UIT.1 Data exchange integrity addresses detection of modifications, deletions, insertions,  
2969 and replay errors of the user data transmitted.

2970 FDP\_UIT.2 Source data exchange recovery addresses recovery of the original user data by the  
2971 receiving TSF with help from the source trusted IT product.

2972 FDP\_UIT.3 Destination data exchange recovery addresses recovery of the original user data by  
2973 the receiving TSF on its own without any help from the source trusted IT product.

### 2974 **10.16.3 Management of FDP\_UIT.1, FDP\_UIT.2, FDP\_UIT.3**

2975 The following actions **could** be considered for the management functions in FMT:

2976 a) There are no management activities foreseen.

### 2977 **10.16.4 Audit of FDP\_UIT.1**

2978 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
2979 in the PP/ST:

2980 a) Minimal: The identity of any user or subject using the data exchange mechanisms.

2981 b) Basic: The identity of any user or subject attempting to use the user data exchange  
2982 mechanisms, but who is unauthorized to do so.

2983 c) Basic: A reference to the names or other indexing information useful in identifying  
2984 the user data that was transmitted or received. This **could** include security  
2985 attributes associated with the user data.

2986 d) Basic: Any identified attempts to block transmission of user data.

2987 e) Detailed: The types and/or effects of any detected modifications of transmitted  
2988 user data.

### 2989 **10.16.5 Audit of FDP\_UIT.2, FDP\_UIT.3**

2990 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
2991 in the PP/ST:

2992 a) Minimal: The identity of any user or subject using the data exchange mechanisms;

2993 b) Minimal: Successful recovery from errors including the type of error that was  
2994 detected;

2995 c) Basic: The identity of any user or subject attempting to use the user data exchange  
2996 mechanisms, but who is unauthorized to do so;

2997 d) Basic: A reference to the names or other indexing information useful in identifying  
2998 the user data that was transmitted or received. This **could** include security  
2999 attributes associated with the user data;

3000 e) Basic: Any identified attempts to block transmission of user data;

3001 f) Detailed: The types and/or effects of any detected modifications of transmitted  
3002 user data.

### 3003 **10.16.6 FDP\_UIT.1 Data exchange integrity**

#### 3004 **10.16.6.1 Component relationships**

3005 Hierarchical to: No other components.

3006 Dependencies: [FDP\_ACC.1 Subset access control, or  
3007 FDP\_IFC.1 Subset information flow control]  
3008 [FTP\_ITC.1 Inter-TSF trusted channel, or  
3009 FTP\_TRP.1 Trusted path]

3010 **10.16.6.2 FDP\_UIT.1.1**

3011 The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow*  
 3012 *control SFP(s)*] to [selection: *transmit, receive*] user data in a manner protected from  
 3013 [selection: *modification, deletion, insertion, replay*] errors.

3014 **10.16.6.3 FDP\_UIT.1.2**

3015 The TSF **shall** be able to determine on receipt of user data, whether [selection:  
 3016 *modification, deletion, insertion, replay*] has occurred.

3017 **10.16.7 FDP\_UIT.2 Source data exchange recovery**3018 **10.16.7.1 Component relationships**

3019	Hierarchical to:	No other components.
3020	Dependencies:	[FDP_ACC.1 Subset access control, or
3021		FDP_IFC.1 Subset information flow control]
3022		[FDP_UIT.1 Data exchange integrity, or
3023		FTP_ITC.1 Inter-TSF trusted channel]

3024 **10.16.7.2 FDP\_UIT.2.1**

3025 The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow*  
 3026 *control SFP(s)*] to be able to recover from [assignment: *list of recoverable errors*] with the  
 3027 help of the source trusted IT product.

3028 **10.16.8 FDP\_UIT.3 Destination data exchange recovery**

3029	Hierarchical to:	FDP_UIT.2 Source data exchange recovery
3030	Dependencies:	[FDP_ACC.1 Subset access control, or
3031		FDP_IFC.1 Subset information flow control]
3032		[FDP_UIT.1 Data exchange integrity, or
3033		FTP_ITC.1 Inter-TSF trusted channel]

3034 **10.16.8.1 FDP\_UIT.3.1**

3035 The TSF **shall** enforce the [assignment: *access control SFP(s) and/or information flow control*  
 3036 *SFP(s)*] to be able to recover from [assignment: *list of recoverable errors*] **without any** help  
 3037 **from** the source trusted IT product.

3038

11 Class FIA: Identification and authentication

11.1 Class description

Families in this class address the requirements for functions to establish and verify a claimed user identity.

Identification and authentication is required to ensure that users are associated with the proper security attributes

The unambiguous identification of authorized users and the correct association of security attributes with users and subjects is critical to the enforcement of the intended security policies. The families in this class deal with determining and verifying the identity of users, determining their authority to interact with the TOE, and with the correct association of security attributes for each authorized user. Other classes of requirements are dependent upon correct identification and authentication of users in order to be effective.

Figure 39 shows the decomposition of this class, it's families and components. Elements are not shown in the figure.

Annex G provides explanatory information for this class and **should** be consulted when using the components identified in this class.

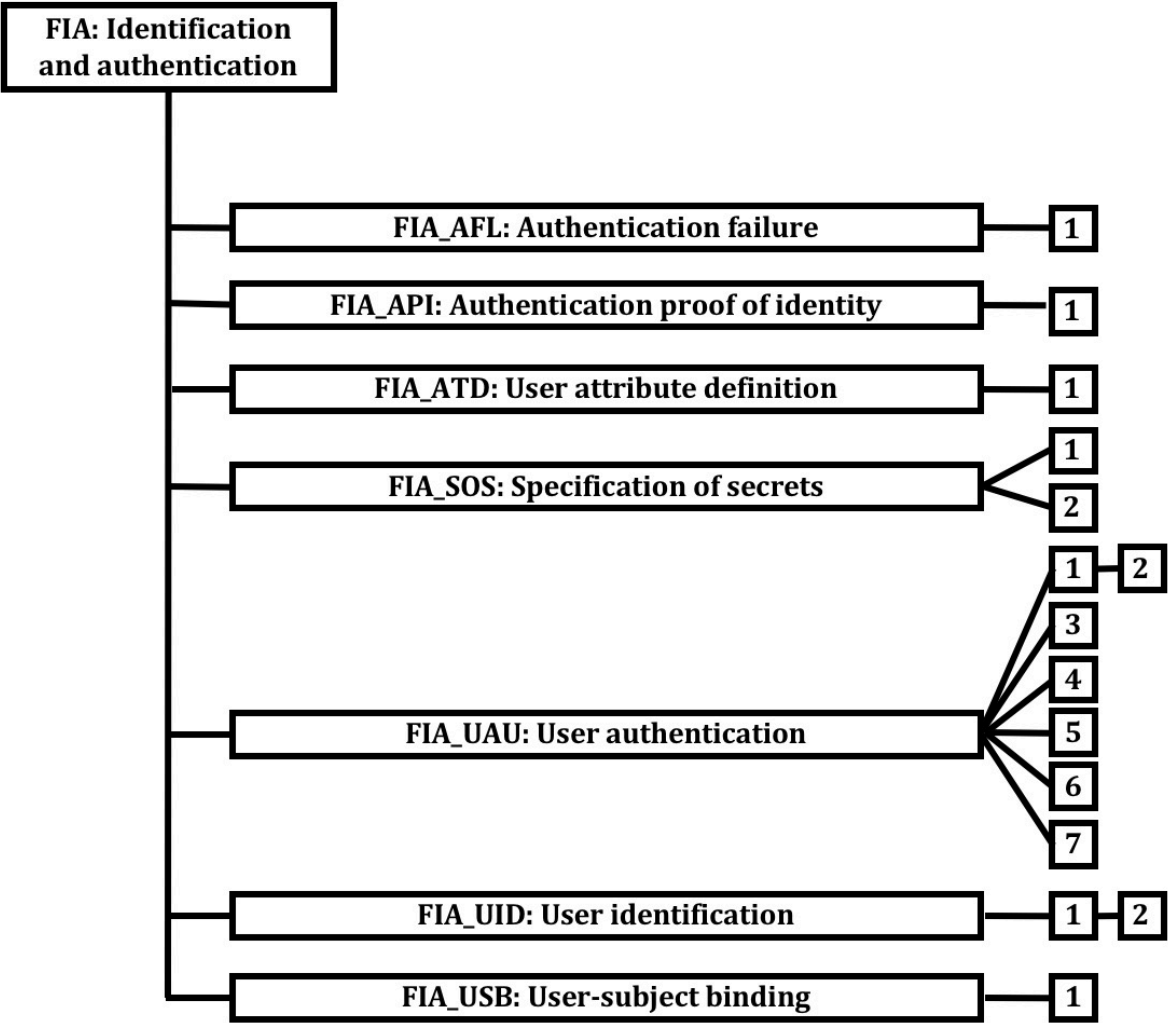


Figure 39 — FIA: Identification and authentication class decomposition

## 11.2 Authentication failures (FIA\_AFL)

### 11.2.1 Family behaviour

This family contains requirements for defining values for some number of unsuccessful authentication attempts and TSF actions in cases of authentication attempt failures. Parameters include, but are not limited to, the number of failed authentication attempts and time thresholds.

### 11.2.2 Components leveling and description

Figure 40 shows the component leveling for this family.

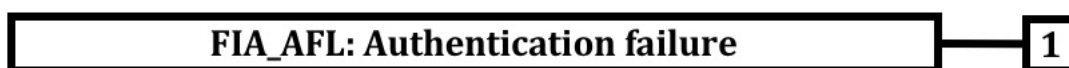


Figure 40 — FIA\_AFL: Component leveling

FIA\_AFL.1 Authentication failure handling, requires that the TSF be able to terminate the session establishment process after a specified number of unsuccessful user authentication attempts. It also requires that, after termination of the session establishment process, the TSF be able to disable the user account or the point of entry from which the attempts were made until an administrator-defined condition occurs.

### 11.2.3 Management of FIA\_AFL.1

The following actions **could** be considered for the management functions in FMT:

- a) Management of the threshold for unsuccessful authentication attempts;
- b) Management of actions to be taken in the event of an authentication failure.

### 11.2.4 Audit of FIA\_AFL.1

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: the reaching of the threshold for the unsuccessful authentication attempts and the actions taken and the subsequent, if appropriate, restoration to the normal state.

### 11.2.5 FIA\_AFL.1 Authentication failure handling

#### 11.2.5.1 Component relationships

Hierarchical to:	No other components.
Dependencies:	FIA_UAU.1 Timing of authentication

#### 11.2.5.2 FIA\_AFL.1.1

The TSF **shall** detect when [selection: [assignment: *positive integer number*], an administrator configurable positive integer within [assignment: *range of acceptable values*]] unsuccessful authentication attempts occur related to [assignment: *list of authentication events*].

#### 11.2.5.3 FIA\_AFL.1.2

When the defined number of unsuccessful authentication attempts has been [selection: *met, surpassed*], the TSF **shall** [assignment: *list of actions*].

3093 **11.3 Authentication proof of identity (FIA\_API)**

3094 **11.3.1 Family behaviour**

3095 This family defines functions provided by the TOE to prove its identity and to be verified by an  
3096 external entity in the TOE IT environment.

3097 **11.3.2 Components leveling and description**

3098 Figure 41 shows the component leveling for this family.

**FIA\_API: Authentication proof of identity**

**1**

3099 **Figure 41 — FIA\_API: Component leveling**

3100 FIA\_API.1 Authentication Proof of Identity, provides prove of the identity of the TOE to an  
3101 external entity.

3102 **11.3.3 Management of FIA\_API.1**

3103 a) There are no management activities foreseen.

3104 **11.3.4 Management of FIA\_API.1**

3105 The following actions **could** be considered for the management functions in FMT:

3106 a) Management of authentication information used to prove the claimed identity.

3107 **11.3.5 Audit of FIA\_API.1**

3108 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3109 in the PP/ST:

3110 a) There are no auditable events foreseen.

3111 **11.3.6 FIA\_API.1 Authentication proof of identity**

3112 **11.3.6.1 Component relationships**

3113 Hierarchical to: No other components.

3114 Dependencies: No dependencies.

3115 **11.3.6.2 FIA\_API.1.1**

3116 The TSF **shall** provide an [assignment: *authentication mechanism*] to prove the identity of  
3117 the [assignment: *object, authorized user, or role*] to an external entity.

3118 **11.4 User attribute definition (FIA\_ATD)**

3119 **11.4.1 Family behaviour**

3120 All authorized users **may** have a set of security attributes, other than the user's identity, that is  
3121 used to enforce the SFRs. This family defines the requirements for associating user security  
3122 attributes with users as needed to support the TSF in making security decisions.

## 3123 11.4.2 Components leveling and description

3124 Figure 42 shows the component leveling for this family.



3125 **Figure 42 — FIA\_ATD: Component leveling**

3126 FIA\_ATD.1 User attribute definition, allows user security attributes for each user to be  
3127 maintained individually.

## 3128 11.4.3 Management of FIA\_ATD.1

3129 The following actions **could** be considered for the management functions in FMT:

- 3130 a) if so indicated in the assignment, the authorized administrator might be able to
- 3131 define additional security attributes for users.

## 3132 11.4.4 Audit of FIA\_ATD.1

3133 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3134 in the PP/ST:

- 3135 a) There are no auditable events foreseen.

## 3136 11.4.5 FIA\_ATD.1 User attribute definition

### 3137 11.4.5.1 Component relationships

3138 Hierarchical to: No other components.

3139 Dependencies: No dependencies.

### 3140 11.4.5.2 FIA\_ATD.1.1

3141 The TSF **shall** maintain the following list of security attributes belonging to individual  
3142 users: [assignment: *list of security attributes*].

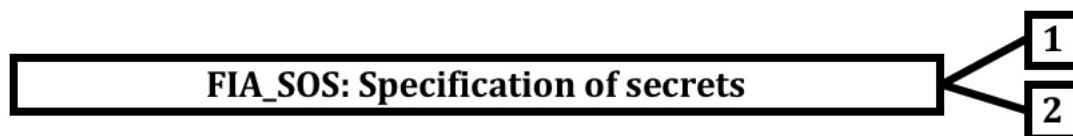
## 3143 11.5 Specification of secrets (FIA\_SOS)

### 3144 11.5.1 Family behaviour

3145 This family defines requirements for mechanisms that enforce defined quality metrics on  
3146 provided secrets and generate secrets to satisfy the defined metric.

### 3147 11.5.2 Components leveling and description

3148 Figure 43 shows the component leveling for this family.



3149 **Figure 43 — FIA\_SOS: Component leveling**

3150 FIA\_SOS.1 Verification of secrets, requires the TSF to verify that secrets meet defined quality  
3151 metrics.

3152 FIA\_SOS.2 TSF Generation of secrets, requires the TSF to be able to generate secrets that meet  
3153 defined quality metrics.

3154 **11.5.3 Management of FIA\_SOS.1**

3155 The following actions **could** be considered for the management functions in FMT:

- 3156 a) the management of the metric used to verify the secrets.

3157 **11.5.4 Management of FIA\_SOS.2**

3158 The following actions **could** be considered for the management functions in FMT:

- 3159 a) the management of the metric used to generate the secrets.

3160 **11.5.5 Audit of FIA\_SOS.1, FIA\_SOS.2**

3161 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3162 in the PP/ST:

- 3163 a) Minimal: Rejection by the TSF of any tested secret;  
3164 b) Basic: Rejection or acceptance by the TSF of any tested secret;  
3165 c) Detailed: Identification of any changes to the defined quality metrics.

3166 **11.5.6 FIA\_SOS.1 Verification of secrets**

3167 **11.5.6.1 Component relationships**

3168 Hierarchical to: No other components.

3169 Dependencies: No dependencies.

3170 **11.5.6.2 FIA\_SOS.1.1**

3171 The TSF **shall** provide a mechanism to verify that secrets meet [assignment: *a defined*  
3172 *quality metric*].

3173 **11.5.7 FIA\_SOS.2 TSF Generation of secrets**

3174 **11.5.7.1 Component relationships**

3175 Hierarchical to: No other components.

3176 Dependencies: No dependencies.

3177 **11.5.7.2 FIA\_SOS.2.1**

3178 The TSF **shall** provide a mechanism to **generate** secrets that meet [assignment: *a defined*  
3179 *quality metric*].

3180 **11.5.7.3 FIA\_SOS.2.2**

3181 The TSF **shall** be able to enforce the use of TSF generated secrets for [assignment: *list of*  
3182 *TSF functions*].

3183 **11.6 User authentication (FIA\_UAU)**

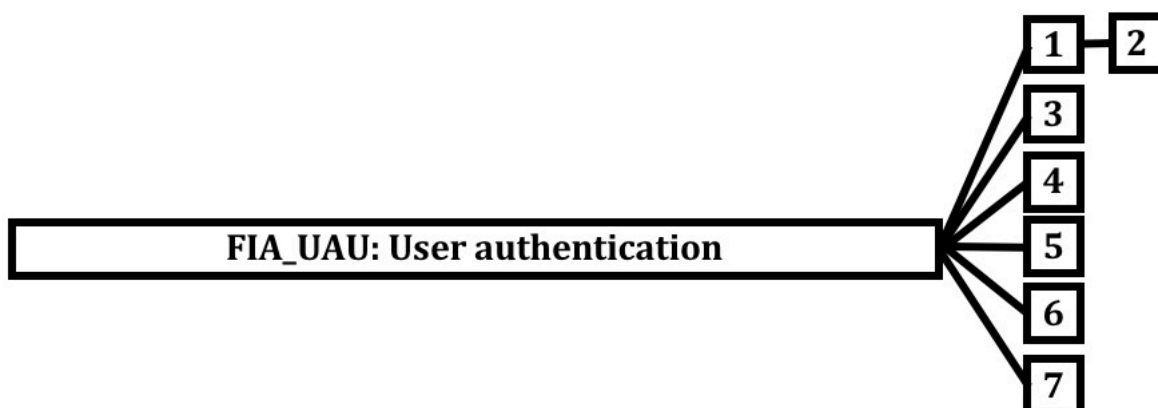
3184 **11.6.1 Family behaviour**

3185 This family defines the types of user authentication mechanisms supported by the TSF. This  
3186 family also defines the required attributes on which the user authentication mechanisms must  
3187 be based.



## 11.6.2 Components leveling and description

Figure 44 shows the component leveling for this family.



**Figure 44 — FIA\_UAU: Component leveling**

FIA\_UAU.1 Timing of authentication, allows a user to perform certain actions prior to the authentication of the user's identity.

FIA\_UAU.2 User authentication before any action, requires that users are authenticated before any other action will be allowed by the TSF.

FIA\_UAU.3 Unforgeable authentication, requires the authentication mechanism to be able to detect and prevent the use of authentication data that has been forged or copied.

FIA\_UAU.4 Single-use authentication mechanisms, requires an authentication mechanism that operates with single-use authentication data.

FIA\_UAU.5 Multiple authentication mechanisms, requires that different authentication mechanisms be provided and used to authenticate user identities for specific events.

FIA\_UAU.6 Re-authenticating, requires the ability to specify events for which the user needs to be re-authenticated.

FIA\_UAU.7 Protected authentication feedback, requires that only limited feedback information is provided to the user during the authentication.

## 11.6.3 Management of FIA\_UAU.1

The following actions **could** be considered for the management functions in FMT:

- a) management of the authentication data by an administrator;
- b) management of the authentication data by the associated user;
- c) managing the list of actions that **can** be taken before the user is authenticated.

## 11.6.4 Management of FIA\_UAU.2

The following actions **could** be considered for the management functions in FMT:

- a) management of the authentication data by an administrator;
- b) management of the authentication data by the user associated with this data.

## 11.6.5 Management of FIA\_UAU.3, FIA\_UAU.4, FIA\_UAU.7

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

3217 **11.6.6 Management of FIA\_UAU.5**

3218 The following actions **could** be considered for the management functions in FMT:

- 3219 a) the management of authentication mechanisms;

3220 **11.6.7 Management of FIA\_UAU.6**

3221 The following actions **could** be considered for the management functions in FMT:

- 3222 a) if an authorized administrator **could** request re-authentication, the management  
3223 includes a re-authentication request.

3224 **11.6.8 Management of FIA\_UAU.7**

3225 The following actions **could** be considered for the management functions in FMT:

- 3226 a) the management of the rules for authentication.

3227 **11.6.9 Audit of FIA\_UAU.1**

3228 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3229 in the PP/ST:

- 3230 a) Minimal: Unsuccessful use of the authentication mechanism;  
3231 b) Basic: All use of the authentication mechanism;  
3232 c) Detailed: All TSF mediated actions performed before authentication of the user.

3233 **11.6.10 Audit of FIA\_UAU.2**

3234 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3235 in the PP/ST:

- 3236 a) Minimal: Unsuccessful use of the authentication mechanism;  
3237 b) Basic: All use of the authentication mechanism.

3238 **11.6.11 Audit of FIA\_UAU.3**

3239 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3240 in the PP/ST:

- 3241 a) Minimal: Detection of fraudulent authentication data;  
3242 b) Basic: All immediate measures taken and results of checks on the fraudulent data.

3243 **11.6.12 Audit of FIA\_UAU.4**

3244 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3245 in the PP/ST:

- 3246 a) Minimal: Attempts to reuse authentication data.

3247 **11.6.13 Audit of FIA\_UAU.5**

3248 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3249 in the PP/ST:

- 3250 a) Minimal: The final decision on authentication;  
3251 b) Basic: The result of each activated mechanism together with the final decision.

3252 **11.6.14 Audit of FIA\_UAU.6**

3253 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3254 in the PP/ST:

- 3255 a) Minimal: Failure of re-authentication;  
 3256 b) Basic: All re-authentication attempts.

### 3257 11.6.15 Audit of FIA\_UAU.7

3258 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 3259 in the PP/ST:

- 3260 a) Well-formedness of rules regarding the semantics of rule-set;  
 3261 b) Basic: verification of rules' enforceability (at their writing).

3262 **Editors' Note:**

3263 **b) should be changed to make it clearer.**

3264 **Comments are requested.**

### 3265 11.6.16 FIA\_UAU.1 Timing of authentication

#### 3266 11.6.16.1 Component relationships

- 3267 Hierarchical to: No other components.  
 3268 Dependencies: FIA\_UID.1 Timing of identification

#### 3269 11.6.16.2 FIA\_UAU.1.1

3270 The TSF **shall** allow [assignment: *list of TSF mediated actions*] on behalf of the user to be  
 3271 performed before the user is authenticated.

#### 3272 11.6.16.3 FIA\_UAU.1.2

3273 The TSF **shall** require each user to be successfully authenticated before allowing any  
 3274 other TSF-mediated actions on behalf of that user.

### 3275 11.6.17 FIA\_UAU.2 User authentication before any action

#### 3276 11.6.17.1 Component relationships

- 3277 Hierarchical to: FIA\_UAU.1 Timing of authentication  
 3278 Dependencies: FIA\_UID.1 Timing of identification

#### 3279 11.6.17.2 FIA\_UAU.2.1

3280 The TSF **shall** require each user to be successfully authenticated before allowing any other TSF-  
 3281 mediated actions on behalf of that user.

### 3282 11.6.18 FIA\_UAU.3 Unforgeable authentication

#### 3283 11.6.18.1 Component relationships

- 3284 Hierarchical to: No other components.  
 3285 Dependencies: No dependencies.

#### 3286 11.6.18.2 FIA\_UAU.3.1

3287 The TSF **shall** [selection: *detect, prevent*] use of authentication data that has been forged  
 3288 by any user of the TSF.

3289	<b>11.6.18.3 FIA_UAU.3.2</b>	
3290	The TSF <b>shall</b> [selection: <i>detect, prevent</i> ] use of authentication data that has been copied	
3291	from any other user of the TSF.	
3292	<b>11.6.19 FIA_UAU.4 Single-use authentication mechanisms</b>	
3293	<b>11.6.19.1 Component relationships</b>	
3294	Hierarchical to:	No other components.
3295	Dependencies:	No dependencies.
3296	<b>11.6.19.2 FIA_UAU.4.1</b>	
3297	The TSF <b>shall</b> prevent reuse of authentication data related to [assignment: <i>identified</i>	
3298	<i>authentication mechanism(s)</i> ].	
3299	<b>11.6.20 FIA_UAU.5 Multiple authentication mechanisms</b>	
3300	<b>11.6.20.1 Component relationships</b>	
3301	Hierarchical to:	No other components.
3302	Dependencies:	No dependencies.
3303	<b>11.6.20.2 FIA_UAU.5.1</b>	
3304	The TSF <b>shall</b> provide [assignment: <i>list of multiple authentication mechanisms</i> ] to support	
3305	user authentication.	
3306	<b>11.6.20.3 FIA_UAU.5.2</b>	
3307	The TSF <b>shall</b> authenticate any user's claimed identity according to the [assignment:	
3308	<i>rules describing how the multiple authentication mechanisms provide authentication</i> ].	
3309	<b>11.6.21 FIA_UAU.6 Re-authenticating</b>	
3310	<b>11.6.21.1 Component relationships</b>	
3311	Hierarchical to:	No other components.
3312	Dependencies:	No dependencies.
3313	<b>11.6.21.2 FIA_UAU.6.1</b>	
3314	The TSF <b>shall</b> re-authenticate the user under the conditions [assignment: <i>list of</i>	
3315	<i>conditions under which re-authentication is required</i> ].	
3316	<b>11.6.22 FIA_UAU.7 Protected authentication feedback</b>	
3317	<b>11.6.22.1 Component relationships</b>	
3318	Hierarchical to:	No other components.
3319	Dependencies:	No dependencies.
3320	<b>Editors' Note:</b>	
3321	Should FIA_UAU.7 be dependent upon "FIA_UID.1 Timing of identification"?	
3322	<b>11.6.22.2 FIA_UAU.7.1</b>	
3323	The TSF <b>shall</b> provide only [assignment: <i>list of feedback</i> ] to the user while the	
3324	authentication is in progress.	

## 3325 11.7 User identification (FIA\_UID)

### 3326 11.7.1 Family behaviour

3327 This family defines the conditions under which users **shall** be required to identify themselves  
3328 before performing any other actions that are to be mediated by the TSF and which require user  
3329 identification.

### 3330 11.7.2 Components leveling and description

3331 Figure 45 shows the component leveling for this family. Figure 44



3332 **Figure 45 — FIA\_UID: Component leveling**

3333 FIA\_UID.1 Timing of identification, allows users to perform certain actions before being  
3334 identified by the TSF.

3335 FIA\_UID.2 User identification before any action, requires that users identify themselves before  
3336 any action will be allowed by the TSF.

### 3337 11.7.3 Management of FIA\_UID.1

3338 The following actions **could** be considered for the management functions in FMT:

- 3339 a) The management of the user identities;
- 3340 b) If an authorized administrator **can** change the actions allowed before identification,
- 3341 the managing of the action lists.

### 3342 11.7.4 Management of FIA\_UID.2

3343 The following actions **could** be considered for the management functions in FMT:

- 3344 a) The management of the user identities;

### 3345 11.7.5 Audit of FIA\_UID.1, FIA\_UID.2

3346 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3347 in the PP/ST:

- 3348 a) Minimal: Unsuccessful use of the user identification mechanism, including the user  
3349 identity provided;
- 3350 b) Basic: All use of the user identification mechanism, including the user identity  
3351 provided.

## 3352 11.7.6 FIA\_UID.1 Timing of identification

### 3353 11.7.6.1 Component relationships

3354 Hierarchical to: No other components.

3355 Dependencies: No dependencies.

### 3356 11.7.6.2 FIA\_UID.1.1

3357 The TSF **shall** allow [assignment: *list of TSF-mediated actions*] on behalf of the user to be  
3358 performed before the user is identified.

3359 **11.7.6.3 FIA\_UID.1.2**

3360 The TSF **shall** require each user to be successfully identified before allowing any TSF-  
 3361 mediated actions on behalf of that user.

3362 **11.7.7 FIA\_UID.2 User identification before any action**

3363 Hierarchical to: FIA\_UID.1 Timing of identification

3364 Dependencies: No dependencies.

3365 **11.7.7.1 FIA\_UID.2.1**

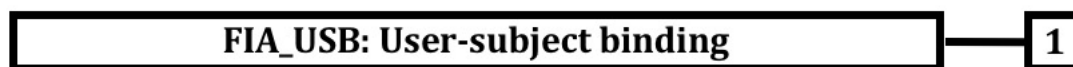
3366 The TSF **shall** require each user to be successfully identified before allowing any TSF-mediated  
 3367 actions on behalf of that user.

3368 **11.8 User-subject binding (FIA\_USB)**3369 **11.8.1 Family behaviour**

3370 An authenticated user, in order to use the TOE, typically activates a subject. The user's security  
 3371 attributes are associated (totally or partially) with this subject. This family defines  
 3372 requirements to create and maintain the association of the user's security attributes to a subject  
 3373 acting on the user's behalf.

3374 **11.8.2 Components leveling and description**

3375 Figure 46 shows the component leveling for this family.



3376 **Figure 46 — FIA\_USB: Component leveling**

3377 FIA\_USB.1 User-subject binding, requires the specification of any rules governing the  
 3378 association between user attributes and the subject attributes into which they are mapped.

3379 **11.8.3 Management of FIA\_USB.1**

3380 The following actions **could** be considered for the management functions in FMT:

3381 a) An authorized administrator **can** define default subject security attributes;

3382 b) An authorized administrator **can** change subject security attributes.

3383 **11.8.4 Audit of FIA\_USB.1**

3384 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 3385 in the PP/ST:

3386 a) Minimal: Unsuccessful binding of user security attributes to a subject

3387 b) Basic: Success and failure of binding of user security attributes to a subject.

3388 **11.8.5 FIA\_USB.1 User-subject binding**3389 **11.8.5.1 Component relationships**

3390 Hierarchical to: No other components.

3391 Dependencies: FIA\_ATD.1 User attribute definition

3392 **11.8.5.2 FIA\_USB.1.1**

3393 The TSF **shall** associate the following user security attributes with subjects acting on the  
3394 behalf of that user: [assignment: *list of user security attributes*].

3395 **11.8.5.3 FIA\_USB.1.2**

3396 The TSF **shall** enforce the following rules on the initial association of user security  
3397 attributes with subjects acting on the behalf of users: [assignment: *rules for the initial*  
3398 *association of attributes*].

3399 **11.8.5.4 FIA\_USB.1.3**

3400 The TSF **shall** enforce the following rules governing changes to the user security  
3401 attributes associated with subjects acting on the behalf of users: [assignment: *rules for*  
3402 *the changing of attributes*].

3403

12 Class FMT: Security management

12.1 Class description

This class is intended to specify the management of several aspects of the TSF: security attributes, TSF data and functions. The different management roles and their interaction, such as separation of capability, can be specified.

This class has several objectives:

- a) Management of TSF data;
- b) Management of security attributes;
- c) Management of functions of the TSF;
- d) Definition of security roles.

Figure 47 shows the decomposition of this class, it's families and components. Elements are not shown in the figure.

Annex H provides explanatory information for this class and should be consulted when using the components identified in this class.

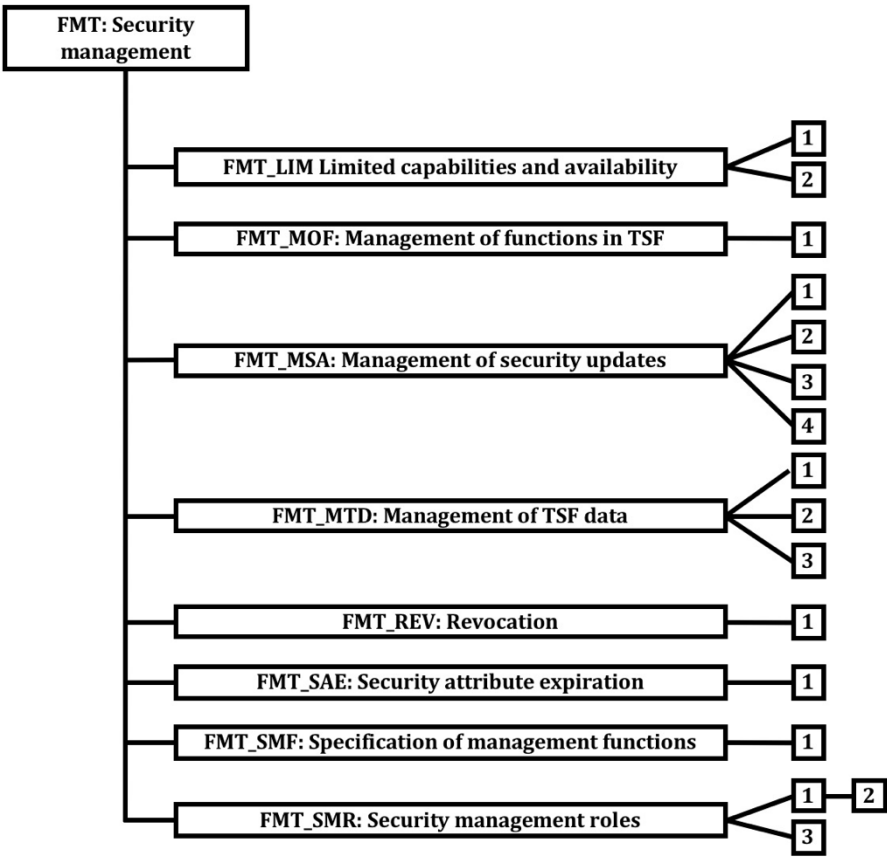


Figure 47 — FMT: Security management class decomposition

12.2 Limited capabilities and availability (FMT\_LIM)

12.2.1 Family behaviour

This family defines requirements that limit the capabilities and availability of functions in a combined manner.



3423 Note FDP\_ACF restricts the access to functions whereas the component Limited Capability of this family  
3424 requires the functions themselves to be designed in a specific manner.

## 3425 12.2.2 Components leveling and description

3426 Figure 48 shows the component leveling for this family.



3427 **Figure 48 — FMT\_LIM: Component leveling**

3428 FMT\_LIM.1 Limited capabilities requires that the TSF is built to provide only the capabilities  
3429 (perform action, gather information) necessary for its genuine purpose.

3430 FMT\_LIM.2 Limited availability requires that the TSF restrict the use of functions (refer to  
3431 Limited capabilities (FMT\_LIM.1)). This **can** be achieved, for instance, by removing or by  
3432 disabling functions in a specific phase of the TOE's life-cycle.

## 3433 12.2.3 Management of FMT\_LIM.1, FMT\_LIM.2

3434 The following actions **could** be considered for the management functions in FMT:

3435 a) There are no management activities foreseen.

## 3436 12.2.4 Audit of FMT\_LIM.1

3437 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3438 in the PP/ST:

3439 a) There are no auditable events foreseen.

## 3440 12.2.5 FMT\_LIM.1 Limited capabilities

### 3441 12.2.5.1 Component relationships

3442 Hierarchical to: No other components.

3443 Dependencies: FMT\_LIM.2 Limited availability

### 3444 12.2.5.2 FMT\_LIM.1.1

3445 The TSF **shall** limit its capabilities so that in conjunction with "Limited availability  
3446 (FMT\_LIM.2)" the following policy is enforced [assignment: Limited capability and  
3447 availability policy].

## 3448 12.2.6 FMT\_LIM.2 Limited availability

### 3449 12.2.6.1 Component relationships

3450 Hierarchical to: No other components.

3451 Dependencies: FMT\_LIM.1 Limited capabilities

### 3452 12.2.6.2 FMT\_LIM.2.1

3453 The TSF **shall** be designed in a manner that limits its availability so that in conjunction  
3454 with "Limited capabilities (FMT\_LIM.1)" the following policy is enforced [assignment:  
3455 *Limited capability and availability policy*].

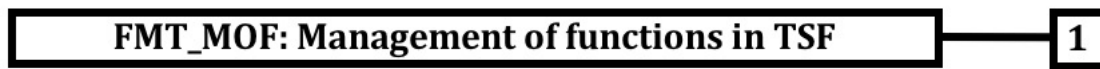
## 3456 12.3 Management of functions in TSF (FMT\_MOF)

### 3457 12.3.1 Family behaviour

3458 This family allows authorized users to control over the management of functions in the TSF.

### 3459 12.3.2 Components leveling and description

3460 Figure 49 shows the component leveling for this family.



3461 **Figure 49 — FMT\_MOF: Component leveling**

3462 FMT\_MOF.1 Management of security functions behaviour allows the authorized users (roles) to  
 3463 manage the behaviour of functions in the TSF that use rules or have specified conditions that  
 3464 **may** be manageable.

### 3465 12.3.3 Management of FMT\_MOF.1

3466 The following actions **could** be considered for the management functions in FMT:

- 3467 a) managing the group of roles that **can** interact with the functions in the TSF.

### 3468 12.3.4 Audit of FMT\_MOF.1

3469 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 3470 in the PP/ST:

- 3471 a) Basic: All modifications in the behaviour of the functions in the TSF.

### 3472 12.3.5 FMT\_MOF.1 Management of security functions behaviour

#### 3473 12.3.5.1 Component relationships

3474	Hierarchical to:	No other components.
3475	Dependencies:	FMT_SMR.1 Security roles
3476		FMT_SMF.1 Specification of Management Functions

#### 3477 12.3.5.2 FMT\_MOF.1.1

3478 The TSF **shall** restrict the ability to [selection: *determine the behaviour of, disable, enable,*  
 3479 *modify the behaviour of*] the functions [assignment: *list of functions*] to [assignment: *the*  
 3480 *authorized identified roles*].

## 3481 12.4 Management of security attributes (FMT\_MSA)

### 3482 12.4.1 Family behaviour

3483 This family allows authorized users control over the management of security attributes. This  
 3484 management might include capabilities for viewing and modifying of security attributes.

## 12.4.2 Components leveling and description

Figure 50 shows the component leveling for this family.



**Figure 50 — FMT\_MSA: Component leveling**

FMT\_MSA.1 Management of security attributes allows authorized users (roles) to manage the specified security attributes.

FMT\_MSA.2 Secure security attributes ensures that values assigned to security attributes are valid with respect to the secure state.

FMT\_MSA.3 Static attribute ensures that the default values of security attributes are appropriately either permissive or restrictive in nature.

FMT\_MSA.4 Security attribute value inheritance allows the rules/policies to be specified that will dictate the value to be inherited by a security attribute.

### 12.4.3 Management of FMT\_MSA.1

The following actions **could** be considered for the management functions in FMT:

- a) Managing the group of roles that **can** interact with the security attributes;
- b) Management of rules by which security attributes inherit specified values.

### 12.4.4 Management of FMT\_MSA.2

The following actions **could** be considered for the management functions in FMT:

- a) Management of rules by which security attributes inherit specified values.

### 12.4.5 Management of FMT\_MSA.3

The following actions **could** be considered for the management functions in FMT:

- a) Managing the group of roles that **can** specify initial values;
- b) Managing the permissive or restrictive setting of default values for a given access control SFP;
- c) Management of rules by which security attributes inherit specified values.

### 12.4.6 Management of FMT\_MSA.4

The following actions **could** be considered for the management functions in FMT:

- a) Specification of the role permitted to establish or modify security attributes.

### 12.4.7 Audit of FMT\_MSA.1

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Basic: All modifications of the values of security attributes.

3516 **12.4.8 Audit of FMT\_MSA.2**

3517 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3518 in the PP/ST:

- 3519 a) Minimal: All offered and rejected values for a security attribute.
- 3520 b) Detailed: All offered and accepted secure values for a security attribute.

3521 **12.4.9 Audit of FMT\_MSA.3**

3522 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3523 in the PP/ST:

- 3524 a) Basic: Modifications of the default setting of permissive or restrictive rules.
- 3525 b) Basic: All modifications of the initial values of security attributes.

3526 **12.4.10 Audit of FMT\_MSA.4**

3527 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3528 in the PP/ST:

- 3529 a) Basic: Modifications of security attributes, possibly with the old and/or values of  
3530 security attributes that were modified.

3531 **12.4.11 FMT\_MSA.1 Management of security attributes**

3532 **12.4.11.1 Component relationships**

3533	Hierarchical to:	No other components.
3534	Dependencies:	[FDP_ACC.1 Subset access control, or
3535		FDP_IFC.1 Subset information flow control]
3536		FMT_SMR.1 Security roles
3537		FMT_SMF.1 Specification of Management Functions

3538 **12.4.11.2 FMT\_MSA.1.1**

3539 The TSF **shall** enforce the [assignment: *access control SFP(s), information flow control*  
3540 *SFP(s)*] to restrict the ability to [selection: *change\_default, query, modify, delete,*  
3541 *[assignment: other operations]*] the security attributes [assignment: *list of security*  
3542 *attributes*] to [assignment: *the authorized identified roles*].

3543 **12.4.12 FMT\_MSA.2 Secure security attributes**

3544 **12.4.12.1 Component relationships**

3545	Hierarchical to:	No other components.
3546	Dependencies:	[FDP_ACC.1 Subset access control, or
3547		FDP_IFC.1 Subset information flow control]
3548		FMT_MSA.1 Management of security attributes
3549		FMT_SMR.1 Security roles

3550 **12.4.12.2 FMT\_MSA.2.1**

3551 The TSF **shall** ensure that only secure values are accepted for [assignment: *list of security*  
3552 *attributes*].

### 12.4.13 FMT\_MSA.3 Static attribute initialization

#### 12.4.13.1 Component relationships

Hierarchical to:	No other components.
Dependencies:	FMT_MSA.1 Management of security attributes
	FMT_SMR.1 Security roles

#### 12.4.13.2 FMT\_MSA.3.1

The TSF **shall** enforce the [assignment: *access control SFP, information flow control SFP*] to provide [selection, choose one of: *restrictive, permissive, [assignment: other property]*] default values for security attributes that are used to enforce the SFP.

#### 12.4.13.3 FMT\_MSA.3.2

The TSF **shall** allow the [assignment: *the authorized identified roles*] to specify alternative initial values to override the default values when an object or information is created.

### 12.4.14 FMT\_MSA.4 Security attribute value inheritance

#### 12.4.14.1 Component relationships

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]

#### 12.4.14.2 FMT\_MSA.4.1

The TSF **shall** use the following rules to set the value of security attributes: [assignment: *rules for setting the values of security attributes*].

## 12.5 Management of TSF data (FMT\_MTD)

### 12.5.1 Family behaviour

This family allows authorized users (roles) control over the management of TSF data.

### 12.5.2 Components leveling and description

Figure 51 shows the component leveling for this family.



**Figure 51 — FMT\_MTD: Component leveling**

FMT\_MTD.1 Management of TSF data allows authorized users to manage TSF data.

FMT\_MTD.2 Management of limits on TSF data specifies the action to be taken if limits on TSF data are reached or exceeded.

FMT\_MTD.3 Secure TSF data ensures that values assigned to TSF data are valid with respect to the secure state.

3584 **12.5.3 Management of FMT\_MTD.1**

3585 The following actions **could** be considered for the management functions in FMT:

- 3586 a) managing the group of roles that **can** interact with the TSF data.

3587 **12.5.4 Management of FMT\_MTD.2**

3588 The following actions **could** be considered for the management functions in FMT:

- 3589 a) managing the group of roles that **can** interact with the limits on the TSF data.

3590 **12.5.5 Management of FMT\_MTD.3**

3591 The following actions **could** be considered for the management functions in FMT:

- 3592 a) There are no management activities foreseen.

3593 **12.5.6 Audit of FMT\_MTD.1**

3594 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3595 in the PP/ST:

- 3596 a) Basic: All modifications to the values of TSF data.

3597 **12.5.7 Audit of FMT\_MTD.2**

3598 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3599 in the PP/ST:

- 3600 a) Basic: All modifications to the limits on TSF data.

- 3601 b) Basic: All modifications in the actions to be taken in case of violation of the limits.

3602 **12.5.8 Audit of FMT\_MTD.3**

3603 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3604 in the PP/ST:

- 3605 a) Minimal: All rejected values of TSF data.

3606 **12.5.9 FMT\_MTD.1 Management of TSF data**

3607 **12.5.9.1 Component relationships**

3608 Hierarchical to: No other components.

3609 Dependencies: FMT\_SMR.1 Security roles

3610 FMT\_SMF.1 Specification of Management Functions

3611 **12.5.9.2 FMT\_MTD.1.1**

3612 The TSF **shall** restrict the ability to [selection: *change\_default, query, modify, delete, clear,*  
3613 *[assignment: other operations]*] the [assignment: *list of TSF data*] to [assignment: *the*  
3614 *authorized identified roles*].

3615 **12.5.10 FMT\_MTD.2 Management of limits on TSF data**

3616 **12.5.10.1 Component relationships**

3617 Hierarchical to: No other components.

3618 Dependencies: FMT\_MTD.1 Management of TSF data

3619 FMT\_SMR.1 Security roles

3620 **12.5.10.2 FMT\_MTD.2.1**

3621 The TSF **shall** restrict the specification of the limits for [assignment: *list of TSF data*] to  
 3622 [assignment: *the authorized identified roles*].

3623 **12.5.10.3 FMT\_MTD.2.2**

3624 The TSF **shall** take the following actions, if the TSF data are at, or exceed, the indicated  
 3625 limits: [assignment: *actions to be taken*].

3626 **12.5.11 FMT\_MTD.3 Secure TSF data**3627 **12.5.11.1 Component relationships**

3628 Hierarchical to: No other components.

3629 Dependencies: FMT\_MTD.1 Management of TSF data

3630 **12.5.11.2 FMT\_MTD.3.1**

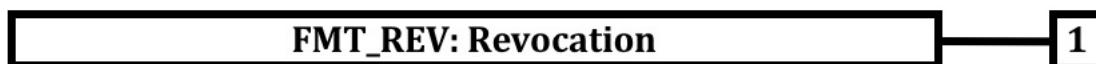
3631 The TSF **shall** ensure that only secure values are accepted for [assignment: *list of TSF*  
 3632 *data*].

3633 **12.6 Revocation (FMT\_REV)**3634 **12.6.1 Family behaviour**

3635 This family addresses revocation of security attributes for a variety of entities within a TOE.

3636 **12.6.2 Components leveling and description**

3637 Figure 52 shows the component leveling for this family.



3638 **Figure 52 — FMT\_REV: Component leveling**

3639 FMT\_REV.1 Revocation provides for revocation of security attributes to be enforced at some  
 3640 point in time.

3641 **12.6.3 Management of FMT\_REV.1**

3642 The following actions **could** be considered for the management functions in FMT:

- 3643 a) Managing the group of roles that **can** invoke revocation of security attributes;
- 3644 b) Managing the lists of users, subjects, objects, and other resources for which  
 3645 revocation is possible;
- 3646 c) Managing the revocation rules.

3647 **12.6.4 Audit of FMT\_REV.1**

3648 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 3649 in the PP/ST:

- 3650 a) Minimal: Unsuccessful revocation of security attributes;
- 3651 b) Basic: All attempts to revoke security attributes.

3652 **12.6.5 FMT\_REV.1 Revocation**3653 **12.6.5.1 Component relationships**

3654 Hierarchical to: No other components.

3655 Dependencies: FMT\_SMR.1 Security roles

3656 **12.6.5.2 FMT\_REV.1.1**

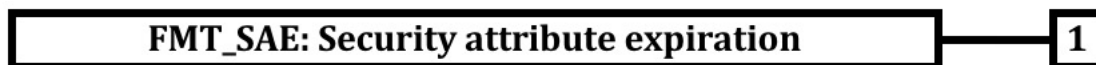
3657 The TSF **shall** restrict the ability to revoke [assignment: *list of security attributes*]  
 3658 associated with the [selection: *users, subjects, objects, [assignment: other additional*  
 3659 *resources*]] under the control of the TSF to [assignment: *the authorized identified roles*].

3660 **12.6.5.3 FMT\_REV.1.2**3661 The TSF **shall** enforce the rules [assignment: *specification of revocation rules*].3662 **12.7 Security attribute expiration (FMT\_SAE)**3663 **12.7.1 Family behaviour**

3664 This family addresses the capability to enforce time limits for the validity of security attributes.

3665 **12.7.2 Components leveling and description**

3666 Figure 53 shows the component leveling for this family.

3667 **Figure 53 — FMT\_SAE: Component leveling**

3668 FMT\_SAE.1 Time-limited authorization provides the capability for an authorized user to specify  
 3669 an expiration time on specified security attributes.

3670 **12.7.3 Management of FMT\_SAE.1**3671 The following actions **could** be considered for the management functions in FMT:

3672 a) Managing the list of security attributes for which expiration is to be supported;

3673 b) The actions to be taken if the expiration time has passed.

3674 **12.7.4 Audit of FMT\_SAE.1**

3675 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 3676 in the PP/ST:

3677 a) Basic: Specification of the expiration time for an attribute;

3678 b) Basic: Action taken due to attribute expiration.

3679 **12.7.5 FMT\_SAE.1 Time-limited authorization**3680 **12.7.5.1 Component relationships**

3681 Hierarchical to: No other components.

3682 Dependencies: FMT\_SMR.1 Security roles

3683 FPT\_STM.1 Reliable time stamps



### 12.7.5.2 FMT\_SAE.1.1

The TSF **shall** restrict the capability to specify an expiration time for [assignment: *list of security attributes for which expiration is to be supported*] to [assignment: *the authorized identified roles*].

### 12.7.5.3 FMT\_SAE.1.2

For each of these security attributes, the TSF **shall** be able to [assignment: *list of actions to be taken for each security attribute*] after the expiration time for the indicated security attribute has passed.

## 12.8 Specification of Management Functions (FMT\_SMF)

### 12.8.1 Family behaviour

This family allows the specification of the management functions to be provided by the TOE. Management functions provide TSFI that allow administrators to define the parameters that control the operation of security-related aspects of the TOE, such as data protection attributes, TOE protection attributes, audit attributes, and identification and authentication attributes. Management functions also include those functions performed by an operator to ensure continued operation of the TOE, such as backup and recovery. This family works in conjunction with the other components in the FMT: Security management class: the component in this family calls out the management functions, and other families in FMT: Security management restrict the ability to use these management functions.

### 12.8.2 Components leveling and description

Figure 54 shows the component leveling for this family.

## FMT\_SMF: Specification of management functions

1

**Figure 54 — FMT\_SMF: Component leveling**

FMT\_SMF.1 Specification of Management Functions requires that the TSF provide specific management functions.

### 12.8.3 Management of FMT\_SMF.1

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

### 12.8.4 Audit of FMT\_SMF.1

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: Use of the management functions.

### 12.8.5 FMT\_SMF.1 Specification of Management Functions

#### 12.8.5.1 Component relationships

Hierarchical to:	No other components.
Dependencies:	No dependencies.

3719 **12.8.5.2 FMT\_SMF.1.1**

3720 The TSF **shall** be capable of performing the following management functions:  
 3721 [assignment: *list of management functions to be provided by the TSF*].

3722 **12.9 Security management roles (FMT\_SMR)**3723 **12.9.1 Family behaviour**

3724 This family is intended to control the assignment of different roles to users. The capabilities of  
 3725 these roles with respect to security management are described in the other families in this class.

3726 **12.9.2 Components leveling and description**

3727 Figure 55 shows the component leveling for this family.



3728 **Figure 55 — FMT\_SMR: Component leveling**

3729 FMT\_SMR.1 Security roles specifies the roles with respect to security that the TSF recognizes.

3730 FMT\_SMR.2 Restrictions on security roles specifies that in addition to the specification of the  
 3731 roles, there are rules that control the relationship between the roles.

3732 FMT\_SMR.3 Assuming roles, requires that an explicit request is given to the TSF to assume a  
 3733 role.

3734 **12.9.3 Management of FMT\_SMR.1**

3735 The following actions **could** be considered for the management functions in FMT:

- 3736 a) Managing the group of users that are part of a role.

3737 **12.9.4 Management of FMT\_SMR.2**

3738 The following actions **could** be considered for the management functions in FMT:

- 3739 a) Managing the group of users that are part of a role;  
 3740 b) Managing the conditions that the roles must satisfy.

3741 **12.9.5 Management of FMT\_SMR.3**

3742 There are no management activities foreseen.

3743 **12.9.6 Audit of FMT\_SMR.1**

3744 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 3745 in the PP/ST:

- 3746 a) Minimal: modifications to the group of users that are part of a role;  
 3747 b) Detailed: every use of the rights of a role.

3748 **12.9.7 Audit of FMT\_SMR.2**

3749 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 3750 in the PP/ST:

- 3751 a) Minimal: modifications to the group of users that are part of a role;

- 3752            b) Minimal: unsuccessful attempts to use a role due to the given conditions on the  
 3753            roles;  
 3754            c) Detailed: every use of the rights of a role.

### 3755    **12.9.8 Audit of FMT\_SMR.3**

3756    The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 3757    in the PP/ST:

- 3758            a) Minimal: explicit request to assume a role.

### 3759    **12.9.9 FMT\_SMR.1 Security roles**

#### 3760    **12.9.9.1 Component relationships**

- |                                  |                                    |
|----------------------------------|------------------------------------|
| 3761            Hierarchical to: | No other components.               |
| 3762            Dependencies:    | FIA_UID.1 Timing of identification |

#### 3763    **12.9.9.2 FMT\_SMR.1.1**

3764    The TSF **shall** maintain the roles [assignment: *the authorized identified roles*].

#### 3765    **12.9.9.3 FMT\_SMR.1.2**

3766    The TSF **shall** be able to associate users with roles.

### 3767    **12.9.10 FMT\_SMR.2 Restrictions on security roles**

#### 3768    **12.9.10.1 Component relationships**

- |                                  |                                    |
|----------------------------------|------------------------------------|
| 3769            Hierarchical to: | FMT_SMR.1 Security roles           |
| 3770            Dependencies:    | FIA_UID.1 Timing of identification |

#### 3771    **12.9.10.2 FMT\_SMR.2.1**

3772    The TSF **shall** maintain the roles: [assignment: *authorized identified roles*].

#### 3773    **12.9.10.3 FMT\_SMR.2.2**

3774    The TSF **shall** be able to associate users with roles.

#### 3775    **12.9.10.4 FMT\_SMR.2.3**

3776    The TSF **shall** ensure that the conditions [assignment: *conditions for the different roles*]  
 3777    are satisfied.

### 3778    **12.9.11 FMT\_SMR.3 Assuming roles**

- |                                  |                          |
|----------------------------------|--------------------------|
| 3779            Hierarchical to: | No other components.     |
| 3780            Dependencies:    | FMT_SMR.1 Security roles |

#### 3781    **12.9.11.1 FMT\_SMR.3.1**

3782    The TSF **shall** require an explicit request to assume the following roles: [assignment: *the*  
 3783    *roles*].

3784

13 Class FPR: Privacy

13.1 Class description

This class contains privacy requirements. These requirements provide a user protection against discovery and misuse of identity by other users.

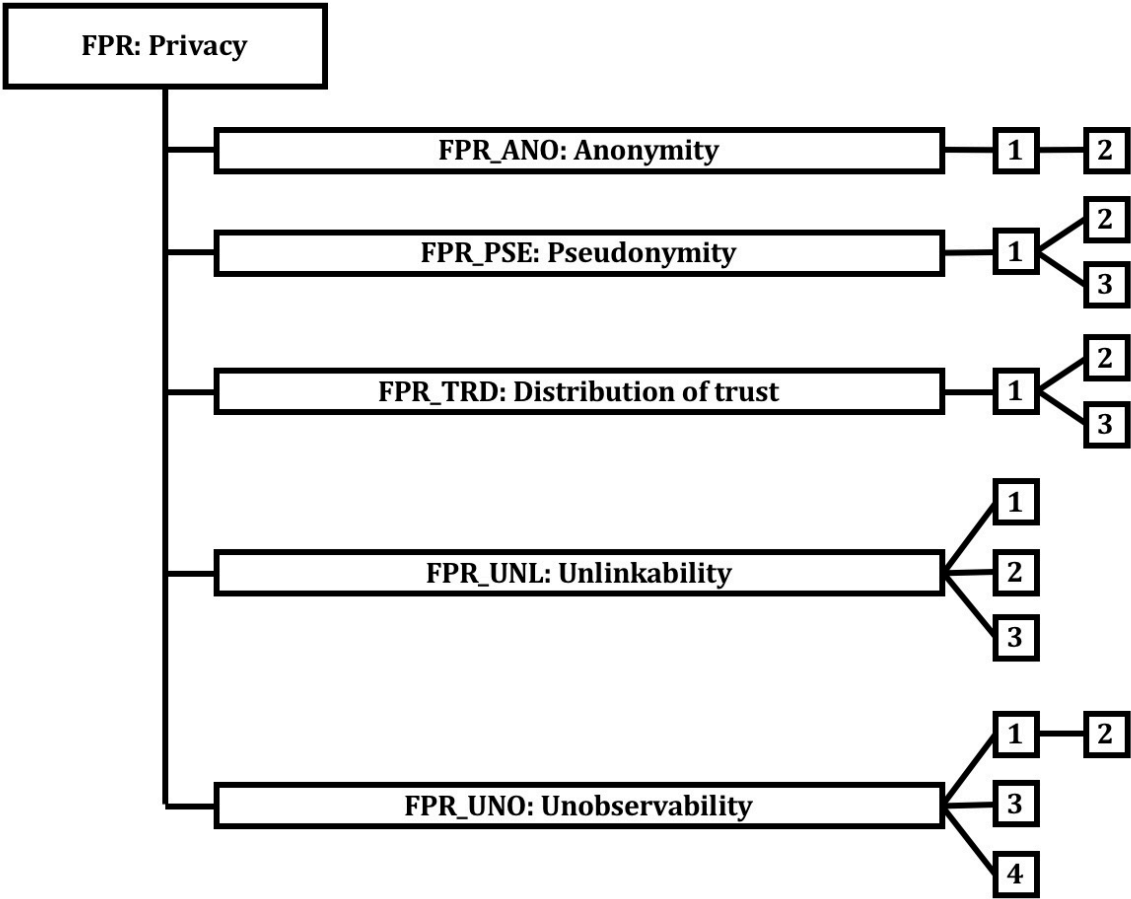
shows the decomposition of this class, it's families and components. Elements are not shown in the figure.

Annex I provides explanatory information for this class and **should** be consulted when using the components identified in this class.

Figure 56 shows the decomposition of this class, it's families and components. Elements are not shown in the figure.

Annex I provides explanatory information for this class and **should** be consulted when using the components identified in this class.

Figure 56 — FPR: Privacy class decomposition



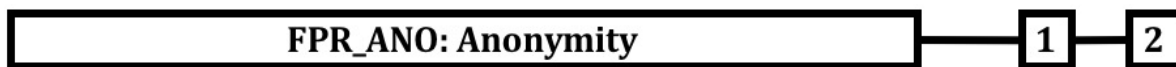
13.2 Anonymity (FPR\_ANO)

13.2.1 Family behaviour

This family ensures that a user **may** use a resource or service without disclosing the user's identity. The requirements for Anonymity provide protection of the user identity. Anonymity is not intended to protect the subject identity.

### 13.2.2 Components leveling and description

Figure 57 shows the component leveling for this family.



**Figure 57 — FPR\_ANO: Component leveling**

FPR\_ANO.1 Anonymity, requires that other users or subjects are unable to determine the identity of a user bound to a subject or operation.

FPR\_ANO.2 Anonymity without soliciting information enhances the requirements of FPR\_ANO.1 Anonymity by ensuring that the TSF does not ask for the user identity.

### 13.2.3 Management of FPR\_ANO.1, FPR\_ANO.2

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

### 13.2.4 Audit of FPR\_ANO.1, FPR\_ANO.2

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: The invocation of the anonymity mechanism.

### 13.2.5 FPR\_ANO.1 Anonymity

#### 13.2.5.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 13.2.5.2 FPR\_ANO.1.1

The TSF **shall** ensure that [assignment: *set of users and/or subjects*] are unable to determine the real user name bound to [assignment: *list of subjects and/or operations and/or objects*].

### 13.2.6 FPR\_ANO.2 Anonymity without soliciting information

#### 13.2.6.1 Component relationships

Hierarchical to: FPR\_ANO.1 Anonymity

Dependencies: No dependencies.

#### 13.2.6.2 FPR\_ANO.2.1

The TSF **shall** ensure that [assignment: *set of users and/or subjects*] are unable to determine the real user name bound to [assignment: *list of subjects and/or operations and/or objects*].

#### 13.2.6.3 FPR\_ANO.2.2

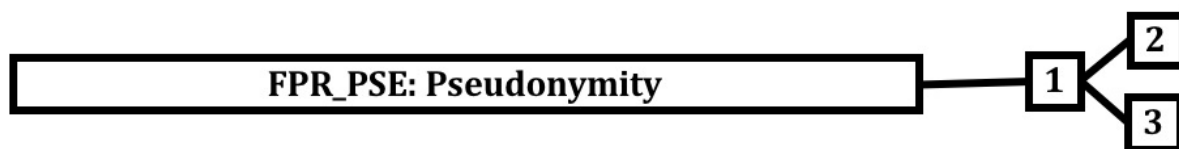
The TSF **shall** provide [assignment: *list of services*] to [assignment: *list of subjects*] without soliciting any reference to the real user name.

3835 **13.3 Pseudonymity (FPR\_PSE)**3836 **13.3.1 Family behaviour**

3837 This family ensures that a user **may** use a resource or service without disclosing its user  
 3838 identity but **can** still be accountable for that use.

3839 **13.3.2 Components leveling and description**

3840 Figure 58 shows the component leveling for this family.



3841 **Figure 58 — FPR\_PSE: Component leveling**

3842 FPR\_PSE.1 Pseudonymity requires that a set of users and/or subjects are unable to determine  
 3843 the identity of a user bound to a subject or operation, but that this user is still accountable for  
 3844 its actions.

3845 FPR\_PSE.2 Reversible pseudonymity, requires the TSF to provide a capability to determine the  
 3846 original user identity based on a provided alias.

3847 FPR\_PSE.3 Alias pseudonymity, requires the TSF to follow certain construction rules for the  
 3848 alias to the user identity.

3849 **13.3.3 Management of FPR\_PSE.1, FPR\_PSE.2, FPR\_PSE.3**

3850 The following actions **could** be considered for the management functions in FMT:

3851 a) There are no management activities foreseen.

3852 **13.3.4 Audit of FPR\_PSE.1, FPR\_PSE.2, FPR\_PSE.3**

3853 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 3854 in the PP/ST:

3855 a) Minimal: The subject/user that requested resolution of the user identity **should** be  
 3856 audited.

3857 **13.3.5 FPR\_PSE.1 Pseudonymity**3858 **13.3.5.1 Component relationships**

3859 Hierarchical to: No other components.

3860 Dependencies: No dependencies.

3861 **13.3.5.2 FPR\_PSE.1.1**

3862 The TSF **shall** ensure that [assignment: *set of users and/or subjects*] are unable to  
 3863 determine the real user name bound to [assignment: *list of subjects and/or operations*  
 3864 *and/or objects*].

3865 **13.3.5.3 FPR\_PSE.1.2**

3866 The TSF **shall** be able to provide [assignment: *number of aliases*] aliases of the real user  
 3867 name to [assignment: *list of subjects*].

3868 **13.3.5.4 FPR\_PSE.1.3**

3869 The TSF **shall** [selection, choose one of: *determine an alias for a user, accept the alias from*  
 3870 *the user*] and verify that it conforms to the [assignment: *alias metric*].

3871 **13.3.6 FPR\_PSE.2 Reversible pseudonymity**3872 **13.3.6.1 Component relationships**

3873 Hierarchical to: FPR\_PSE.1 Pseudonymity

3874 Dependencies: FIA\_UID.1 Timing of identification

3875 **13.3.6.2 FPR\_PSE.2.1**

3876 The TSF **shall** ensure that [assignment: *set of users and/or subjects*] are unable to determine the  
 3877 real user name bound to [assignment: *list of subjects and/or operations and/or objects*].

3878 **13.3.6.3 FPR\_PSE.2.2**

3879 The TSF **shall** be able to provide [assignment: *number of aliases*] aliases of the real user name to  
 3880 [assignment: *list of subjects*].

3881 **13.3.6.4 FPR\_PSE.2.3**

3882 The TSF **shall** [selection, choose one of: *determine an alias for a user, accept the alias from the*  
 3883 *user*] and verify that it conforms to the [assignment: *alias metric*].

3884 **13.3.6.5 FPR\_PSE.2.4**

3885 The TSF **shall** provide [selection: *an authorized user, [assignment: list of trusted subjects]*]  
 3886 a capability to determine the user identity based on the provided alias only under the  
 3887 following [assignment: *list of conditions*].

3888 **13.3.7 FPR\_PSE.3 Alias pseudonymity**3889 **13.3.7.1 Component relationships**

3890 Hierarchical to: FPR\_PSE.1 Pseudonymity

3891 Dependencies: No dependencies.

3892 **13.3.7.2 FPR\_PSE.3.1**

3893 The TSF **shall** ensure that [assignment: *set of users and/or subjects*] are unable to determine the  
 3894 real user name bound to [assignment: *list of subjects and/or operations and/or objects*].

3895 **13.3.7.3 FPR\_PSE.3.2**

3896 The TSF **shall** be able to provide [assignment: *number of aliases*] aliases of the real user name to  
 3897 [assignment: *list of subjects*].

3898 **13.3.7.4 FPR\_PSE.3.3**

3899 The TSF **shall** [selection, choose one of: *determine an alias for a user, accept the alias from the*  
 3900 *user*] and verify that it conforms to the [assignment: *alias metric*].

3901 **13.3.7.5 FPR\_PSE.3.4**

3902 The TSF **shall** provide an alias to the real user name which **shall** be identical to an alias  
 3903 provided previously under the following [assignment: *list of conditions*] otherwise the  
 3904 alias provided **shall** be unrelated to previously provided aliases.



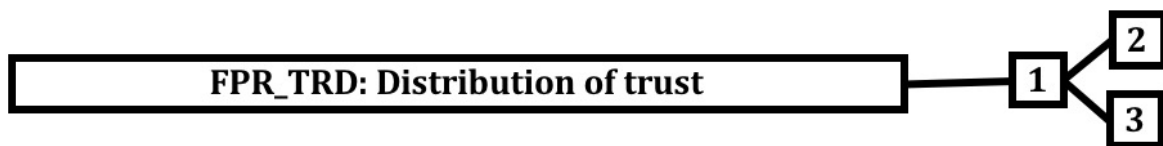
## 3905 13.4 Distribution of trust (FPR\_TRD)

### 3906 13.4.1 Family behaviour

3907 This family addresses the need to ensure that privacy-relevant information referring to a user  
 3908 of a TOE is divided among different parts of the TOE or stored in such a manner (as with  
 3909 encryption) to make it impossible that a part of the TOE under a single administrative domain is  
 3910 able to access such information.

### 3911 13.4.2 Components leveling and description

3912 Figure 59 shows the component leveling for this family.



3913 **Figure 59 — FPR\_TRD: Component leveling**

3914 FPR\_TRD.1 Administrative domains requires that the TOE be divided in distinct administrative  
 3915 domains (AD), with separate authentication and access control procedures; administrators of  
 3916 one administrative domain **may** not access other ADs.

3917 FPR\_TRD.2 Allocation of information assets requires that the TSF ensure that selected  
 3918 information impacting privacy be allocated among different parts of the TOE in such a way that  
 3919 in no state a single administrative domain will be able to access such information.

3920 FPR\_TRD.3 Allocation of processing activities requires that the TSF ensure that selected  
 3921 processing activities impacting privacy be executed on different parts of the TOE in such a way  
 3922 that no single administrative domain will be able to make use of information gathered from the  
 3923 processing activity.

### 3924 13.4.3 Management of FPR\_TRD.1

3925 The following actions **could** be considered for the management functions in FMT:

- 3926 a) There are no management activities foreseen for this component.

### 3927 13.4.4 Management of FPR\_TRD.2

3928 The following actions **could** be considered for the management functions in FMT:

- 3929 a) The FMT\_SMR.1 component **could** define a new security role “information owner”  
 3930 with regard to a specific data object or operation; this role represents the  
 3931 originator, and main user and beneficiary of such object or operation, and is the  
 3932 only subject or user allowed to specify distribution policies as security attributes  
 3933 for these entities;
- 3934 b) An information owner **could** define default object security attributes;
- 3935 c) An information owner **could** define and change security attributes on objects he or  
 3936 she owns.

### 3937 13.4.5 Management of FPR\_TRD.3

3938 The following actions **could** be considered for the management functions in FMT:

- 3939 a) The FMT\_SMR component **could** define a new security role “information owner”  
 3940 with regard to a specific data object or operation; this role represents the  
 3941 originator, and main user and beneficiary of such object or operation, and is the



3942 only subject or user allowed to specify distribution policies as security attributes  
3943 for these entities;

3944 b) An information owner **could** define default operation security attributes;

3945 c) An information owner **could** define and change security attributes on operations it  
3946 initiates.

#### 3947 **13.4.6 Audit of FPR\_TRD.1, FPR\_TRD.2, FPR\_TRD.3**

3948 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
3949 in the PP/ST:

3950 a) There are no events identified that **should** be auditable.

#### 3951 **13.4.7 FPR\_TRD.1 Administrative domains**

##### 3952 **13.4.7.1 Component relationships**

3953 Hierarchical to: No other components.

3954 Dependencies: No dependencies.

##### 3955 **13.4.7.2 FPR\_TRD.1.1**

3956 The TOE **shall** be divided in separate, independent, intercommunicating parts  
3957 (administrative domains) governed by distinct access control and authentication  
3958 configurations.

##### 3959 **13.4.7.3 FPR\_TRD.1.2**

3960 The distinct administrative domains of the TOE **shall** explicitly request access to data  
3961 stored on other parts of the TOE to be granted access to it.

#### 3962 **13.4.8 FPR\_TRD.2 Allocation of information assets**

##### 3963 **13.4.8.1 Component relationships**

3964 Hierarchical to: FPR\_TRD.1 Administrative domains.

3965 Dependencies: No dependencies.

##### 3966 **13.4.8.2 FPR\_TRD.2.1**

3967 The TOE **shall** be divided in separate, independent, intercommunicating parts (administrative  
3968 domains) governed by distinct access control and authentication configurations.

##### 3969 **13.4.8.3 FPR\_TRD.2.2**

3970 The distinct administrative domains of the TOE **shall** explicitly request access to data stored on  
3971 other parts of the TOE to be granted access to it.

##### 3972 **13.4.8.4 FPR\_TRD.2.3**

3973 The TSF **shall** ensure that [assignment: *list of objects*] **shall** be stored [selection: *on*  
3974 *different administrative domains of the TOE, in a form unreadable by a single*  
3975 *administrative domain of the TOE*] as to maintain the following conditions: [assignment:  
3976 *list of conditions on objects*].

#### 3977 **13.4.9 FPR\_TRD.3 Allocation of processing activities**

##### 3978 **13.4.9.1 Component relationships**

3979 Hierarchical to: FPR\_TRD.1 Administrative domains.

3980 Dependencies: No dependencies.

3981 **13.4.9.2 FPR\_TRD.3.1**

3982 The TOE **shall** be divided in separate, independent, intercommunicating parts (administrative  
3983 domains) governed by distinct access control and authentication configurations.

3984 **13.4.9.3 FPR\_TRD.3.2**

3985 The distinct administrative domains of the TOE **shall** explicitly request access to data stored on  
3986 other parts of the TOE to be granted access to it.

3987 **13.4.9.4 FPR\_TRD.3.3**

3988 **The TSF **shall** ensure that [assignment: *list of operations*] **shall** be performed by different**  
3989 **administrative domains of the TOE, so that the following conditions are maintained:**  
3990 **[assignment: *list of conditions on operations*].**

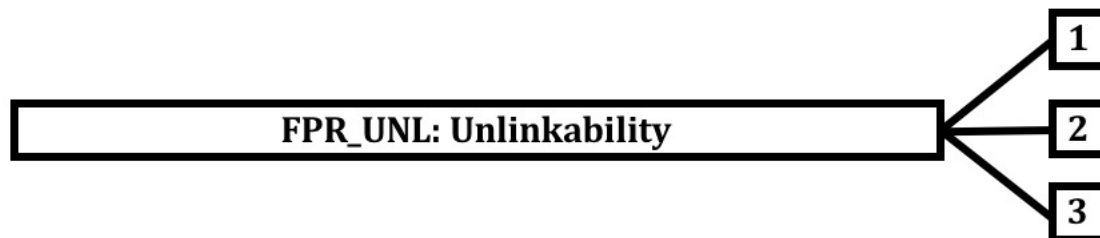
3991 **13.5 Unlinkability (FPR\_UNL)**

3992 **13.5.1 Family behaviour**

3993 This family ensures that selected entities **may** be linked together without external entities being  
3994 able to back trace these links.

3995 **13.5.2 Components leveling and description**

3996 Figure 60 shows the component leveling for this family.



3997 **Figure 60 — FPR\_UNL: Component leveling**

3998 **FPR\_UNL.1 Unlinkability of operations** requires that users and/or subjects are unable to  
3999 determine whether the same user caused certain specific operations in the system, or whether  
4000 operations are related in some other manner. This component ensures that users cannot link  
4001 different operations in the system and thereby obtain information.

4002 **FPR\_UNL.2 Unlinkability of users** requires that users and/or subjects are unable to determine  
4003 whether two users are referenced to by the same object, subject or operation, or are linked in  
4004 some other manner. This component ensures that users cannot link different users of the  
4005 system and thereby obtain information on the communication patterns and relationships  
4006 between users.

4007 **FPR\_UNL.3 Unlinkability of subjects** requires that users and/or subjects are unable to  
4008 determine whether two subjects are referenced to by the same object, user or operation, or are  
4009 linked in some other manner. This component ensures that users cannot link different subjects  
4010 in the system and thereby obtain information on the usage and operation patterns of the  
4011 subjects.

4012 **13.5.3 Management of FPR\_UNL.1, FPR\_UNL.2, FPR\_UNL.3**

4013 The following actions **could** be considered for the management functions in FMT:

4014 a) The management of the unlinkability function.

#### 4015 13.5.4 Audit of FPR\_UNL.1, FPR\_UNL.2, FPR\_UNL.3

4016 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
4017 in the PP/ST:

4018 a) Minimal: The invocation of the unlinkability mechanism.

#### 4019 13.5.5 FPR\_UNL.1 Unlinkability of operations

##### 4020 13.5.5.1 Component relationships

4021 Hierarchical to: No other components.

4022 Dependencies: No dependencies.

##### 4023 13.5.5.2 FPR\_UNL.1.1

4024 The TSF **shall** ensure that [assignment: *set of entities and/or operations*] are unable to  
4025 determine whether [assignment: *list of entities and/or operations*] [selection: *were*  
4026 *caused by the same user, are related as follows [assignment: list of relations]*].

4027 NOTE For “operations” the term transactions should be used.

#### 4028 13.5.6 FPR\_UNL.2 Unlinkability of users

##### 4029 13.5.6.1 Component relationships

4030 Hierarchical to: No other components.

4031 Dependencies: No dependencies.

##### 4032 13.5.6.2 FPR\_UNL.2.1

4033 The TSF shall ensure that [assignment: *set of users and/or subjects*] are unable to determine  
4034 whether [assignment: *list of users*] [selection: *are referenced by the same operation, are*  
4035 *referenced by the same object, are referenced by the same subject, are related as follows*  
4036 *[assignment: list of relations]*].

#### 4037 13.5.7 FPR\_UNL.3 Unlinkability of subjects

##### 4038 13.5.7.1 Component relationships

4039 Hierarchical to: No other components.

4040 Dependencies: No dependencies.

##### 4041 13.5.7.2 FPR\_UNL.3.1

4042 The TSF shall ensure that [assignment: *set of users and/or subjects*] are unable to determine  
4043 whether [assignment: *list of subjects*] [selection: *act on behalf of the same user, are*  
4044 *referenced by the same object, are referenced by the same operation, are related as follows*  
4045 *[assignment: list of relations]*].

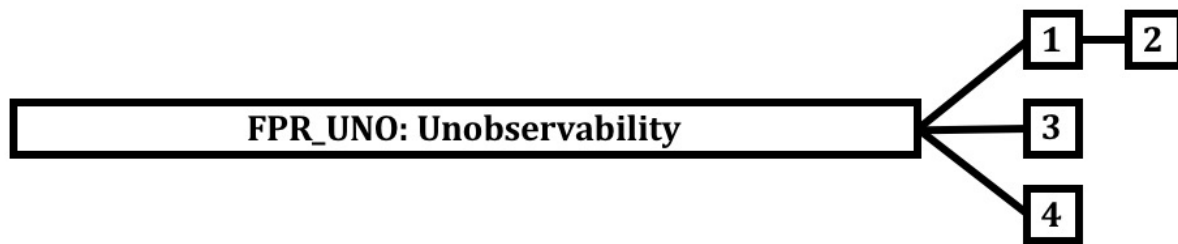
#### 4046 13.6 Unobservability (FPR\_UNO)

##### 4047 13.6.1 Family behaviour

4048 This family ensures that a user **may** use a resource or service without others, especially third  
4049 parties, being able to observe that the resource or service is being used.

4050 **13.6.2 Components leveling and description**

4051 Figure 61 shows the component leveling for this family.

4052 **Figure 61 — FPR\_UNO: Component leveling**

4053 FPR\_UNO.1 Unobservability, requires that users and/or subjects cannot determine whether an  
 4054 operation is being performed.

4055 FPR\_UNO.2 Allocation of information impacting unobservability, requires that the TSF provide  
 4056 specific mechanisms to avoid the concentration of privacy related information within the TOE.  
 4057 Such concentrations might impact unobservability if a security compromise occurs.

4058 FPR\_UNO.3 Unobservability without soliciting information, requires that the TSF does not try to  
 4059 obtain privacy related information that might be used to compromise unobservability.

4060 FPR\_UNO.4 Authorized user observability, requires the TSF to provide one or more authorized  
 4061 users with a capability to observe the usage of resources and/or services.

4062 **13.6.3 Management of FPR\_UNO.1, FPR\_UNO.2**4063 The following actions **could** be considered for the management functions in FMT:

4064 a) The management of the behaviour of the unobservability function.

4065 **13.6.4 Management of FPR\_UNO.3**4066 The following actions **could** be considered for the management functions in FMT:

4067 a) There are no management activities foreseen.

4068 **13.6.5 Management of FPR\_UNO.4**4069 The following actions **could** be considered for the management functions in FMT:

4070 a) The list of authorized users that are capable of determining the occurrence of  
 4071 operations.

4072 **13.6.6 Audit of FPR\_UNO.1, FPR\_UNO.2**

4073 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 4074 in the PP/ST:

4075 a) Minimal: The invocation of the unobservability mechanism.

4076 **13.6.7 Audit of FPR\_UNO.3**

4077 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 4078 in the PP/ST:

4079 a) There are no auditable events foreseen.

4080 **13.6.8 Audit of FPR\_UNO.4**

4081 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 4082 in the PP/ST:

4083 a) Minimal: The observation of the use of a resource or service by a user or subject.

#### 4084 13.6.9 FPR\_UNO.1 Unobservability

##### 4085 13.6.9.1 Component relationships

4086 Hierarchical to: No other components.

4087 Dependencies: No dependencies.

##### 4088 13.6.9.2 FPR\_UNO.1.1

4089 The TSF **shall** ensure that [assignment: *list of users and/or subjects*] are unable to  
4090 observe the operation [assignment: *list of operations*] on [assignment: *list of objects*] by  
4091 [assignment: *list of protected users and/or subjects*].

#### 4092 13.6.10 FPR\_UNO.2 Allocation of information impacting unobservability

##### 4093 13.6.10.1 Component relationships

4094 Hierarchical to: FPR\_UNO.1 Unobservability

4095 Dependencies: No dependencies.

##### 4096 13.6.10.2 FPR\_UNO.2.1

4097 The TSF **shall** ensure that [assignment: *list of users and/or subjects*] are unable to observe the  
4098 operation [assignment: *list of operations*] on [assignment: *list of objects*] by [assignment: *list of*  
4099 *protected users and/or subjects*].

##### 4100 13.6.10.3 FPR\_UNO.2.2

4101 The TSF **shall** allocate the [assignment: *unobservability related information*] among  
4102 different parts of the TOE such that the following conditions hold during the lifetime of  
4103 the information: [assignment: *list of conditions*].

#### 4104 13.6.11 FPR\_UNO.3 Unobservability without soliciting information

##### 4105 13.6.11.1 Component relationships

4106 Hierarchical to: No other components.

4107 Dependencies: FPR\_UNO.1 Unobservability

##### 4108 13.6.11.2 FPR\_UNO.3.1

4109 The TSF **shall** provide [assignment: *list of services*] to [assignment: *list of subjects*]  
4110 without soliciting any reference to [assignment: *privacy related information*].

#### 4111 13.6.12 FPR\_UNO.4 Authorized user observability

##### 4112 13.6.12.1 Component relationships

4113 Hierarchical to: No other components.

4114 Dependencies: No dependencies.

##### 4115 13.6.12.2 FPR\_UNO.4.1

4116 The TSF **shall** provide [assignment: *set of authorized users*] with the capability to observe  
4117 the usage of [assignment: *list of resources and/or services*].

4118

## 4119 **14 Class FPT: Protection of the TSF**

### 4120 **14.1 Class description**

4121 This class contains families of functional requirements that relate to the integrity and  
 4122 management of the mechanisms that constitute the TSF and to the integrity of TSF data. In some  
 4123 sense, families in this class **may** appear to duplicate components in the FDP: User data  
 4124 protection class; they **may** even be implemented using the same mechanisms. However, FDP:  
 4125 User data protection focuses on user data protection, while FPT: Protection of the TSF focuses  
 4126 on TSF data protection. In fact, Components from the FPT: Protection of the TSF class are  
 4127 necessary to provide requirements that the SFPs in the TOE cannot be tampered with or  
 4128 bypassed.

4129 From the point of view of this class, regarding to the TSF there are three significant elements:

- 4130 a) The TSF's implementation, which executes and implements the mechanisms that  
 4131 enforce the SFRs.
- 4132 b) The TSF's data, which are the administrative databases that guide the enforcement  
 4133 of the SFRs.
- 4134 c) The external entities that the TSF **may** interact with in order to enforce the SFRs.

4135 Figure 62 shows the decomposition of this class, it's families and components. Elements are not  
 4136 shown in the figure.

4137 Annex J provides explanatory information for this class and **should** be consulted when using the  
 4138 components identified in this class.

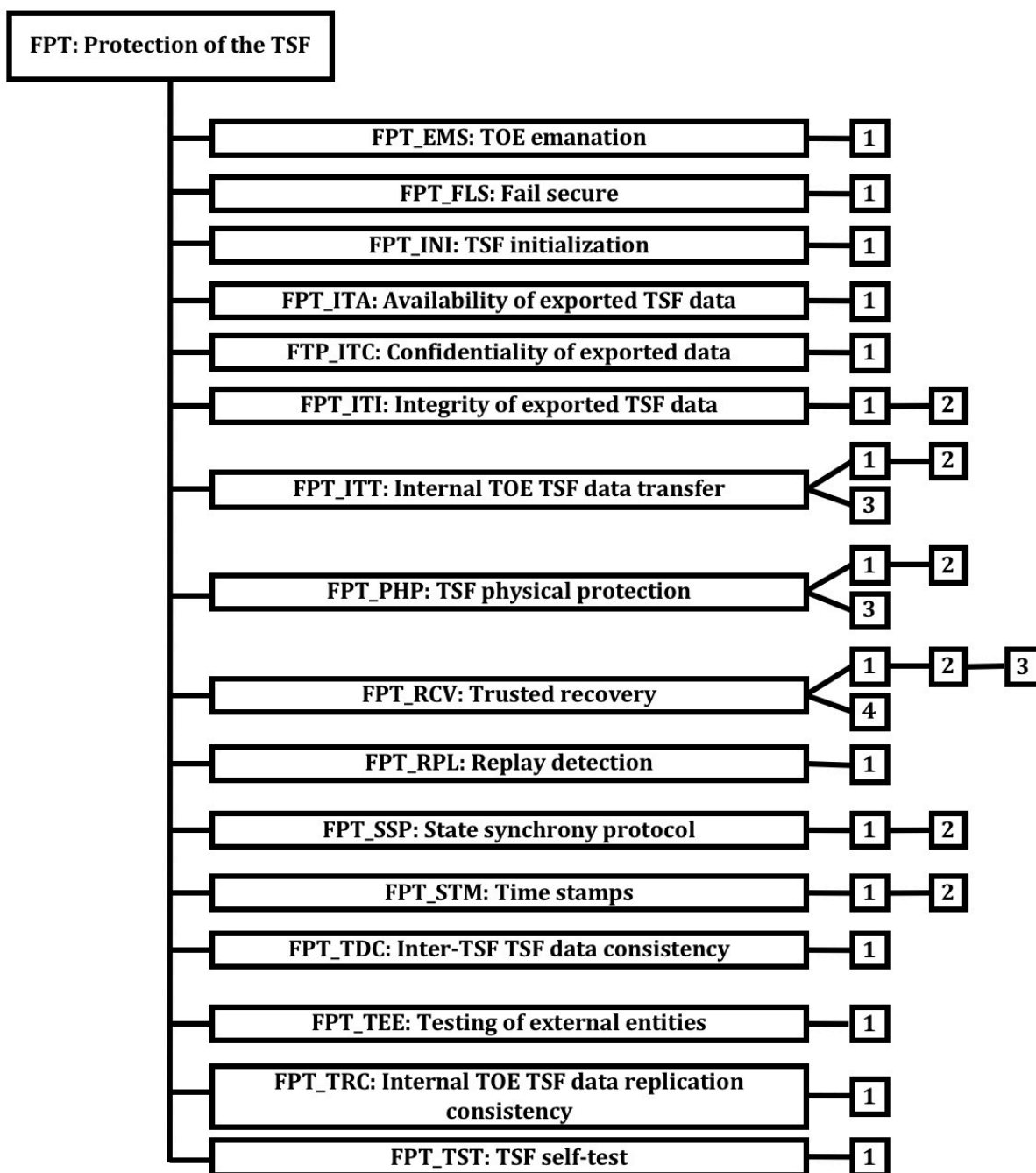


Figure 62 — FPT: Protection of the TSF class decomposition

## 14.2 TOE emanation (FPT\_EMS)

### Editors' Note:

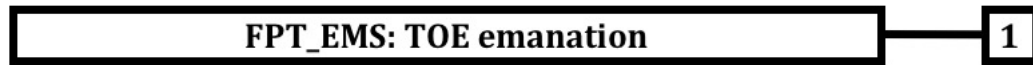
Per GB/TS04: Comments are solicited in regard to whether this SFR can be removed since the requirement can be expressed through more fundamental SFRs as in PP0084.

### 14.2.1 Family behaviour

This family defines the requirements for the TSF to be able to prevent attacks against secret data stored in and used by the TOE where the attack is based on external observable physical phenomena of the TOE.

4148 **14.2.2 Components leveling and description**

4149 Figure 63 shows the component leveling for this family.

4150 **Figure 63 — FPT\_EMS: Component leveling**

4151 This family consists of only one component, FPT\_EMS.1 Emanation of TSF and User data, which  
 4152 defines requirements to mitigate intelligible emanations.

4153 **14.2.3 Management of FPT\_EMS.1**4154 The following actions **could** be considered for the management functions in FMT:

4155 a) There are no management activities foreseen.

4156 **14.2.4 Audit of FPT\_EMS.1**

4157 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 4158 in the PP/ST:

4159 a) There are no auditable events foreseen.

4160 **14.2.5 FPT\_EMS.1 Emanation of TSF and User data**4161 **14.2.5.1 Component relationships**

4162 Hierarchical to: No other components.

4163 Dependencies: No dependencies.

4164 **14.2.5.2 FPT\_EMS.1.1**

4165 The TOE **shall** not emit [assignment: *types of emissions*] in excess of [assignment:  
 4166 *specified limits*] enabling access to [assignment: *list of types of TSF data*] and [assignment:  
 4167 *list of types of user data*].

4168 **14.2.5.3 FPT\_EMS.1.2**

4169 The TSF **shall** ensure [assignment: *type of users*] are unable to use the following interface  
 4170 [assignment: *type of connection*] to gain access to [assignment: *list of types of TSF data*]  
 4171 and [assignment: *list of types of user data*].

4172 **14.3 Fail secure (FPT\_FLS)**4173 **14.3.1 Family behaviour**

4174 The requirements of this family ensure that the TOE will always enforce its SFRs in the event of  
 4175 identified categories of failures in the TSF.

4176

4177 **14.3.2 Components leveling and description**

4178 Figure 64 shows the component leveling for this family.

4179 **Figure 64 — FPT\_FLS: Component leveling**



This family consists of only one component, FPT\_FLS.1 Failure with preservation of secure state, which requires that the TSF preserve a secure state in the face of the identified failures.

### 14.3.3 Management of FPT\_FLS.1

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

### 14.3.4 Audit of FPT\_FLS.1

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Basic: Failure of the TSF.

### 14.3.5 FPT\_FLS.1 Failure with preservation of secure state

#### 14.3.5.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 14.3.5.2 FPT\_FLS.1.1

The TSF **shall** preserve a secure state when the following types of failures occur: [assignment: *list of types of failures in the TSF*].

## 14.4 TSF initialization (FPT\_INI)

Editors' Note

This is a new family added according to WD2 FR/CL5.

### 14.4.1 Family behaviour

This family describes the functional requirements for the initialization of the TSF by a dedicated function of the TOE that ensures the initialization in a correct and secure operational state.

### 14.4.2 Components leveling and description

Figure 65 shows the component leveling for this family.



Figure 65 — FPT\_INI: Component leveling

This family consists of only one component, Component FPT\_INI.1. This component requires the TOE to provide a TSF initialization function that brings the TSF into a secure operational state at power-on.

### 14.4.3 Management of FPT\_INI.1

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

### 14.4.4 Audit of FPT\_INI.1

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

4214 a) There are no auditable events foreseen.

#### 4215 14.4.5 FPT\_INI.1 TSF initialization

##### 4216 14.4.5.1 Component relationships

4217 Hierarchical to: No other components.

4218 Dependencies: No dependencies.

##### 4219 14.4.5.2 FPT\_INI.1.1

4220 The TOE **shall** provide an initialization function which is intrinsically protected with  
4221 regard to the following properties [multiple selection: *integrity, authenticity, unicity*,  
4222 [assignment: *list of properties or none*].

##### 4223 14.4.5.3 FPT\_INI.1.2

4224 The TOE initialization function **shall** verify the [multiple selection: *authenticity, integrity*]  
4225 of [assignment: *list of TSF firmware, software, or data*] prior to establishing the TSF in a  
4226 secure initial state.

##### 4227 14.4.5.4 FPT\_INI.1.3

4228 The TOE initialization function **shall** detect and respond to errors and failures during  
4229 initialization such that the TOE either successfully completes initialization or is halted.

##### 4230 14.4.5.5 FPT\_INI.1.4

4231 The TOE initialization function **shall** not be able to arbitrarily interact with the TSF after  
4232 TOE initialization completes.

#### 4233 14.5 Availability of exported TSF data (FPT\_ITA)

##### 4234 14.5.1 Family behaviour

4235 This family defines the rules for the prevention of loss of availability of TSF data moving  
4236 between the TSF and another trusted IT product.

##### 4237 14.5.2 Components leveling and description

4238 Figure 66 shows the component leveling for this family.

**FPT\_ITA: Availability of exported TSF data**

**1**

4239 **Figure 66 — FPT\_ITA: Component leveling**

4240 This family consists of only one component, FPT\_ITA.1 Inter-TSF availability within a defined  
4241 availability metric. This component requires that the TSF ensure, to an identified degree of  
4242 probability, the availability of TSF data provided to another trusted IT product.

##### 4243 14.5.3 Management of FPT\_ITA.1

4244 The following actions **could** be considered for the management functions in FMT:

4245 a) management of the list of types of TSF data that must be available to another  
4246 trusted IT product.

#### 4247 14.5.4 Audit of FPT\_ITA.1

4248 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
4249 in the PP/ST:

- 4250 a) Minimal: the absence of TSF data when required by a TOE.

#### 4251 14.5.5 FPT\_ITA.1 Inter-TSF availability within a defined availability metric

##### 4252 14.5.5.1 Component relationships

4253 Hierarchical to: No other components.

4254 Dependencies: No dependencies.

##### 4255 14.5.5.2 FPT\_ITA.1.1

4256 The TSF **shall** ensure the availability of [assignment: *list of types of TSF data*] provided to  
4257 another trusted IT product within [assignment: *a defined availability metric*] given the  
4258 following conditions [assignment: *conditions to ensure availability*].

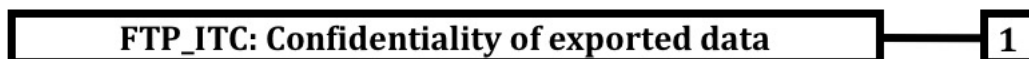
### 4259 14.6 Confidentiality of exported TSF data (FPT\_ITC)

#### 4260 14.6.1 Family behaviour

4261 This family defines the rules for the protection from unauthorized disclosure of TSF data during  
4262 transmission between the TSF and another trusted IT product.

#### 4263 14.6.2 Components leveling and description

4264 Figure 67 shows the component leveling for this family.



4265 **Figure 67 — FPT\_ITC: Component leveling**

4266 This family consists of only one component, FPT\_ITC.1 Inter-TSF confidentiality during  
4267 transmission, which requires that the TSF ensure that data transmitted between the TSF and  
4268 another trusted IT product is protected from disclosure while in transit.

#### 4269 14.6.3 Management of FPT\_ITC.1

4270 The following actions **could** be considered for the management functions in FMT:

- 4271 a) There are no management activities foreseen.

#### 4272 14.6.4 Audit of FPT\_ITC.1

4273 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
4274 in the PP/ST:

- 4275 a) There are no auditable events foreseen.

#### 4276 14.6.5 FPT\_ITC.1 Inter-TSF confidentiality during transmission

##### 4277 14.6.5.1 Component relationships

4278 Hierarchical to: No other components.

4279 Dependencies: No dependencies.

4280 **14.6.5.2 FPT\_ITC.1.1**

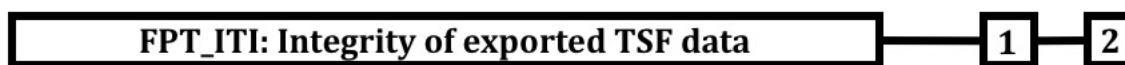
4281 **The TSF shall protect all TSF data transmitted from the TSF to another trusted IT product**  
 4282 **from unauthorized disclosure during transmission.**

4283 **14.7 Integrity of exported TSF data (FPT\_ITI)**4284 **14.7.1 Family behaviour**

4285 This family defines the rules for the protection, from unauthorized modification, of TSF data  
 4286 during transmission between the TSF and another trusted IT product.

4287 **14.7.2 Components leveling and description**

4288 Figure 68 shows the component leveling for this family.



4289 **Figure 68 — FPT\_ITI: Component leveling**

4290 FPT\_ITI.1 Inter-TSF detection of modification, provides the ability to detect modification of TSF  
 4291 data during transmission between the TSF and another trusted IT product, under the  
 4292 assumption that another trusted IT product is cognizant of the mechanism used.

4293 FPT\_ITI.2 Inter-TSF detection and correction of modification, provides the ability for another  
 4294 trusted IT product not only to detect modification, but to correct modified TSF data under the  
 4295 assumption that another trusted IT product is cognizant of the mechanism used.

4296 **14.7.3 Management of FPT\_ITI.1**

4297 The following actions could be considered for the management functions in FMT:

- 4298 a) There are no management activities foreseen.

4299 **14.7.4 Management of FPT\_ITI.2**

4300 The following actions could be considered for the management functions in FMT:

- 4301 a) Management of the types of TSF data that the TSF should try to correct if modified  
 4302 in transit;
- 4303 b) Management of the types of action that the TSF could take if TSF data is modified in  
 4304 transit.

4305 **14.7.5 Audit of FPT\_ITI.1**

4306 The following actions should be auditable if FAU\_GEN Security audit data generation is included  
 4307 in the PP/ST:

- 4308 a) Minimal: the detection of modification of transmitted TSF data.
- 4309 b) Basic: the action taken upon detection of modification of transmitted TSF data.

4310 **14.7.6 Audit of FPT\_ITI.2**

4311 The following actions should be auditable if FAU\_GEN Security audit data generation is included  
 4312 in the PP/ST:

- 4313 a) Minimal: the detection of modification of transmitted TSF data.
- 4314 b) Basic: the action taken upon detection of modification of transmitted TSF data.
- 4315 c) Basic: the use of the correction mechanism.

4316 **14.7.7 FPT\_ITI.1 Inter-TSF detection of modification**

4317 **14.7.7.1 Component relationships**

4318 Hierarchical to: No other components.

4319 Dependencies: No dependencies.

4320 **14.7.7.2 FPT\_ITI.1.1**

4321 The TSF **shall** provide the capability to detect modification of all TSF data during  
4322 transmission between the TSF and another trusted IT product within the following  
4323 metric: [assignment: *a defined modification metric*].

4324 **14.7.7.3 FPT\_ITI.1.2**

4325 The TSF **shall** provide the capability to verify the integrity of all TSF data transmitted  
4326 between the TSF and another trusted IT product and perform [assignment: *action to be*  
4327 *taken*] if modifications are detected.

4328 **14.7.8 FPT\_ITI.2 Inter-TSF detection and correction of modification**

4329 **14.7.8.1 Component relationships**

4330 Hierarchical to: FPT\_ITI.1 Inter-TSF detection of modification

4331 Dependencies: No dependencies.

4332 **14.7.8.2 FPT\_ITI.2.1**

4333 The TSF **shall** provide the capability to detect modification of all TSF data during transmission  
4334 between the TSF and another trusted IT product within the following metric: [assignment: *a*  
4335 *defined modification metric*].

4336 **14.7.8.3 FPT\_ITI.2.2**

4337 The TSF **shall** provide the capability to verify the integrity of all TSF data transmitted between  
4338 the TSF and another trusted IT product and perform [assignment: *action to be taken*] if  
4339 modifications are detected.

4340 **14.7.8.4 FPT\_ITI.2.3**

4341 The TSF **shall** provide the capability to correct [assignment: *type of modification*] of all  
4342 TSF data transmitted between the TSF and another trusted IT product.

4343 **14.8 Internal TOE TSF data transfer (FPT\_ITT)**

4344 **14.8.1 Family behaviour**

4345 This family provides requirements that address protection of TSF data when it is transferred  
4346 between separate parts of a TOE across an internal channel.

4347 **14.8.2 Components leveling and description**

4348 Figure 69 shows the component leveling for this family.



4349 **Figure 69 — FPT\_ITT: Component leveling**

4350 FPT\_ITT.1 Basic internal TSF data transfer protection, requires that TSF data be protected when  
4351 transmitted between separate parts of the TOE.

4352 FPT\_ITT.2 TSF data transfer separation, requires that the TSF separate user data from TSF data  
4353 during transmission.

4354 FPT\_ITT.3 TSF data integrity monitoring, requires that the TSF data transmitted between  
4355 separate parts of the TOE is monitored for identified integrity errors.

#### 4356 **14.8.3 Management of FPT\_ITT.1**

4357 The following actions **could** be considered for the management functions in FMT:

- 4358 a) management of the types of modification against which the TSF **should** protect;
- 4359 b) management of the mechanism used to provide the protection of the data in transit
- 4360 between different parts of the TSF.

#### 4361 **14.8.4 Management of FPT\_ITT.2**

4362 The following actions **could** be considered for the management functions in FMT:

- 4363 a) management of the types of modification against which the TSF **should** protect;
- 4364 b) management of the mechanism used to provide the protection of the data in transit
- 4365 between different parts of the TSF;
- 4366 c) management of the separation mechanism.

#### 4367 **14.8.5 Management of FPT\_ITT.3**

4368 The following actions **could** be considered for the management functions in FMT:

- 4369 a) management of the types of modification against which the TSF **should** protect;
- 4370 b) management of the mechanism used to provide the protection of the data in transit
- 4371 between different parts of the TSF;
- 4372 c) management of the types of modification of TSF data the TSF **should** try to detect;
- 4373 d) management of the actions that will be taken.

#### 4374 **14.8.6 Audit of FPT\_ITT.1, FPT\_ITT.2**

4375 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
4376 in the PP/ST:

- 4377 a) There are no auditable events foreseen.

#### 4378 **14.8.7 Audit of FPT\_ITT.3**

4379 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
4380 in the PP/ST:

- 4381 a) Minimal: the detection of modification of TSF data;
- 4382 b) Basic: the action taken following detection of an integrity error.

#### 4383 **14.8.8 FPT\_ITT.1 Basic internal TSF data transfer protection**

##### 4384 **14.8.8.1 Component relationships**

- 4385 Hierarchical to: No other components.
- 4386 Dependencies: No dependencies.

4387 **14.8.8.2 FPT\_ITT.1.1**

4388 The TSF **shall** protect TSF data from [selection: *disclosure, modification*] when it is  
 4389 transmitted between separate parts of the TOE.

4390 **14.8.9 FPT\_ITT.2 TSF data transfer separation**4391 **14.8.9.1 Component relationships**

4392 Hierarchical to: FPT\_ITT.1 Basic internal TSF data transfer  
 4393 protection

4394 Dependencies: No dependencies.

4395 **14.8.9.2 FPT\_ITT.2.1**

4396 The TSF **shall** protect TSF data from [selection: *disclosure, modification*] when it is transmitted  
 4397 between separate parts of the TOE.

4398 **14.8.9.3 FPT\_ITT.2.2**

4399 The TSF **shall** separate user data from TSF data when such data is transmitted between  
 4400 separate parts of the TOE.

4401 **14.8.10 FPT\_ITT.3 TSF data integrity monitoring**4402 **14.8.10.1 Component relationships**

4403 Hierarchical to: No other components.

4404 Dependencies: FPT\_ITT.1 Basic internal TSF data transfer  
 4405 protection

4406 **14.8.10.2 FPT\_ITT.3.1**

4407 The TSF **shall** be able to detect [selection: *modification of data, substitution of data, re-*  
 4408 *ordering of data, deletion of data, [assignment: other integrity errors]] for TSF data*  
 4409 *transmitted between separate parts of the TOE.*

4410 **14.8.10.3 FPT\_ITT.3.2**

4411 Upon detection of a data integrity error, the TSF **shall** take the following actions:  
 4412 [assignment: *specify the action to be taken*].

4413 **14.9 TSF physical protection (FPT\_PHP)**4414 **14.9.1 Family behaviour**

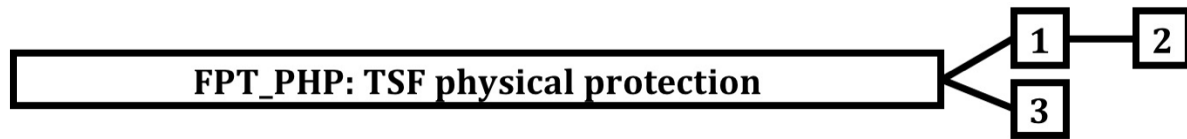
4415 TSF physical protection components refer to restrictions on unauthorized physical access to the  
 4416 TSF, and to the deterrence of, and resistance to, unauthorized physical modification, or  
 4417 substitution of the TSF.

4418 The requirements of components in this family ensure that the TSF is protected from physical  
 4419 tampering and interference. Satisfying the requirements of these components results in the TSF  
 4420 being packaged and used in such a manner that physical tampering is detectable, or resistance  
 4421 to physical tampering is enforced. Without these components, the protection functions of a TSF  
 4422 lose their effectiveness in environments where physical damage cannot be prevented. This  
 4423 family also provides requirements regarding how the TSF **shall** respond to physical tampering  
 4424 attempts.



**14.9.2 Components leveling and description**

Figure 70 shows the component leveling for this family.



**Figure 70 — FPT\_PHP: Component leveling**

FPT\_PHP.1 Passive detection of physical attack, provides for features that indicate when a TSF device or TSF element is subject to tampering. However, notification of tampering is not automatic; an authorized user must invoke a security administrative function or perform manual inspection to determining if tampering has occurred.

FPT\_PHP.2 Notification of physical attack, provides for automatic notification of tampering for an identified subset of physical penetrations.

FPT\_PHP.3 Resistance to physical attack, provides for features that prevent or resist physical tampering with TSF devices and TSF elements.

**14.9.3 Management of FPT\_PHP.1**

The following actions **could** be considered for the management functions in FMT:

- a) Management of the user or role that determines whether physical tampering has occurred.

**14.9.4 Management of FPT\_PHP.2**

The following actions **could** be considered for the management functions in FMT:

- a) Management of the user or role that gets informed about intrusions;
- b) Management of the list of devices that **should** inform the indicated user or role about the intrusion.

**14.9.5 Management of FPT\_PHP.3**

The following actions **could** be considered for the management functions in FMT:

- a) Management of the automatic responses to physical tampering.

**14.9.6 Audit of FPT\_PHP.1**

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: if detection by IT means, detection of intrusion.

**14.9.7 Audit of FPT\_PHP.2**

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: detection of intrusion.

**14.9.8 Audit of FPT\_PHP.3**

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) There are no auditable events foreseen.



4460 **14.9.9 FPT\_PHP.1 Passive detection of physical attack**

4461 **14.9.9.1 Component relationships**

4462 Hierarchical to: No other components.

4463 Dependencies: No dependencies.

4464 **14.9.9.2 FPT\_PHP.1.1**

4465 The TSF **shall** provide unambiguous detection of physical tampering that might  
4466 compromise the TSF.

4467 **14.9.9.3 FPT\_PHP.1.2**

4468 The TSF **shall** provide the capability to determine whether physical tampering with the  
4469 TSF's devices or TSF's elements has occurred.

4470 **14.9.10 FPT\_PHP.2 Notification of physical attack**

4471 **14.9.10.1 Component relationships**

4472 Hierarchical to: FPT\_PHP.1 Passive detection of physical attack

4473 Dependencies: FMT\_LIM.1 Limited capabilities

4474 **14.9.10.2 FPT\_PHP.2.1**

4475 The TSF **shall** provide unambiguous detection of physical tampering that might compromise the  
4476 TSF.

4477 **14.9.10.3 FPT\_PHP.2.2**

4478 The TSF **shall** provide the capability to determine whether physical tampering with the TSF's  
4479 devices or TSF's elements has occurred.

4480 **14.9.10.4 FPT\_PHP.2.3**

4481 For [assignment: *list of TSF devices/elements for which active detection is required*], the  
4482 TSF **shall** monitor the devices and elements and notify [assignment: *a designated user or*  
4483 *role*] when physical tampering with the TSF's devices or TSF's elements has occurred.

4484 **14.9.11 FPT\_PHP.3 Resistance to physical attack**

4485 **14.9.11.1 Component relationships**

4486 Hierarchical to: No other components.

4487 Dependencies: No dependencies.

4488 **14.9.11.2 FPT\_PHP.3.1**

4489 The TSF **shall** resist [assignment: *physical tampering scenarios*] to the [assignment: *list of*  
4490 *TSF devices/elements*] by responding automatically such that the SFRs are always  
4491 enforced.

4492 **14.10 Trusted recovery (FPT\_RCV)**

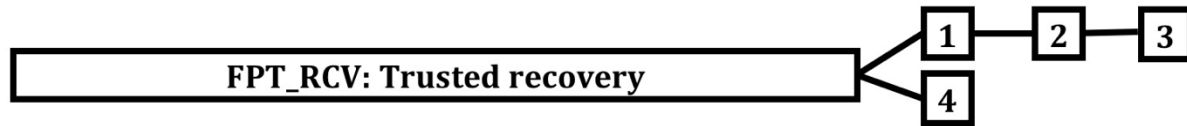
4493 **14.10.1 Family behaviour**

4494 The requirements of this family ensure that the TSF **can** determine that the TOE is started up  
4495 without protection compromise and **can** recover without protection compromise after

4496 discontinuity of operations. This family is important because the start-up state of the TSF  
4497 determines the protection of subsequent states.

#### 4498 **14.10.2 Components leveling and description**

4499 Figure 71 shows the component leveling for this family.



4500 **Figure 71 — FPT\_RCV: Component leveling**

4501 FPT\_RCV.1 Manual recovery, allows a TOE to only provide mechanisms that involve human  
4502 intervention to return to a secure state.

4503 FPT\_RCV.2 Automated recovery, provides, for at least one type of service discontinuity,  
4504 recovery to a secure state without human intervention; recovery for other discontinuities **may**  
4505 **can** require human intervention.

4506 FPT\_RCV.3 Automated recovery without undue loss, also provides for automated recovery, but  
4507 strengthens the requirements by disallowing undue loss of protected objects.

4508 FPT\_RCV.4 Function recovery, provides for recovery at the level of particular functions,  
4509 ensuring either successful completion or rollback of TSF data to a secure state.

#### 4510 **14.10.3 Management of FPT\_RCV.1**

4511 The following actions **could** be considered for the management functions in FMT:

- 4512 a) Management of who **can** access the restore capability within the maintenance  
4513 mode.

#### 4514 **14.10.4 Management of FPT\_RCV.2, FPT\_RCV.3**

4515 The following actions **could** be considered for the management functions in FMT:

- 4516 a) Management of who **can** access the restore capability within the maintenance  
4517 mode;
- 4518 b) Management of the list of failures/service discontinuities that will be handled  
4519 through the automatic procedures.

#### 4520 **14.10.5 Management of FPT\_RCV.4**

4521 The following actions **could** be considered for the management functions in FMT:

- 4522 a) There are no management activities foreseen.

#### 4523 **14.10.6 Audit of FPT\_RCV.1, FPT\_RCV.2, FPT\_RCV.3**

4524 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
4525 in the PP/ST:

- 4526 a) Minimal: the fact that a failure or service discontinuity occurred;
- 4527 b) Minimal: resumption of the regular operation;
- 4528 c) Basic: type of failure or service discontinuity.

#### 4529 **14.10.7 Audit of FPT\_RCV.4**

4530 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
4531 in the PP/ST:

4532 a) Minimal: if possible, the impossibility to return to a secure state after a failure of  
4533 the TSF;

4534 b) Basic: if possible, the detection of a failure of a function.

#### 4535 **14.10.8 FPT\_RCV.1 Manual recovery**

##### 4536 **14.10.8.1 Component relationships**

4537 Hierarchical to: No other components.

4538 Dependencies: AGD\_OPE.1 Operational user guidance

##### 4539 **14.10.8.2 FPT\_RCV.1.1**

4540 **After [assignment: *list of failures/service discontinuities*] the TSF shall enter a**  
4541 **maintenance mode where the ability to return to a secure state is provided.**

#### 4542 **14.10.8.3 FPT\_RCV.2 Automated recovery**

##### 4543 **14.10.8.4 Component relationships**

4544 Hierarchical to: FPT\_RCV.1 Manual recovery

4545 Dependencies: AGD\_OPE.1 Operational user guidance

##### 4546 **14.10.8.5 FPT\_RCV.2.1**

4547 **When automated recovery from [assignment: *list of failures/service discontinuities*] is not**  
4548 **possible, the TSF shall enter a maintenance mode where the ability to return to a secure state is**  
4549 **provided.**

##### 4550 **14.10.8.6 FPT\_RCV.2.2**

4551 **For [assignment: *list of failures/service discontinuities*], the TSF shall ensure the return of**  
4552 **the TOE to a secure state using automated procedures.**

#### 4553 **14.10.9 FPT\_RCV.3 Automated recovery without undue loss**

##### 4554 **14.10.9.1 Component relationships**

4555 Hierarchical to: FPT\_RCV.2 Automated recovery

4556 Dependencies: AGD\_OPE.1 Operational user guidance

##### 4557 **14.10.9.2 FPT\_RCV.3.1**

4558 When automated recovery from [assignment: *list of failures/service discontinuities*] is not  
4559 possible, the TSF shall enter a maintenance mode where the ability to return to a secure state is  
4560 provided.

##### 4561 **14.10.9.3 FPT\_RCV.3.2**

4562 For [assignment: *list of failures/service discontinuities*], the TSF shall ensure the return of the  
4563 TOE to a secure state using automated procedures.

##### 4564 **14.10.9.4 FPT\_RCV.3.3**

4565 **The functions provided by the TSF to recover from failure or service discontinuity shall**  
4566 **ensure that the secure initial state is restored without exceeding [assignment:**  
4567 ***quantification*] for loss of TSF data or objects under the control of the TSF.**

4568 **14.10.9.5 FPT\_RCV.3.4**

4569 The TSF **shall** provide the capability to determine the objects that were or were not  
 4570 capable of being recovered.

4571 **14.10.10 FPT\_RCV.4 Function recovery**4572 **14.10.10.1 Component relationships**

4573 Hierarchical to: No other components.

4574 Dependencies: No dependencies.

4575 **14.10.10.2 FPT\_RCV.4.1**

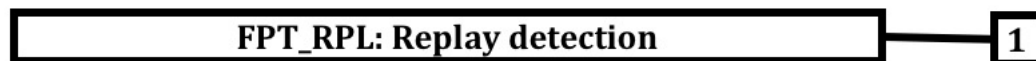
4576 The TSF **shall** ensure that [assignment: *list of functions and failure scenarios*] have the  
 4577 property that the function either completes successfully, or for the indicated failure  
 4578 scenarios, recovers to a consistent and secure state.

4579 **14.11 Replay detection (FPT\_RPL)**4580 **14.11.1 Family behaviour**

4581 This family addresses detection of replay for various types of entities and subsequent actions to  
 4582 correct. In the case where replay **may** be detected, this effectively prevents it.

4583 **14.11.2 Components leveling and description**

4584 Figure 72 shows the component leveling for this family.



4585 **Figure 72 — FPT\_RPL: Component leveling**

4586 The family consists of only one component, FPT\_RPL.1 Replay detection, which requires that  
 4587 the TSF **shall** be able to detect the replay of identified entities.

4588 **14.11.3 Management of FPT\_RPL.1**

4589 The following actions **could** be considered for the management functions in FMT:

- 4590 a) Management of the list of identified entities for which replay **shall must** be  
 4591 detected;
- 4592 b) Management of the list of actions that need to be taken in case of replay.

4593 **14.11.4 Audit of FPT\_RPL.1**

4594 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 4595 in the PP/ST:

- 4596 a) Basic: Detected replay attacks.
- 4597 b) Detailed: Action to be taken based on the specific actions.

4598 **14.11.5 FPT\_RPL.1 Replay detection**4599 **14.11.5.1 Component relationships**

4600 Hierarchical to: No other components.

4601 Dependencies: No dependencies.

#### 14.11.5.2 FPT\_RPL.1.1

The TSF **shall** detect replay for the following entities: [assignment: *list of identified entities*].

#### 14.11.5.3 FPT\_RPL.1.2

The TSF **shall** perform [assignment: *list of specific actions*] when replay is detected.

### 14.12 State synchrony protocol (FPT\_SSP)

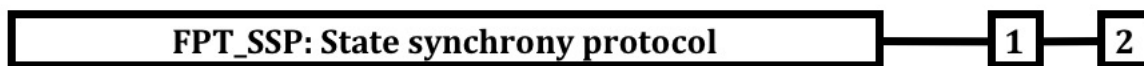
#### 14.12.1 Family behaviour

Distributed TOEs **may-can** give rise to greater complexity than monolithic TOEs through the potential for differences in state between parts of the TOE, and through delays in communication. In most cases synchronization of state between distributed functions involves an exchange protocol, not a simple action. When malice exists in the distributed environment of these protocols, more complex defensive protocols are required.

State synchrony protocol (FPT\_SSP) establishes the requirement for certain critical functions of the TSF to use this trusted protocol. State synchrony protocol (FPT\_SSP) ensures that two distributed parts of the TOE have synchronized their states after a security-relevant action.

#### 14.12.2 Components leveling and description

Figure 73 shows the component leveling for this family.



**Figure 73 — FPT\_SSP: Component leveling**

FPT\_SSP.1 Simple trusted acknowledgement, requires only a simple acknowledgment by the data recipient.

FPT\_SSP.2 Mutual trusted acknowledgement, requires mutual acknowledgment of the data exchange.

#### 14.12.3 Management of FPT\_SSP.1, FPT\_SSP.2

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

#### 14.12.4 Audit of FPT\_SSP.1, FPT\_SSP.2

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: failure to receive an acknowledgement when expected.

#### 14.12.5 FPT\_SSP.1 Simple trusted acknowledgement

##### 14.12.5.1 Component relationships

Hierarchical to:	No other components.
Dependencies:	FPT_ITT.1 Basic internal TSF data transfer protection

4636 **14.12.5.2 FPT\_SSP.1.1**

4637 The TSF **shall** acknowledge, when requested by another part of the TSF, the receipt of an  
 4638 unmodified TSF data transmission.

4639 **14.12.6 FPT\_SSP.2 Mutual trusted acknowledgement**4640 **14.12.6.1 Component relationships**

4641 Hierarchical to: FPT\_SSP.1 Simple trusted acknowledgement

4642 Dependencies: FPT\_ITT.1 Basic internal TSF data transfer  
 4643 protection

4644 **14.12.6.2 FPT\_SSP.2.1**

4645 The TSF **shall** acknowledge, when requested by another part of the TSF, the receipt of an  
 4646 unmodified TSF data transmission.

4647 **14.12.6.3 FPT\_SSP.2.2**

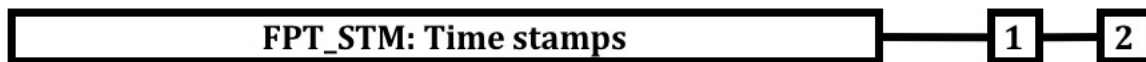
4648 The TSF **shall** ensure that the relevant parts of the TSF know the correct status of  
 4649 transmitted data among its different parts, using acknowledgements.

4650 **14.13 Time stamps (FPT\_STM)**4651 **14.13.1 Family behaviour**

4652 This family addresses requirements for a reliable time stamp function within a TOE.

4653 **14.13.2 Components leveling and description**

4654 Figure 74 shows the component leveling for this family.



4655 **Figure 74 — FPT\_STM: Component leveling**

4656 FPT\_STM.1 Reliable time stamps, requires that the TSF provide reliable time stamps for TSF  
 4657 functions.

4658 FPT\_STM.2 Time source, requires the description of the time source used in timestamps

4659 **14.13.3 Management of FPT\_STM.1**

4660 The following actions **could** be considered for the management functions in FMT:

4661 a) Management of the time.

4662 **14.13.4 Management of FPT\_STM.2**

4663 The following actions **could** be considered for the management functions in FMT:

4664 a) Setting of time by user authorized according to security policy.

4665 **14.13.5 Audit of FPT\_STM.1**

4666 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 4667 in the PP/ST:

4668 a) Minimal: changes to the time.

4669 b) Detailed: providing a timestamp.

**14.13.6 Audit of FPT\_STM.2**

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: discontinuous changes to the time;
- b) Detailed: changes to the time source.

**14.13.7 FPT\_STM.1 Reliable time stamps****14.13.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**14.13.7.2 FPT\_STM.1.1**

The TSF **shall** be able to provide reliable time stamps.

**14.13.8 FPT\_STM.2 Time source****14.13.8.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FPT\_STM.1 Reliable time stamps

FMT\_SMR.1 Security roles

**14.13.8.2 FPT\_STM.2.1**

The TSF **shall** allow the [assignment: *user authorized by security policy*] to [assignment: *set the time, configure another time source*].

**14.14 Inter-TSF TSF data consistency (FPT\_TDC)****14.14.1 Family behaviour**

In a distributed environment, a TOE **may** need to exchange TSF data with another trusted IT product. This family defines the requirements for sharing and consistent interpretation of these attributes between the TSF of the TOE and a different trusted IT product.

**14.14.2 Components leveling and description**

Figure 75 shows the component leveling for this family.

<b>FPT_TDC: Inter-TSF TSF data consistency</b>	<b>1</b>
--	----------

**Figure 75 — FPT\_TDC: Component leveling**

FPT\_TDC.1 Inter-TSF basic TSF data consistency, requires that the TSF provide the capability to ensure consistency of attributes between TSFs.

**14.14.3 Management of FPT\_TDC.1**

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

4702 **14.14.4 Audit of FPT\_TDC.1**

4703 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 4704 in the PP/ST:

- 4705 a) Minimal: Successful use of TSF data consistency mechanisms.
- 4706 b) Basic: Use of the TSF data consistency mechanisms.
- 4707 c) Basic: Identification of which TSF data have been interpreted.
- 4708 d) Basic: Detection of modified TSF data.

4709 **14.14.5 FPT\_TDC.1 Inter-TSF basic TSF data consistency**4710 **14.14.5.1 Component relationships**

- |      |                  |                      |
|------|------------------|----------------------|
| 4711 | Hierarchical to: | No other components. |
| 4712 | Dependencies:    | No dependencies.     |

4713 **14.14.5.2 FPT\_TDC.1.1**

4714 The TSF **shall** provide the capability to consistently interpret [assignment: *list of TSF data*  
 4715 *types*] when shared between the TSF and another trusted IT product.

4716 **14.14.5.3 FPT\_TDC.1.2**

4717 The TSF **shall** use [assignment: *list of interpretation rules to be applied by the TSF*] when  
 4718 interpreting the TSF data from another trusted IT product.

4719 **14.15 Testing of external entities (FPT\_TEE)**4720 **14.15.1 Family behaviour**

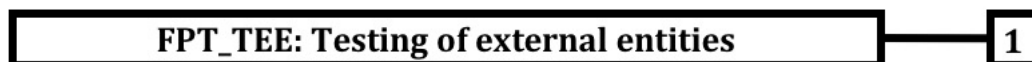
4721 This family defines requirements for the TSF to perform tests on one or more external entities.

4722 This component is not intended to be applied to human users.

4723 External entities **may can** include applications running on the TOE, hardware or software  
 4724 running “underneath” the TOE (platforms, operating systems etc.) or applications/boxes  
 4725 connected to the TOE (intrusion detection systems, firewalls, login servers, time servers etc.).

4726 **14.15.2 Components leveling and description**

4727 Figure 76 shows the component leveling for this family.



4728 **Figure 76 — FPT\_TEE: Component leveling**

4729 FPT\_TEE.1 Testing of external entities, provides for testing of the external entities by the TSF.

4730 **14.15.3 Management of FPT\_TEE.1**

4731 The following actions **could** be considered for the management functions in FMT:

- 4732 a) Management of the conditions under which the testing of external entities occurs,  
 4733 such as during initial start-up, regular interval, or under specified conditions;
- 4734 b) Management of the time interval if appropriate.



#### 14.15.4 Audit of FPT\_TEE.1

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Basic: Execution of the tests of the external entities and the results of the tests.

#### 14.15.5 FPT\_TEE.1 Testing of external entities

##### 14.15.5.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

##### 14.15.5.2 FPT\_TEE.1.1

The TSF **shall** run a suite of tests [selection: *during initial start-up, periodically during normal operation, at the request of an authorized user, [assignment: other conditions]*] to check the fulfilment of [assignment: *list of properties of the external entities*].

##### 14.15.5.3 FPT\_TEE.1.2

If the test fails, the TSF **shall** [assignment: *action(s)*].

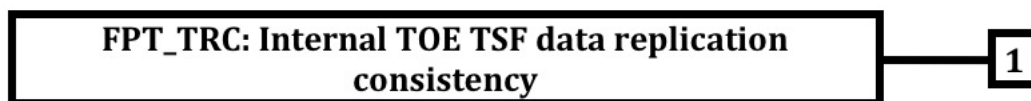
#### 14.16 Internal TOE TSF data replication consistency (FPT\_TRC)

##### 14.16.1 Family behaviour

The requirements of this family are needed to ensure the consistency of TSF data when such data is replicated internal to the TOE. Such data **may** become inconsistent if the internal channel between parts of the TOE becomes inoperative. If the TOE is internally structured as a network and parts of the TOE network connections are broken, this **may** occur when parts become disabled.

##### 14.16.2 Components leveling and description

Figure 77 shows the component leveling for this family.



**Figure 77 — FPT\_TRC: Component leveling**

This family consists of only one component, FPT\_TRC.1 Internal TSF consistency, which requires that the TSF ensure the consistency of TSF data that is replicated in multiple locations.

##### 14.16.3 Management of FPT\_TRC.1

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

##### 14.16.4 Audit of FPT\_TRC.1

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

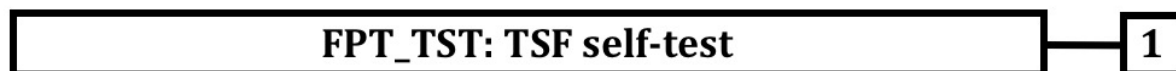
- a) Minimal: restoring consistency upon reconnection;
- b) Basic: Detected inconsistency between TSF data.

4769 **14.16.5 FPT\_TRC.1 Internal TSF consistency**4770 **14.16.5.1 Component relationships**

4771 Hierarchical to: No other components.

4772 Dependencies: FPT\_ITT.1 Basic internal TSF data transfer  
4773 protection4774 **14.16.5.2 FPT\_TRC.1.1**4775 **The TSF shall ensure that TSF data is consistent when replicated between parts of the**  
4776 **TOE.**4777 **14.16.5.3 FPT\_TRC.1.2**4778 **When parts of the TOE containing replicated TSF data are disconnected, the TSF shall**  
4779 **ensure the consistency of the replicated TSF data upon reconnection before processing**  
4780 **any requests for [assignment: *list of functions dependent on TSF data replication***  
4781 **consistency].**4782 **14.17 TSF self-test (FPT\_TST)**4783 **14.17.1 Family behaviour**4784 The family defines the requirements for the self-testing of the TSF with respect to some  
4785 expected correct operation. Examples are interfaces to enforcement functions, and sample  
4786 arithmetical operations on critical parts of the TOE. These tests can be carried out at start-up,  
4787 periodically, at the request of the authorized user, or when other conditions are met. The  
4788 actions to be taken by the TOE as the result of self-testing are defined in other families.4789 The requirements of this family are also needed to detect the corruption of TSF data and TSF  
4790 itself (i.e. TSF executable code or TSF hardware component) by various failures that do not  
4791 necessarily stop the TOE's operation (which would be handled by other families). These checks  
4792 must be performed because these failures may not cannot necessarily be prevented. Such  
4793 failures can occur either because of unforeseen failure modes or associated oversights in the  
4794 design of hardware, firmware, or software, or because of malicious corruption of the TSF due to  
4795 inadequate logical and/or physical protection.4796 **14.17.2 Components leveling and description**

4797 Figure 78 shows the component leveling for this family.

4798 **Figure 78 — FPT\_TST: Component leveling**4799 FPT\_TST.1 TSF self-testing, provides the ability to test the TSF's correct operation. These tests  
4800 may can be performed at start-up, periodically, at the request of the authorized user, or when  
4801 other conditions are met. It also provides the ability to verify the integrity of TSF data and TSF  
4802 itself.4803 **14.17.3 Management of FPT\_TST.1**

4804 The following actions could be considered for the management functions in FMT:

- 4805 a) Management of the conditions under which TSF self-testing occurs, such as during
- 4806 initial start-up, regular interval, or under specified conditions;
- 4807 b) Management of the time interval if appropriate.

4808 **14.17.4 Audit of FPT\_TST.1**

4809 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 4810 in the PP/ST:

- 4811 a) Minimal: Indication that the TSF self-tests were completed and any failures of the
- 4812 tests.
- 4813 b) Basic: Execution of the TSF self-tests and the results of the tests.

4814 **14.17.5 FPT\_TST.1 TSF self-testing**4815 **14.17.5.1 Component relationships**

4816 Hierarchical to: No other components.

4817 Dependencies: No dependencies.

4818 **14.17.5.2 FPT\_TST.1.1**

4819 The TSF **shall** run a suite of the following self-tests [selection: *during initial start-up,*  
 4820 *periodically during normal operation, at the request of the authorized user, at the*  
 4821 *conditions [assignment: conditions under which self-test **should** occur]* to demonstrate the  
 4822 correct operation of [selection: *[assignment: parts of TSF], the TSF*]: [assignment: *list of*  
 4823 *self-tests run by the TSF*].

4824 **14.17.5.3 FPT\_TST.1.2**

4825 The TSF **shall** provide authorized users with the capability to verify the integrity of  
 4826 [selection: *[assignment: parts of TSF data], TSF data*].

4827 **14.17.5.4 FPT\_TST.1.3**

4828 The TSF **shall** provide authorized users with the capability to verify the integrity of  
 4829 [selection: *[assignment: parts of TSF], TSF*].

4830

15 Class FRU: Resource utilization

15.1 Class description

This class provides three families that support the availability of required resources such as processing capability and/or storage capacity. The family Fault Tolerance provides protection against unavailability of capabilities caused by failure of the TOE. The family Priority of Service ensures that the resources will be allocated to the more important or time-critical tasks and cannot be monopolized by lower priority tasks. The family Resource Allocation provides limits on the use of available resources, therefore preventing users from monopolizing the resources.

Figure 79 shows the decomposition of this class, it's families and components. Elements are not shown in the figure.

Annex K provides explanatory information for this class and **should** be consulted when using the components identified in this class.

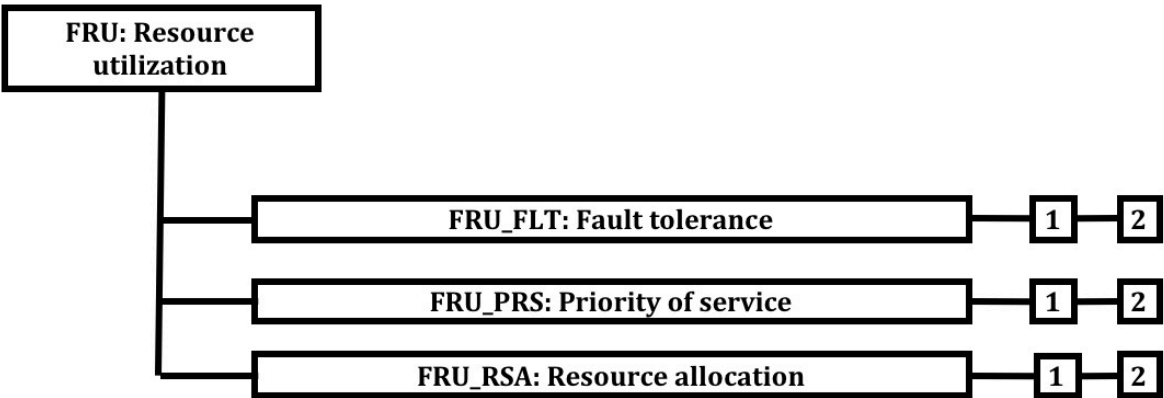


Figure 79 — FRU: Resource utilization class decomposition

15.2 Fault tolerance (FRU\_FLT)

15.2.1 Family behaviour

The requirements of this family ensure that the TOE will maintain correct operation even in the event of failures.

15.2.2 Components leveling and description

Figure 80 shows the component leveling for this family.



Figure 80 — FRU\_FLT: Component leveling

FRU\_FLT.1 Degraded fault tolerance, requires the TOE to continue correct operation of identified capabilities in the event of identified failures.

FRU\_FLT.2 Limited fault tolerance, requires the TOE to continue correct operation of all capabilities in the event of identified failures.

### 15.2.3 Management of FRU\_FLT.1, FRU\_FLT.2

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

### 15.2.4 Audit of FRU\_FLT.1

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: Any failure detected by the TSF.
- b) Basic: All TOE capabilities being discontinued due to a failure.

### 15.2.5 Audit of FRU\_FLT.2

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: Any failure detected by the TSF.

### 15.2.6 FRU\_FLT.1 Degraded fault tolerance

#### 15.2.6.1 Component relationships

Hierarchical to:	No other components.
Dependencies:	FPT_FLS.1 Failure with preservation of secure state

#### 15.2.6.2 FRU\_FLT.1.1

The TSF **shall** ensure the operation of [assignment: *list of TOE capabilities*] when the following failures occur: [assignment: *list of type of failures*].

### 15.2.7 FRU\_FLT.2 Limited fault tolerance

#### 15.2.7.1 Component relationships

Hierarchical to:	FRU_FLT.1 Degraded fault tolerance
Dependencies:	FPT_FLS.1 Failure with preservation of secure state

#### 15.2.7.2 FRU\_FLT.2.1

The TSF **shall** ensure the operation of **all the TOE's capabilities** when the following failures occur: [assignment: *list of type of failures*].

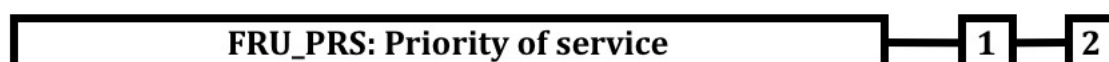
## 15.3 Priority of service (FRU\_PRS)

### 15.3.1 Family behaviour

The requirements of this family allow the TSF to control the use of resources under the control of the TSF by users and subjects such that high priority activities under the control of the TSF will always be accomplished without undue interference or delay caused by low priority activities.

### 15.3.2 Components leveling and description

Figure 81 shows the component leveling for this family.

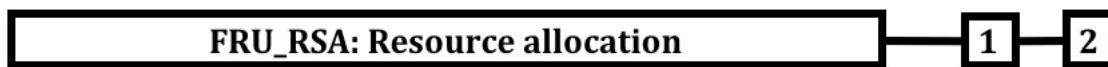


**Figure 81 — FRU\_PRS: Component leveling**

- 4891
- 4892 FRU\_PRS.1 Limited priority of service, provides priorities for a subject's use of a subset of the  
4893 resources under the control of the TSF.
- 4894 FRU\_PRS.2 Full priority of service, provides priorities for a subject's use of all of the resources  
4895 under the control of the TSF.
- 4896 **15.3.3 Management of FRU\_PRS.1, FRU\_PRS.2**
- 4897 The following actions **could** be considered for the management functions in FMT:
- 4898 a) Assignment of priorities to each subject in the TSF.
- 4899 **15.3.4 Audit of FRU\_PRS.1, FRU\_PRS.2**
- 4900 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
4901 in the PP/ST:
- 4902 a) Minimal: Rejection of operation based on the use of priority within an allocation.
- 4903 b) Basic: All attempted uses of the allocation function which involves the priority of  
4904 the service functions.
- 4905 **15.3.5 FRU\_PRS.1 Limited priority of service**
- 4906 Hierarchical to: No other components.
- 4907 Dependencies: No dependencies.
- 4908 **15.3.5.1 FRU\_PRS.1.1**
- 4909 The TSF **shall** assign a priority to each subject in the TSF.
- 4910 **15.3.5.2 FRU\_PRS.1.2**
- 4911 The TSF **shall** ensure that each access to [assignment: *controlled resources*] **shall** be  
4912 mediated on the basis of the subjects assigned priority.
- 4913 **15.3.6 FRU\_PRS.2 Full priority of service**
- 4914 **15.3.6.1 Component relationships**
- 4915 Hierarchical to: FRU\_PRS.1 Limited priority of service
- 4916 Dependencies: No dependencies.
- 4917 **15.3.6.2 FRU\_PRS.2.1**
- 4918 The TSF **shall** assign a priority to each subject in the TSF.
- 4919 **15.3.6.3 FRU\_PRS.2.2**
- 4920 The TSF **shall** ensure that each access to **all shareable resources** **shall** be mediated on the  
4921 basis of the subjects assigned priority.
- 4922 **15.4 Resource allocation (FRU\_RSA)**
- 4923 **15.4.1 Family behaviour**
- 4924 The requirements of this family allow the TSF to control the use of resources by users and  
4925 subjects such that denial of service will not occur because of unauthorized monopolization of  
4926 resources.

## 15.4.2 Components leveling and description

Figure 82 shows the component leveling for this family.



**Figure 82 — FRU\_RSA: Component leveling**

FRU\_RSA.1 Maximum quotas, provides requirements for quota mechanisms that ensure that users and subjects will not monopolize a controlled resource.

FRU\_RSA.2 Minimum and maximum quotas, provides requirements for quota mechanisms that ensure that users and subjects will always have at least a minimum of a specified resource and that they will not be able to monopolize a controlled resource.

## 15.4.3 Management of FRU\_RSA.1

The following actions **could** be considered for the management functions in FMT:

- a) Specifying maximum limits for a resource for groups and/or individual users and/or subjects by an administrator.

## 15.4.4 Management of FRU\_RSA.2

The following actions **could** be considered for the management functions in FMT:

- a) Specifying minimum and maximum limits for a resource for groups and/or individual users and/or subjects by an administrator.

## 15.4.5 Audit of FRU\_RSA.1, FRU\_RSA.2

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) Minimal: Rejection of allocation operation due to resource limits.
- b) Basic: All attempted uses of the resource allocation functions for resources that are under control of the TSF.

## 15.4.6 FRU\_RSA.1 Maximum quotas

### 15.4.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

### 15.4.6.2 FRU\_RSA.1.1

The TSF **shall** enforce maximum quotas of the following resources: [assignment: *controlled resources*] that [selection: *individual user, defined group of users, subjects*] **can** use [selection: *simultaneously, over a specified period of time*].

## 15.4.7 FRU\_RSA.2 Minimum and maximum quotas

### 15.4.7.1 Component relationships

Hierarchical to: FRU\_RSA.1 Maximum quotas

Dependencies: No dependencies.

4961 **15.4.7.2 FRU\_RSA.2.1**

4962 The TSF **shall** enforce maximum quotas of the following resources [assignment: *controlled*  
4963 *resources*] that [selection: *individual user, defined group of users, subjects*] **can** use [selection:  
4964 *simultaneously, over a specified period of time*].

4965 **15.4.7.3 FRU\_RSA.2.2**

4966 The TSF **shall** ensure the provision of minimum quantity of each [assignment: *controlled*  
4967 *resource*] that is available for [selection: *an individual user, defined group of users,*  
4968 *subjects*] to use [selection: *simultaneously, over a specified period of time*].

4969



## 16 Class FTA: TOE access

### 16.1 Class description

This family specifies functional requirements for controlling the establishment of a user's session.

Figure 83 shows the decomposition of this class, it's families and components. Elements are not shown in the figure.

Annex L provides explanatory information for this class and **should** be consulted when using the components identified in this class.

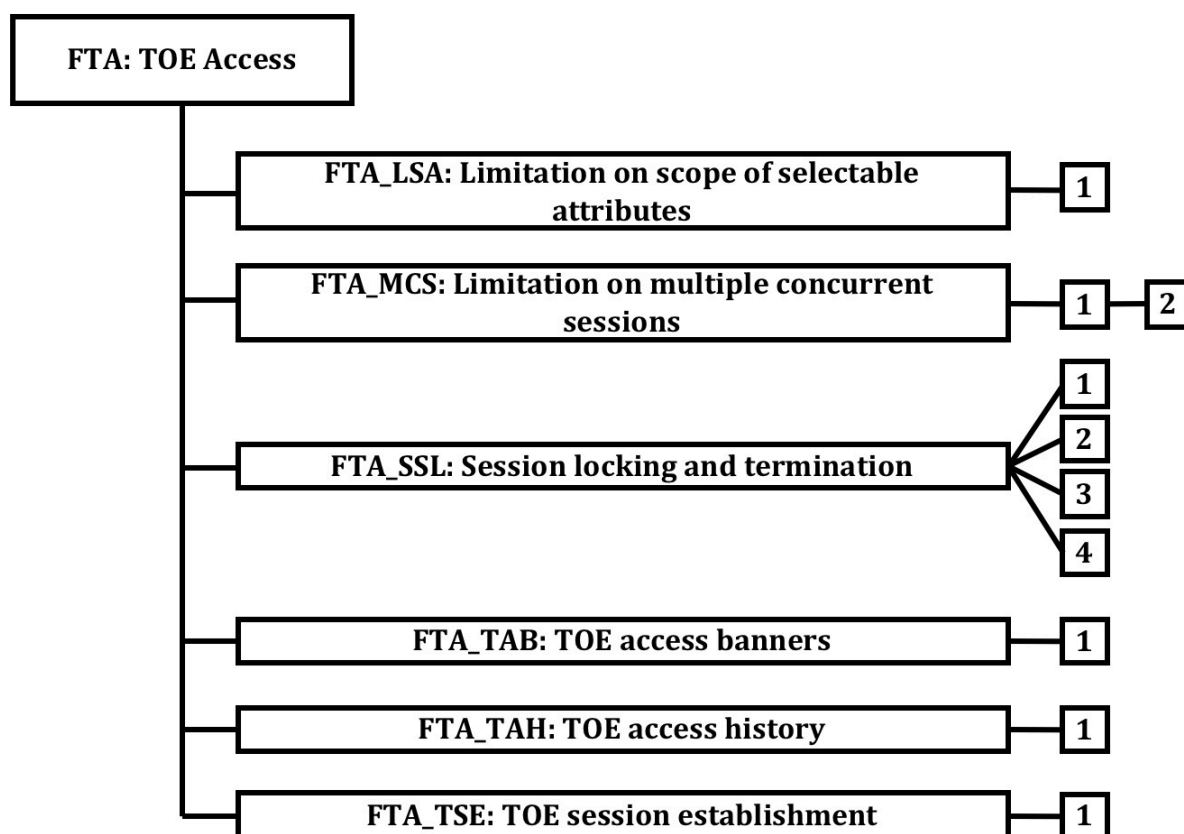


Figure 83 — FTA: TOE access class decomposition

### 16.2 Limitation on scope of selectable attributes (FTA\_LSA)

#### 16.2.1 Family behaviour

This family defines requirements to limit the scope of session security attributes that a user **may can** select for a session.

#### 16.2.2 Components leveling and description

Figure 84 shows the component leveling for this family.

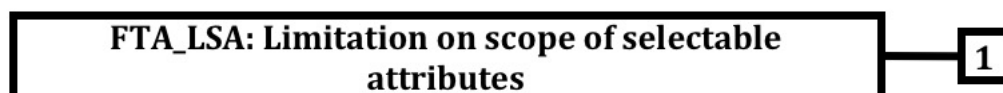


Figure 84 — FTA\_LSA: Component leveling

4986 FTA\_LSA.1 Limitation on scope of selectable attributes, provides the requirement for a TOE to  
4987 limit the scope of the session security attributes during session establishment.

### 4988 **16.2.3 Management of FTA\_LSA.1**

4989 The following actions **could** be considered for the management functions in FMT:

4990 a) Management of the scope of the session security attributes by an administrator.

### 4991 **16.2.4 Audit of FTA\_LSA.1**

4992 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
4993 in the PP/ST:

4994 a) Minimal: All failed attempts at selecting a session security attributes.

4995 b) Basic: All attempts at selecting a session security attributes.

4996 c) Detailed: Capture of the values of each session security attributes.

### 4997 **16.2.5 FTA\_LSA.1 Limitation on scope of selectable attributes**

#### 4998 **16.2.5.1 Component relationships**

4999 Hierarchical to: No other components.

5000 Dependencies: No dependencies.

#### 5001 **16.2.5.2 FTA\_LSA.1.1**

5002 The TSF **shall** restrict the scope of the session security attributes [assignment: *session*  
5003 *security attributes*], based on [assignment: *attributes*].

### 5004 **16.3 Limitation on multiple concurrent sessions (FTA\_MCS)**

#### 5005 **16.3.1 Family behaviour**

5006 This family defines requirements to place limits on the number of concurrent sessions that  
5007 belong to the same user.

#### 5008 **16.3.2 Components leveling and description**

5009 Figure 85 shows the component leveling for this family.



5010 **Figure 85 — FTA\_MCS: Component leveling**

5011 FTA\_MCS.1 Basic limitation on multiple concurrent sessions, provides limitations that apply to  
5012 all users of the TSF.

5013 FTA\_MCS.2 Per user attribute limitation on multiple concurrent sessions extends FTA\_MCS.1  
5014 Basic limitation on multiple concurrent sessions by requiring the ability to specify limitations  
5015 on the number of concurrent sessions based on the related security attributes.

### 5016 **16.3.3 Management of FTA\_MCS.1**

5017 The following actions **could** be considered for the management functions in FMT:

5018 a) Management of the maximum allowed number of concurrent user sessions by an  
5019 administrator.

#### 5020 16.3.4 Management of FTA\_MCS.2

5021 The following actions **could** be considered for the management functions in FMT:

- 5022           a) Management of the rules that govern the maximum allowed number of concurrent  
5023           user sessions by an administrator.

#### 5024 16.3.5 Audit of FTA\_MCS.1, FTA\_MCS.2

5025 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
5026 in the PP/ST:

- 5027           a) Minimal: Rejection of a new session based on the limitation of multiple concurrent  
5028           sessions.  
5029           b) Detailed: Capture of the number of currently concurrent user sessions and the user  
5030           security attribute(s).

#### 5031 16.3.6 FTA\_MCS.1 Basic limitation on multiple concurrent sessions

##### 5032 16.3.6.1 Component relationships

5033           Hierarchical to:                               No other components.  
5034           Dependencies:                                 FIA\_UID.1 Timing of identification

##### 5035 16.3.6.2 FTA\_MCS.1.1

5036 The TSF **shall** restrict the maximum number of concurrent sessions that belong to the  
5037 same user.

##### 5038 16.3.6.3 FTA\_MCS.1.2

5039 The TSF **shall** enforce, by default, a limit of [assignment: *default number*] sessions per  
5040 user.

#### 5041 16.3.7 FTA\_MCS.2 Per user attribute limitation on multiple concurrent sessions

##### 5042 16.3.7.1 Component relationships

5043           Hierarchical to:                               FTA\_MCS.1 Basic limitation on multiple concurrent  
5044           sessions  
5045           Dependencies:                                 FIA\_UID.1 Timing of identification

##### 5046 16.3.7.2 FTA\_MCS.2.1

5047 The TSF **shall** restrict the maximum number of concurrent sessions that belong to the same user  
5048 according to the rules [assignment: *rules for the number of maximum concurrent*  
5049 *sessions*].

##### 5050 16.3.7.3 FTA\_MCS.2.2

5051 The TSF **shall** enforce, by default, a limit of [assignment: *default number*] sessions per user.

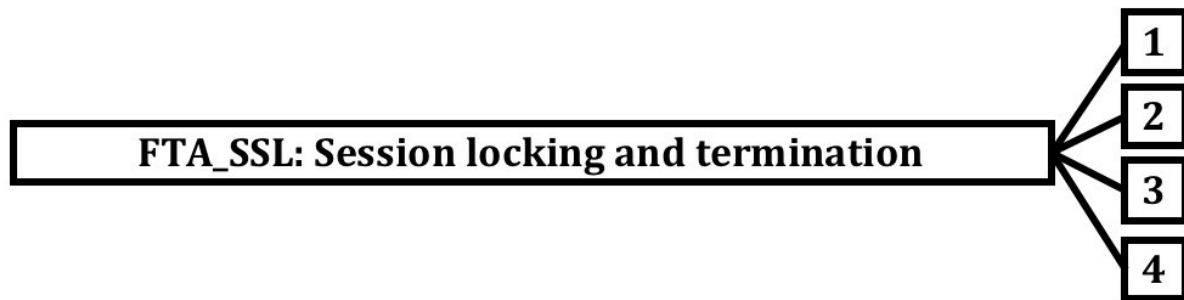
#### 5052 16.4 Session locking and termination (FTA\_SSL)

##### 5053 16.4.1 Family behaviour

5054 This family defines requirements for the TSF to provide the capability for TSF-initiated and  
5055 user-initiated locking, unlocking, and termination of interactive sessions.

5056 **16.4.2 Components leveling and description**

5057 Figure 86 shows the component leveling for this family.

5058 **Figure 86 — FTA\_SSL: Component leveling**

5059 FTA\_SSL.1 TSF-initiated session locking includes system-initiated locking of an interactive  
 5060 session after a specified period of user inactivity.

5061 FTA\_SSL.2 User-initiated locking, provides capabilities for the user to lock and unlock the user's  
 5062 own interactive sessions.

5063 FTA\_SSL.3 TSF-initiated termination, provides requirements for the TSF to terminate the  
 5064 session after a specified period of user inactivity.

5065 FTA\_SSL.4 User-initiated termination, provides capabilities for the user to terminate the user's  
 5066 own interactive sessions.

5067 **16.4.3 Management of FTA\_SSL.1**5068 The following actions **could** be considered for the management functions in FMT:

- 5069 a) Specification of the time of user inactivity after which lock-out occurs for an  
 5070 individual user;
- 5071 b) Specification of the default time of user inactivity after which lock-out occurs;
- 5072 c) Management of the events that **should** occur prior to unlocking the session.

5073 **16.4.4 Management of FTA\_SSL.2**5074 The following actions **could** be considered for the management functions in FMT:

- 5075 a) Management of the events that **should** occur prior to unlocking the session.

5076 **16.4.5 Management of FTA\_SSL.3**5077 The following actions **could** be considered for the management functions in FMT:

- 5078 a) Specification of the time of user inactivity after which termination of the interactive  
 5079 session occurs for an individual user;
- 5080 b) Specification of the default time of user inactivity after which termination of the  
 5081 interactive session occurs.

5082 **16.4.6 Management of FTA\_SSL.4**5083 The following actions **could** be considered for the management functions in FMT:

- 5084 a) There are no management activities foreseen.

5085 **16.4.7 Audit of FTA\_SSL.1, FTA\_SSL.2**

5086 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
5087 in the PP/ST:

- 5088 a) Minimal: Locking of an interactive session by the session locking mechanism.
- 5089 b) Minimal: Successful unlocking of an interactive session.
- 5090 c) Basic: Any attempts at unlocking an interactive session.

5091 **16.4.8 Audit of FTA\_SSL.3**

5092 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
5093 in the PP/ST:

- 5094 a) Minimal: Termination of an interactive session by the session locking mechanism.

5095 **16.4.9 Audit of FTA\_SSL.4**

5096 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
5097 in the PP/ST:

- 5098 a) Minimal: Termination of an interactive session by the user.

5099 **16.4.10 FTA\_SSL.1 TSF-initiated session locking**

5100 **16.4.10.1 Component relationships**

5101	Hierarchical to:	No other components.
5102	Dependencies:	FIA_UAU.1 Timing of authentication

5103 **16.4.10.2 FTA\_SSL.1.1**

5104 The TSF **shall** lock an interactive session after [assignment: *time interval of user*  
5105 *inactivity*] by:

- 5106 a) clearing or overwriting display devices, making the current contents  
5107 unreadable;
- 5108 b) disabling any activity of the user's data access/display devices other than  
5109 unlocking the session.

5110 **16.4.10.3 FTA\_SSL.1.2**

5111 The TSF **shall** require the following events to occur prior to unlocking the session:  
5112 [assignment: *events to occur*].

5113 **16.4.11 FTA\_SSL.2 User-initiated locking**

5114 **16.4.11.1 Component relationships**

5115	Hierarchical to:	No other components.
5116	Dependencies:	FIA_UAU.1 Timing of authentication

5117 **16.4.11.2 FTA\_SSL.2.1**

5118 The TSF **shall** allow user-initiated locking of the user's own interactive session, by:

- 5119 a) clearing or overwriting display devices, making the current contents  
5120 unreadable;
- 5121 b) disabling any activity of the user's data access/display devices other than  
5122 unlocking the session.

5123 **16.4.11.3 FTA\_SSL.2.2**

5124 The TSF **shall** require the following events to occur prior to unlocking the session:  
 5125 [assignment: *events to occur*].

5126 **16.4.12 FTA\_SSL.3 TSF-initiated termination**5127 **16.4.12.1 Component relationships**

5128 Hierarchical to: No other components.

5129 Dependencies: FMT\_SMR.1 Security roles

5130 **16.4.12.2 FTA\_SSL.3.1**

5131 The TSF **shall** terminate an interactive session after a [assignment: time interval of user  
 5132 inactivity].

5133 **16.4.13 FTA\_SSL.4 User-initiated termination**5134 **16.4.13.1 Component relationships**

5135 Hierarchical to: No other components.

5136 Dependencies: No dependencies.

5137 **16.4.13.2 FTA\_SSL.4.1**

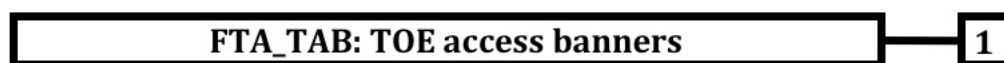
5138 The TSF **shall** allow user-initiated termination of the user's own interactive session.

5139 **16.5 TOE access banners (FTA\_TAB)**5140 **16.5.1 Family behaviour**

5141 This family defines requirements to display a configurable advisory warning message to users  
 5142 regarding the appropriate use of the TOE.

5143 **16.5.2 Components leveling and description**

5144 Figure 87 shows the component leveling for this family.



5145 **Figure 87 — FTA\_TAB: Component leveling**

5146 FTA\_TAB.1 Default TOE access banners, provides the requirement for a TOE Access Banner.  
 5147 This banner is displayed prior to the establishment dialogue for a session.

5148 **16.5.3 Management of FTA\_TAB.1**

5149 The following actions **could** be considered for the management functions in FMT:

5150 a) Maintenance of the banner by the authorized administrator.

5151 **16.5.4 Audit of FTA\_TAB.1**

5152 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 5153 in the PP/ST:

5154 a) There are no auditable events foreseen.

## 16.5.5 FTA\_TAB.1 Default TOE access banners

### 16.5.5.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

### 16.5.5.2 FTA\_TAB.1.1

Before establishing a user session, the [selection: *TSF, TOE platform*] **shall** display an [assignment: *description of the message*] message.

## 16.6 TOE access history (FTA\_TAH)

### 16.6.1 Family behaviour

This family defines requirements for the TSF to display to a user, upon successful session establishment, a history of successful and unsuccessful attempts to access the user's account.

### 16.6.2 Components leveling and description

Figure 88 shows the component leveling for this family.

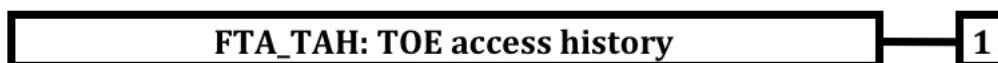


Figure 88 — FTA\_TAH: Component leveling

FTA\_TAH.1 TOE access history, provides the requirement for a TOE to display information related to previous attempts to establish a session.

### 16.6.3 Management of FTA\_TAH.1

The following actions **could** be considered for the management functions in FMT:

- a) There are no management activities foreseen.

### 16.6.4 Audit of FTA\_TAH.1

The following actions **should** be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- a) There are no auditable events foreseen.

### 16.6.5 FTA\_TAH.1 TOE access history

#### 16.6.5.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 16.6.5.2 FTA\_TAH.1.1

Upon successful session establishment, the TSF **shall** display the [selection: *date, time, method, location*] of the last successful session establishment to the user.

#### 16.6.5.3 FTA\_TAH.1.2

Upon successful session establishment, the TSF **shall** display the [selection: *date, time, method, location*] of the last unsuccessful attempt to session establishment and the number of unsuccessful attempts since the last successful session establishment.

5189 **16.6.5.4 FTA\_TAH.1.3**

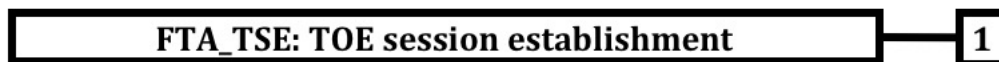
5190 The TSF **shall** not erase the access history information from the user interface without  
 5191 giving the user an opportunity to review the information.

5192 **16.7 TOE session establishment (FTA\_TSE)**5193 **16.7.1 Family behaviour**

5194 This family defines requirements to deny a user permission to establish a session with the TOE.

5195 **16.7.2 Components leveling and description**

5196 Figure 89 shows the component leveling for this family.



5197 **Figure 89 — FTA\_TSE: Component leveling**

5198 FTA\_TSE.1 TOE session establishment, provides requirements for denying users access to the  
 5199 TOE based on attributes.

5200 **16.7.3 Management of FTA\_TSE.1**

5201 The following actions **could** be considered for the management functions in FMT:

- 5202 a) Management of the session establishment conditions by the authorized  
 5203 administrator.

5204 **16.7.4 Audit of FTA\_TSE.1**

5205 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 5206 in the PP/ST:

- 5207 a) Minimal: Denial of a session establishment due to the session establishment  
 5208 mechanism.
- 5209 b) Basic: All attempts at establishment of a user session.
- 5210 c) Detailed: Capture of the value of the selected access parameters.

5211 **16.7.5 FTA\_TSE.1 TOE session establishment**5212 **16.7.5.1 Component relationships**

5213 Hierarchical to: No other components.

5214 Dependencies: No dependencies.

5215 **16.7.5.2 FTA\_TSE.1.1**

5216 The TSF **shall** be able to deny session establishment based on [assignment: *attributes*].

5217



## 17 Class FTP: Trusted path/channels

### 17.1 Class description

Families in this class provide requirements for a trusted communication path between users and the TSF, and for a trusted communication channel between the TSF and other trusted IT products. Trusted paths and channels have the following general characteristics:

- The communications path is constructed using internal and external communications channels (as appropriate for the component) that isolate an identified subset of TSF data and commands from the remainder of the TSF and user data.
- Use of the communications path **may can** be initiated by the user and/or the TSF (as appropriate for the component).
- The communications path is capable of providing assurance that the user is communicating with the correct TSF, and that the TSF is communicating with the correct user (as appropriate for the component).

In this paradigm, a trusted channel is a communication channel that **may can** be initiated by either side of the channel and provides non-repudiation characteristics with respect to the identity of the sides of the channel.

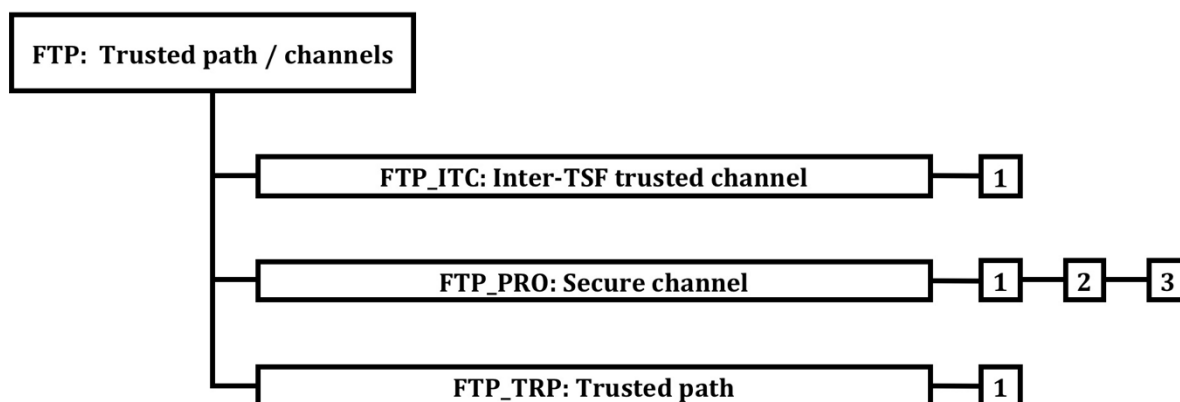
A trusted path provides a means for users to perform functions through an assured direct interaction with the TSF. Trusted path is usually desired for user actions such as initial identification and/or authentication but **may can** also be desired at other times during a user's session. Trusted path exchanges **may can** be initiated by a user or the TSF. User responses via the trusted path are guaranteed to be protected from modification by or disclosure to untrusted applications.

Families describing the use of commonly used communication protocols used in the provision of trusted channels and paths are also given.

Figure 90 shows the decomposition of this class, it's families and components. Elements are not shown in the figure.

Annex M provides explanatory information for this class and **should** be consulted when using the components identified in this class.

**Figure 90 — FTP: Trusted path/channels class decomposition**



### 17.2 Inter-TSF trusted channel (FTP\_ITC)

**Editors' note**

Editors are waiting for contribution from the CCDB Crypto Working Group

Editors await input from CCDB on FTP\_ITC: See N1462 DE/FG17 and the CCDB liaison statement from WG3 after Wuhan

5252 **17.2.1 Family behaviour**

5253 This family defines requirements for the creation of a trusted channel between the TSF and  
 5254 other trusted IT products for the performance of security critical operations. This family ~~should~~  
 5255 **can** be included whenever there are requirements for the secure communication of user or TSF  
 5256 data between the TOE and other trusted IT products.

5257 **17.2.2 Components leveling and description**

5258 Figure 91 shows the component leveling for this family.



5259 **Figure 91 — FTP\_ITC: Component leveling**

5260 FTP\_ITC.1 Inter-TSF trusted channel, requires that the TSF provide a trusted communication  
 5261 channel between itself and another trusted IT product.

5262 **17.2.3 Management of FTP\_ITC.1**

5263 The following actions **could** be considered for the management functions in FMT:

- 5264 a) Configuring the actions that require trusted channel, if supported.

5265 **17.2.4 Audit of FTP\_ITC.1**

5266 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 5267 in the PP/ST:

- 5268 a) Minimal: Failure of the trusted channel functions.  
 5269 b) Minimal: Identification of the initiator and target of failed trusted channel  
 5270 functions.  
 5271 c) Basic: All attempted uses of the trusted channel functions.  
 5272 d) Basic: Identification of the initiator and target of all trusted channel functions.

5273 **17.2.5 FTP\_ITC.1 Inter-TSF trusted channel**5274 **17.2.5.1 Component relationships**

5275	Hierarchical to:	No other components.
5276	Dependencies:	No dependencies.

5277 **17.2.5.2 FTP\_ITC.1.1**

5278 The TSF **shall** provide a communication channel between itself and another trusted IT  
 5279 product that is logically distinct from other communication channels and provides  
 5280 assured identification of its end points and protection of the channel data from  
 5281 modification or disclosure.

5282 **17.2.5.3 FTP\_ITC.1.2**

5283 The TSF **shall** permit [selection: *the TSF, another trusted IT product*] to initiate  
 5284 communication via the trusted channel.

5285 **17.2.5.4 FTP\_ITC.1.3**

5286 The TSF **shall** initiate communication via the trusted channel for [assignment: *list of*  
 5287 *functions for which a trusted channel is required*].

### 5288 17.3 Secure channel (FTP\_PRO)

5289 This family defines requirements for establishing a secure channel and using the secure channel  
5290 to transfer data securely.

#### 5291 17.3.1 Components leveling and description

5292 Figure 92 shows the component leveling for this family.



5293 **Figure 92 — FTP\_PRO: Family decomposition**

- 5294 a) Minimal: Establishment of the secure channel.
- 5295 b) Minimal: Failures of the secure channel functions.
- 5296 c) Minimal: Identification of the user associated with all secure channel failures, if  
5297 available.
- 5298 d) Basic: All attempted uses of the secure channel functions.
- 5299 e) Basic: Identification of the user associated with all secure channel invocations, if  
5300 available.

#### 5301 Editors' Note

5302 The Editors have proposed the text for management and audit above.

5303 Please review carefully.

5304 FTP\_PRO.1 Trusted channel protocol requires that communication be established in accordance  
5305 with a defined protocol.

#### 5306 17.3.1.1 FTP\_PRO.1.4

5307 The TSF **shall** enforce the following static protocol options: [assignment: *list of options*  
5308 *and references to standards in which each is defined*].

#### 5309 17.3.1.2 FTP\_PRO.1.5

5310 The TSF **shall** negotiate one of the following protocol configurations with its peer:  
5311 [assignment: *list of configurations and reference to standards in which each is defined*].

5312 FTP\_PRO.2 Trusted channel key establishment requires that keys be securely established  
5313 between the peers.

5314 FTP\_PRO.3 Trusted channel data protection requires that data in transit be protected.

#### 5315 17.3.2 Management of FTP\_PRO.1

5316 The following actions **could** be considered for the management functions in FMT:

- 5317 a) Configuring the actions that require secure channel, if supported.

#### 5318 17.3.3 Audit of FTP\_PRO.1

5319 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
5320 in the PP/ST:

- 5321 b) Minimal: Establishment of the secure channel.
- 5322 c) Minimal: Failures of the secure channel functions.

- 5323 d) Minimal: Identification of the user associated with all secure channel failures, if  
5324 available.
- 5325 e) Basic: All attempted uses of the secure channel functions.
- 5326 f) Basic: Identification of the user associated with all secure channel invocations, if  
5327 available.

**Editors' Note**

The Editors have proposed the text for management and audit above.

**Please review carefully.**

5331 **17.3.4 FTP\_PRO.1 Trusted channel protocol**

5332 **17.3.4.1 Component relationships**

- 5333 Hierarchical to: No other components.
- 5334 Dependencies: FTP\_PRO.2 Trusted channel key establishment  
5335 FTP\_PRO.3 Trusted channel data protection.

5336 **17.3.4.2 FTP\_PRO.1.1**

5337 The TSF **shall** implement [assignment: *trusted channel protocol*] acting as [assignment:  
5338 *defined protocol role(s)*] in accordance with: [assignment: *list of standards*].

5339 **17.3.4.3 FTP\_PRO.1.2**

5340 The TSF **shall** permit [selection: *itself, its peer*] to initiate communication via the trusted  
5341 channel.

5342 **17.3.4.4 FTP\_PRO.1.3**

5343 The TSF **shall** enforce the following rules for the trusted channel: [assignment: *rules*  
5344 *governing operation and use of the trusted channel and/or its protocol*].

5345 The TSF **shall** enforce usage of the trusted channel for [assignment: *purpose of the trusted*  
5346 *channel*] in accordance with: [assignment: *list of standards*].

5347 **17.3.4.5 FTP\_PRO.1.4**

5348 The TSF **shall** enforce the following static protocol options: [assignment: *list of options*  
5349 *and references to standards in which each is defined*].

5350 **17.3.4.6 FTP\_PRO.1.5**

5351 The TSF **shall** negotiate one of the following protocol configurations with its peer:  
5352 [assignment: *list of configurations and reference to standards in which each is defined*].

5353 **17.3.5 FTP\_PRO.2 Trusted channel key establishment**

5354 **17.3.5.1 Component relationships**

- 5355 Hierarchical to: No other components.
- 5356 g) Dependencies: Minimal: Establishment of the secure channel.
- 5357 h) Minimal: Failures of the secure channel functions.
- 5358 i) Minimal: Identification of the user associated with all secure channel failures, if  
5359 available.
- 5360 j) Basic: All attempted uses of the secure channel functions.

5361 k) Basic: Identification of the user associated with all secure channel invocations, if  
5362 available.

5363 **Editors' Note**

5364 The Editors have proposed the text for management and audit above.

5365 **Please review carefully.**

5366 FTP\_PRO.1 Trusted channel protocol  
5367 [FCS\_CKM.1 Cryptographic key generation, or  
5368 FCS\_CKM.2 Cryptographic key distribution]  
5369 FCS\_CKM.5 Cryptographic key derivation  
5370 FCS\_COP.1 Cryptographic operation.

5371 **17.3.5.2 FTP\_PRO.2.1**

5372 The TSF **shall** establish a shared secret with its peer using one of the following  
5373 mechanisms: [assignment: *list of key establishment mechanisms*].

5374 **17.3.5.3 FTP\_PRO.2.2**

5375 The TSF **shall** authenticate [selection: *its peer, itself to its peer*] using one of the following  
5376 mechanisms: [assignment: *list of authentication mechanisms*].

5377 **17.3.5.4 FTP\_PRO.2.3**

5378 The TSF **shall** use [assignment: *key derivation function*] to derive the following  
5379 cryptographic keys from a shared secret: [assignment: *list of cryptographic keys*]

5380 **17.3.6 FTP\_PRO.3 Trusted channel data protection**

5381 **17.3.6.1 Component relationships**

5382 Hierarchical to: No other components.

5383 l) Dependencies: Minimal: Establishment of the secure channel.

5384 m) Minimal: Failures of the secure channel functions.

5385 n) Minimal: Identification of the user associated with all secure channel failures, if  
5386 available.

5387 o) Basic: All attempted uses of the secure channel functions.

5388 p) Basic: Identification of the user associated with all secure channel invocations, if  
5389 available.

5390 **Editors' Note**

5391 The Editors have proposed the text for management and audit above.

5392 **Please review carefully.**

5393 FTP\_PRO.1 Trusted channel protocol  
5394 FTP\_PRO.2 Trusted channel key establishment  
5395 FCS\_COP.1 Cryptographic operation.

5396 **17.3.6.2 FTP\_PRO.3.1**

5397 The TSF **shall** protect data in transit from unauthorised disclosure using one of the  
5398 following mechanisms: [assignment: *list of encryption mechanisms*].

5399 **17.3.6.3 FTP\_PRO.3.2**

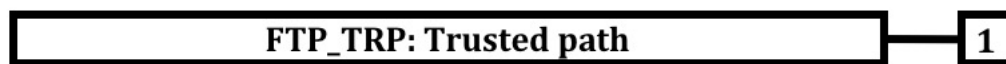
5400 The TSF **shall** protect data in transit from [selection: *modification, deletion, insertion,*  
 5401 *replay, [assignment: other]*] using one of the following mechanisms: [assignment: *list of*  
 5402 *integrity protection mechanisms*].

5403 **17.4 Trusted path (FTP\_TRP)**5404 **17.4.1 Family behaviour**

5405 This family defines the requirements to establish and maintain trusted communication to or  
 5406 from users and the TSF. A trusted path **may can** be required for any security-relevant  
 5407 interaction. Trusted path exchanges **may can** be initiated by a user during an interaction with  
 5408 the TSF, or the TSF **may can** establish communication with the user via a trusted path.

5409 **17.4.2 Components leveling and description**

5410 Figure 93 shows the component leveling for this family.



5411 **Figure 93 — FTP\_TRP: Component leveling**

5412 FTP\_TRP.1 Trusted path, requires that a trusted path between the TSF and a user be provided  
 5413 for a set of events defined by a PP/ST author. The user and/or the TSF **may can** have the ability  
 5414 to initiate the trusted path.

5415 **17.4.3 Management of FTP\_TRP.1**

5416 The following actions **could** be considered for the management functions in FMT:

- 5417 a) Configuring the actions that require trusted path, if supported.

5418 **17.4.4 Audit of FTP\_TRP.1**

5419 The following actions **should** be auditable if FAU\_GEN Security audit data generation is included  
 5420 in the PP/ST:

- 5421 b) Minimal: Failures of the trusted path functions.  
 5422 c) Minimal: Identification of the user associated with all trusted path failures, if  
 5423 available.  
 5424 d) Basic: All attempted uses of the trusted path functions.  
 5425 e) Basic: Identification of the user associated with all trusted path invocations, if  
 5426 available.

5427 **17.4.5 FTP\_TRP.1 Trusted path**5428 **17.4.5.1 Component relationships**

5429	Hierarchical to:	No other components.
5430	Dependencies:	No dependencies.

5431 **17.4.5.2 FTP\_TRP.1.1**

5432 The TSF **shall** provide a communication path between itself and [selection: *remote, local*]  
 5433 users that is logically distinct from other communication paths and provides assured  
 5434 identification of its end points and protection of the communicated data from [selection:  
 5435 *modification, disclosure, [assignment: other types of integrity or confidentiality violation]*].

5436 **17.4.5.3 FTP\_TRP.1.2**

5437 The TSF **shall** permit [selection: *the TSF, local users, remote users*] to initiate  
5438 communication via the trusted path.

5439 **17.4.5.4 FTP\_TRP.1.3**

5440 The TSF **shall** require the use of the trusted path for [selection: *initial user*  
5441 *authentication, [assignment: other services for which trusted path is required]*].

5442

5443

5444

Annex A

(normative)

Security functional requirements structure of the application notes

5445

A.1 General information

5446

5447

5448

5449

5450

This annex contains additional guidance for the families and components defined in the elements of this document, which **may** be required by users, developers, or evaluators to use the components. To facilitate finding the appropriate information, the presentation of the classes, families and components in this annex is similar to the presentation within the elements.

5451

A.2 Structure of the notes

5452

5453

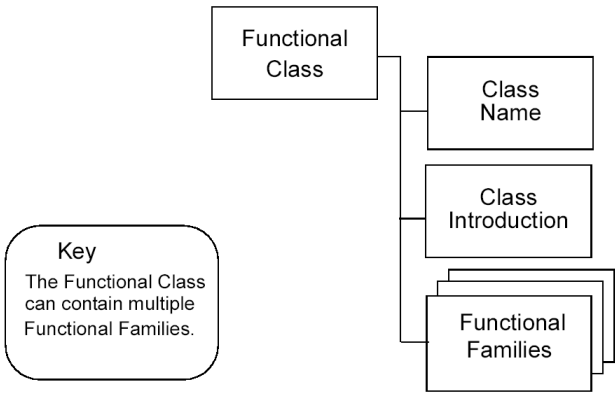
This clause defines the content and presentation of the notes related to functional requirements in this document.

5454

A.2.1 Class structure

5455

Figure 94 below illustrates the functional class structure in this annex.



5456

5457

Figure 94 — Functional class structure

5458

A.2.1.1 Class name

5459

This is the unique name of the class defined within the normative elements of this document.

5460

A.2.1.2 Class introduction

5461

5462

5463

5464

The class introduction in this annex provides information about the use of the families and components of the class. This information is completed with the informative diagram that describes the organization of each class with the families in each class and the hierarchical relationship between components in each family.

5465

A.2.2 Family structure

5466

Figure 95 illustrates the functional family structure for application notes in diagrammatic form.



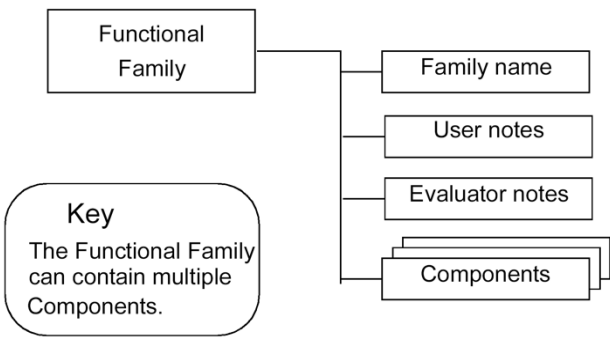


Figure 95 — Functional family structure for application notes

**A.2.2.1 Family name**

This is the unique name of the family defined within the normative elements of this document.

**A.2.2.2 User notes**

The user notes contain additional information that is of interest to potential users of the family, that is PP, ST and functional package authors, and developers of TOEs incorporating the functional components. The presentation is informative and might cover warnings about limitations of use and areas where specific attention might be required when using the components.

**A.2.2.3 Evaluator notes**

The evaluator notes contain any information that is of interest to developers and evaluators of TOEs that claim compliance with a component of the family. The presentation is informative and **can** cover a variety of areas where specific attention might be needed when evaluating the TOE. This **can** include clarifications of meaning and specification of the way to interpret requirements, as well as caveats and warnings of specific interest to evaluators.

These User Notes and Evaluator Notes subclauses are not mandatory and appear only if appropriate.

**A.2.3 Component structure**

Figure 96 illustrates the functional component structure for the application notes.

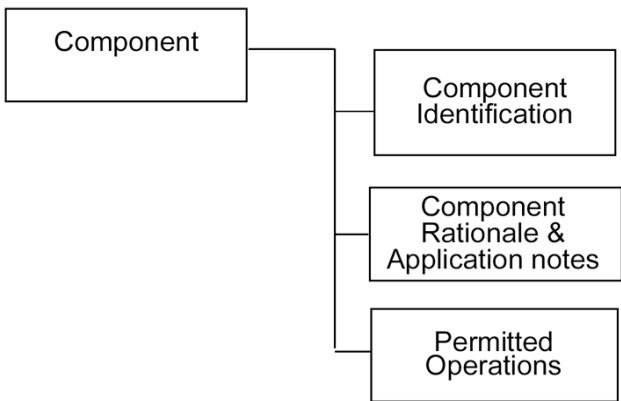


Figure 96 — Functional component structure

**A.2.3.1 Component identification**

This is the unique name of the component defined within the normative elements of this document.

**A.2.3.2 Component rationale and application notes**

- 5493 Any specific information related to the component ~~can be~~ is found in this subclause.
- 5494 — The *rationale* contains the specifics of the rationale that refine the general statements on  
 5495 rationale for the specific level and ~~should~~ is only be used if level specific amplification is  
 5496 required.
- 5497 — The *application notes* contain additional refinement in terms of narrative qualification as it  
 5498 pertains to a specific component. This refinement ~~can~~ pertain to user notes, and/or  
 5499 evaluator notes as described in A.2.2. This refinement ~~can~~ be used to explain the nature of  
 5500 the dependencies.

EXAMPLE

Shared information, or shared operation.

5501 This subclause is not mandatory and appears only if appropriate.

5502 **A.2.3.3 Permitted operations**

5503 This portion of each component contains advice relating to the permitted operations of the  
 5504 component.

5505 This subclause is not mandatory and appears only if appropriate.

5506  
5507  
5508

**Annex B**  
**(informative)**  
**Dependency tables for security functional components**

5509  
5510  
5511  
5512  
5513

Editors' Note:  
There is a proposal that these dependencies tables are not needed and can be removed.  
Comments from WG 3 Experts on this notion are requested.  
This annex will need updating, once the new SFRs and their dependencies have settled down. In this draft placeholders have been created.

5514  
5515  
5516  
5517  
5518  
5519  
5520

**B.1 Dependency tables**  
The following dependency tables for functional components show their direct, indirect, and optional dependencies. Each of the components that is a dependency of some functional component is allocated a column. Each functional component is allocated a row. The value in the table cell indicate whether the column label component is directly required (indicated by a cross "X"), indirectly required (indicated by a dash "-"), or optionally required (indicated by a "O") by the row label component.

EXAMPLE  
An example of a component with optional dependencies is FDP\_ETC.1 Export of user data without security attributes, which requires either FDP\_ACC.1 Subset access control or FDP\_IFC.1 Subset information flow control to be present. So, if FDP\_ACC.1 Subset access control is present, FDP\_IFC.1 Subset information flow control is not necessary and vice versa.

5521  
5522

If no character is presented, the component is not dependent upon another component.

5523

**Table B.2 — Dependency table for Class FAU: Security audit**

	FAU_GEN.1	FAU_SAA.1	FAU_SAR.1	FAU_STG.1	FAU_STG.4	FIA_UID.1	FMT_MTD.1	FMT_SMF.1	FMT_SMR.1	FPT_STM.1
FAU_ARP.1	-	X								-
FAU_GEN.1										X
FAU_GEN.2	X					X				-
FAU_SAA.1	X									-
FAU_SAA.2						X				
FAU_SAA.3										
FAU_SAA.4										
FAU_SAR.1	X									-
FAU_SAR.2	-		X							-
FAU_SAR.3	-		X							-
FAU_SEL.1	X					-	X	-	-	-
FAU_STG.1	X									-
FAU_STG.2	X									-
FAU_STG.3	-			X						-
FAU_STG.4	-			X						-
FAU_STG.5				-	X					-

5524

5525

**Table B.3 — Dependency table for Class FCO: Communication**

	FIA_UID.1	FCO_TCO.1
FCO_NRO.1	X	
FCO_NRO.2	X	
FCO_NRR.1	X	
FCO_NRR.2	X	
FCO_TCO.1		
FCO_TCO.2		X

5526

5527

**Table B.4 — Dependency table for Class FCS: Cryptographic support**

	FCS_CKM.1	FCS_CKM.2	FCS_CKM.4	FCS_COP.1	FCS_RBG.2	FDP_ACC.1	FDP_ACF.1	FDP_IFC.1	FDP_IFF.1	FDP_JTC.1	FDP_JTC.2	FIA_UID.1	FMT_MSA.1	FMT_MSA.3	FMT_SMF.1	FMT_SMR.1	FPT_TDC.1	FTP_JTC.1	FTP_TRP.1
<b>FCS_CKM.1</b>	-	0	X	0		-	-	-	-	-	-	-	-	-	-	-	-	-	-
<b>FCS_CKM.2</b>	0	-	X	-		-	-	-	-	0	0	-	-	-	-	-	-	-	-
<b>FCS_CKM.3</b>	0	-	X	-		-	-	-	-	0	0	-	-	-	-	-	-	-	-
<b>FCS_CKM.5</b>	0	-	-	-		-	-	-	-	0	0	-	-	-	-	-	-	-	-
<b>FCS_CKM.6</b>																			
<b>FCS_COP.1</b>	0	-	X	-		-	-	-	-	0	0	-	-	-	-	-	-	-	-
<b>FCS_RBG.1</b>																			
<b>FCS_RBG.2</b>																			
<b>FCS_RBG.3</b>																			
<b>FCS_RBG.4</b>																			
<b>FCS_RBG.5</b>																			
<b>FCS_RBG.6</b>																			
<b>FCS_RNG.1</b>																			

5528

5529

Table B.5 — Dependency table for Class FDP: User data protection

	FDP_ACC.1	FDP_ACF.1	FDP_IFC.1	FDP_IFF.1	FDP_ITT.1	FDP_ITT.2	FDP_UIT.1	FIA_UID.1	FMT_MSA.1	FMT_MSA.3	FMT_SMF.1	FMT_SMR.1	FPT_TDC.1	FTP_ITC.1	FTP_TRP.1
FDP_ACC.1	-	X	-	-				-	-	-	-	-			
FDP_ACC.2	-	X	-	-				-	-	-	-	-			
FDP_ACF.1	X	-	-	-				-	-	X	-	-			
FDP_DAU.1															
FDP_DAU.2								X							
FDP_ETC.1	0	-	0	-				-	-	-	-	-			
FDP_ETC.2	0	-	0	-				-	-	-	-	-			
FDP_IFC.1	-	-	-	X				-	-	-	-	-			
FDP_IFC.2	-	-	-	X				-	-	-	-	-			
FDP_IFF.1	-	-	X	-				-	-	X	-	-			
FDP_IFF.2	-	-	X	-				-	-	X	-	-			
FDP_IFF.3	-	-	X	-				-	-	-	-	-			
FDP_IFF.4	-	-	X	-				-	-	-	-	-			
FDP_IFF.5	-	-	X	-				-	-	-	-	-			
FDP_IFF.6	-	-	X	-				-	-	-	-	-			
FDP_IRC.1															
FDP_IRC.2															
FDP_ITC.1	0	-	0	-				-	-	X	-	-			
FDP_ITC.2	0	-	0	-				-	-	-	-	-	X	0	0
FDP_ITT.1	0	-	0	-				-	-	-	-	-			
FDP_ITT.2	0	-	0	-				-	-	-	-	-			
FDP_ITT.3	0	-	0	-	X			-	-	-	-	-			
FDP_ITT.4	0	-	0	-		X		-	-	-	-	-			
FDP_RIP.1															
FDP_RIP.2															
FDP_ROL.1	0	-	0	-				-	-	-	-	-			
FDP_ROL.2	0	-	0	-				-	-	-	-	-			
FDP_SDC.1															
FDP_SDC.2															
FDP_SDI.1															

	FDP_TRP.1	FTP_ITC.1	FPT_TDC.1	FMT_SMR.1	FMT_SMF.1	FMT_MSA.3	FMT_MSA.1	FIA_UID.1	FDP_UIT.1	FDP_ITT.2	FDP_ITT.1	FDP_IFF.1	FDP_IFC.1	FDP_ACF.1	FDP_ACC.1
FDP_SDI.2															
FDP_UCT.1	0	0		-	-	-	-	-				-	0	-	0
FDP_UIT.1	0	0		-	-	-	-	-				-	0	-	0
FDP_UIT.2	0	-		-	-	-	-	-	0			-	0	-	0
FDP_UIT.3	0	-		-	-	-	-	-	0			-	0	-	0

5531  
5532

5533 **Table B.6 — Dependency table for Class FIA: Identification and authentication**

	FIA_ATD.1	FIA_UAU.1	FIA_UID.1	FMT_SMR.1
FIA_AFL.1		X	-	
FIA_API.1				
FIA_ATD.1				
FIA_SOS.1				
FIA_SOS.2				
FIA_UAU.1			X	
FIA_UAU.2			X	
FIA_UAU.3				
FIA_UAU.4				
FIA_UAU.5				
FIA_UAU.6				
FIA_UAU.7		X	-	
FIA_UID.1				
FIA_UID.2				
FIA_USB.1	X			

5534  
5535 **Table B.7 — Dependency table for Class FMT: Security management**

	FDP_ACC.1	FDP_ACF.1	FDP_IFC.1	FDP_IFF.1	FIA_UID.1	FMT_LIM.1	FMT_LIM.2	FMT_MSA.1	FMT_MSA.3	FMT_MTD.1	FMT_SMF.1	FMT_SMR.1	FPT_STM.1
FMT_LIM.1						-	X						
FMT_LIM.2						X	-						
FMT_MOF.1					-						X	X	
FMT_MSA.1	0	-	0	-	-			-	-		X	X	
FMT_MSA.2	0	-	0	-	-			X	-		-	X	
FMT_MSA.3	-	-	-	-	-			X	-		-	X	
FMT_MSA.4	0	-	0	-	-			-	-		-	-	
FMT_MTD.1					-						X	X	
FMT_MTD.2					-					X	-	X	
FMT_MTD.3					-					X	-	-	
FMT_REV.1					-							X	
FMT_SAE.1					-							X	X



	FPT_STM.1	FMT_SMR.1	FMT_SMF.1	FMT_MTD.1	FMT_MSA.3	FMT_MSA.1	FMT_LIM.2	FMT_LIM.1	FIA_UID.1	FDP_IFF.1	FDP_IFC.1	FDP_ACF.1	FDP_ACG.1
FMT_SMF.1													
FMT_SMR.1									X				
FMT_SMR.2									X				
FMT_SMR.3		X							-				

5536  
5537

5538

**Table B.8 — Dependency table for Class FPR: Privacy**

	<b>FIA_UID.1</b>	<b>FPR_ANO.1</b>	<b>FPR_UNO.1</b>
<b>FPR_ANO.1</b>			
<b>FPR_ANO.2</b>		X	
<b>FPR_PSE.1</b>			
<b>FPR_PSE.2</b>	X		
<b>FPR_PSE.3</b>			
<b>FPR_TRD.1</b>			
<b>FPR_TRD.2</b>			
<b>FPR_TRD.3</b>			
<b>FPR_UNL.1</b>			
<b>FPR_UNL.2</b>			
<b>FPR_UNL.3</b>			
<b>FPR_UNO.1</b>			
<b>FPR_UNO.2</b>			
<b>FPR_UNO.3</b>			X
<b>FPR_UNO.4</b>			

5539

5540

5541

Table B.9 — Dependency table for Class FPT: Protection of the TSF

	AGD_OPE.1	FIA_UID.1	FMT_LIM.1	FMT_SMF.1	FMT_SMR.1	FPT_ITT.1	FPT_STM.1
FPT_ADM							
FPT_EMS.1							
FPT_FLS.1							
FPT_ITA.1							
FPT_ITC.1							
FPT_ITL.1							
FPT_ITL.2							
FPT_ITT.1							
FPT_ITT.2							
FPT_ITT.3						X	
FPT_PHP.1							
FPT_PHP.2		-	X	-	-		
FPT_PHP.3							
FPT_RCV.1	X						
FPT_RCV.2	X						
FPT_RCV.3	X						
FPT_RCV.4							
FPT_RPL.1							
FPT_SSP.1						X	
FPT_SSP.2						X	
FPT_STM.1							
FPT_STM.2					X		X
FPT_TDC.1							
FPT_TEE.1							
FPT_TRC.1						X	
FPT_TST.1							

5542

5543

Table B.10 — Dependency table for Class FRU: Resource utilization

	FPT_FLS.1
FRU_FLT.1	X
FRU_FLT.2	X

5544

	FPT_FIS.1
FRU_PRS.1	
FRU_PRS.2	
FRU_RSA.1	
FRU_RSA.2	

5545

**Table B.11 — Dependency table for Class FTA: TOE access**

	<b>FIA_UAU.1</b>	<b>FIA_UID.1</b>
<b>FTA_LSA.1</b>		
<b>FTA_MCS.1</b>		X
<b>FTA_MCS.2</b>		X
<b>FTA_SSL.1</b>	X	-
<b>FTA_SSL.2</b>	X	-
<b>FTA_SSL.3</b>		
<b>FTA_SSL.4</b>		
<b>FTA_TAB.1</b>		
<b>FTA_TAH.1</b>		
<b>FTA_TSE.1</b>		

5546

5547

**Table B.12 — Dependency table for Class FTP: Trusted Path/channels**

<b>FTP_ITC.1</b>	
<b>FTP_PRO.1</b>	
<b>FTP_PRO.2</b>	
<b>FTP_PRO.3</b>	
<b>FTP_TRP.1</b>	

## Annex C (normative)

### Class FAU: Security audit - application notes

#### Editor' Notes

In this and following annexes the Editors' are attempting to modernize a little so as to present the standard as appropriate for use in the 21<sup>st</sup> Century.

E.g.

Examples including "floppy disks" have been adjusted.

The notion of "The Internet" has been mentioned.

The Editors' request more suggestions for improvement.

#### C.1 General information

ISO/IEC 15408 audit families allow PP/ST authors the ability to define requirements for monitoring user activities and, in some cases, detecting real, possible, or imminent violations of the enforcement of the SFRs. The TOE's security audit functions are defined to help monitor security-relevant events, and act as a deterrent against security violations. The requirements of the audit families refer to functions that include audit data protection, record format, and event selection, as well as analysis tools, violation alarms, and real-time analysis. The audit records trail **should** be presented in human-readable format either directly or indirectly or both.

##### EXAMPLE 1

An example of direct presentation is storing the audit records in human-readable format

An example of indirect presentation is by using audit reduction tools.

While developing the security audit requirements, the PP/ST author **should** take note of the inter-relationships among the audit families and components. The potential exists to specify a set of audit requirements that comply with the family/component dependencies lists, while at the same time resulting in a deficient audit function.

##### EXAMPLE 2

An audit function that requires all security relevant events to be audited but without the selectivity to control them on any reasonable basis such as individual user or object.

#### C.2 Audit requirements in a distributed environment

The implementation of audit requirements for networks and other large systems **may can** differ significantly from those needed for stand-alone systems. Larger, more complex, and active systems require more thought concerning which audit data to collect and how this **should can** be managed, due to lowered feasibility of interpreting (or even storing) what gets collected. The traditional notion of a time-ordered list, set of records or "trail" of audited events **may** is not always applicable in a global asynchronous network with many arbitrary events occurring at once.

Also, different hosts and servers on a distributed TOE **may can** have differing naming policies and values. Further, the use of symbolic names for audit review **may** requires a net-wide convention to avoid redundancies and "name clashes."

A multi-object audit repository, portions of which are accessible by a potentially wide variety of authorized users, **may be** are usually required if audit repositories are to serve a useful function in distributed systems.

5585 Finally, misuse of authority by authorized users ~~should~~ **can** be addressed by systematically  
 5586 avoiding local storage of audit data pertaining to administrator actions.

### 5587 **C.3 Security audit automatic response (FAU\_ARP)**

#### 5588 **C.3.1 User notes**

5589 The Security audit automatic response family describes requirements for the handling of audit  
 5590 events. The requirement **could** include requirements for alarms or TSF action (automatic  
 5591 response).

##### EXAMPLE

the TSF **could** include the generation of real time alarms, termination of the offending process, disabling of a service, or disconnection or invalidation of a user account.

5592 An audit event is defined to be an “potential security violation” if so indicated by the Security  
 5593 audit analysis (FAU\_SAA) components.

#### 5594 **C.3.2 FAU\_ARP.1 Security alarms**

##### 5595 **C.3.2.1 User application notes**

5596 An action **should** be taken for follow up action in the event of an alarm. This action ~~can~~ **may** be  
 5597 to inform the authorized user, to present the authorized user with a set of possible containment  
 5598 actions, or to take corrective actions. The timing of the actions **should** be carefully considered  
 5599 by the PP/ST author.

##### Editors' Note

Is the list of actions intended to be an exhaustive list?

##### 5602 **C.3.2.2 Operations**

##### 5603 **C.3.2.2.1 Assignment**

5604 In FAU\_ARP.1.1, the PP/ST author **should** specify the actions to be taken in case of a potential  
 5605 security violation.

##### EXAMPLE

An example of such a list is: “inform the authorized user, disable the subject that created the potential security violation.”

5606 It ~~can~~ **may** also specify that the action to be taken **can** be specified by an authorized user.

### 5607 **C.4 Security audit data generation (FAU\_GEN)**

#### 5608 **C.4.1 User notes**

5609 The Security audit data generation family includes requirements to specify the audit events that  
 5610 **should** be generated by the TSF for security-relevant events.

5611 This family is presented in a manner that avoids a dependency on all components requiring  
 5612 audit support. Each component has an audit subclause developed in which the events to be  
 5613 audited for that functional area are listed. When the PP/ST author assembles the PP/ST, the  
 5614 items in the audit area are used to complete the variable in this component. Thus, the  
 5615 specification of what **could** be audited for a functional area is localized in that functional area.

5616 The list of auditable events is entirely dependent on the other functional families within the  
 5617 PP/ST. Each family definition **should** therefore include a list of its family-specific auditable  
 5618 events. Each auditable event in the list of auditable events specified in the functional family  
 5619 **should** correspond to one of the levels of audit event generation specified in this family (i.e.  
 5620 minimal, basic, detailed). This provides the PP/ST author with information necessary to ensure  
 5621 that all appropriate auditable events are specified in the PP/ST. The following example shows  
 5622 how auditable events are to be specified in appropriate functional families:

5623 “The following actions **should** be auditable if Security audit data generation (FAU\_GEN) is  
5624 included in the PP/ST:

- 5625 a) Minimal: Successful use of the user security attribute administration functions.
- 5626 b) Basic: All attempted uses of the user security attribute administration functions.
- 5627 c) Basic: Identification of which user security attributes have been modified.
- 5628 d) Detailed: With the exception of specific sensitive attribute data items, the new  
5629 values of the attributes **should** be captured.”

EXAMPLE 1

Sensitive attribute data items include passwords and cryptographic keys.

5630 For each functional component that is chosen, the auditable events that are indicated in that  
5631 component, at and below the level indicated in Security audit data generation (FAU\_GEN)  
5632 **should** be auditable. If, for example, in the previous example “Basic” would be selected in  
5633 Security audit data generation (FAU\_GEN), the auditable events mentioned in a), b) and c)  
5634 **should** be auditable.

5635 Observe that the categorization of auditable events is hierarchical.

EXAMPLE

For example, when Basic Audit Generation is desired, all auditable events identified as being either Minimal or Basic, **should** also be included in the PP/ST through the use of the appropriate assignment operation, except when the higher-level event simply provides more detail than the lower level event.

5636

5637 When Detailed Audit Generation is desired, all identified auditable events (Minimal, Basic, and  
5638 Detailed) **should** be included in the PP/ST.

5639 A PP/ST author **may** decide to include other auditable events beyond those required for a given  
5640 audit level.

EXAMPLE 1

For example, the PP/ST **may** claim only minimal audit capabilities while including most of the basic capabilities because the few excluded capabilities conflict with other PP/ST constraints (perhaps because they require the collection of unavailable data).

5641 The functionality that creates the auditable event **should** be specified in the PP or ST as a  
5642 functional requirement.

EXAMPLE 2

The following are examples of the types of the events that **should can** be defined as auditable within each PP/ST functional component:

- a) Introduction of objects within the control of the TSF into a subject's address space;
- b) Deletion of objects;
- c) Distribution or revocation of access rights or capabilities;
- d) Changes to subject or object security attributes;
- e) Policy checks performed by the TSF as a result of a request by a subject;
- f) The use of access rights to bypass a policy check;
- g) Use of Identification and Authentication functions;
- h) Actions taken by an operator, and/or authorized user (such as. suppression of a TSF protection mechanism as human-readable labels);
- i) Import/export of data from/to removable media (such as printed output, tapes, USB sticks).

## 5643 C.4.2 FAU\_GEN.1 Audit data generation

### 5644 C.4.2.1 User application notes



This component defines requirements to identify the auditable events for which audit records **should** be generated, and the information to be provided in the audit records.

FAU\_GEN.1 Audit data generation by itself might be used when the SFRs do not require that individual user identities be associated with audit events. This **could** be appropriate when the PP/ST also contains privacy requirements. If the user identity must be incorporated FAU\_GEN.2 User identity association **could** be used in addition.

If the subject is a user, the user identity **may** be recorded as the subject identity. The identity of the user **may** not yet been verified if User authentication (FIA\_UAU) has not been applied. Therefore, in the instance of an invalid login the claimed user identity **should** be recorded. It **should** be considered to indicate when a recorded identity has not been authenticated.

#### **C.4.2.2 Evaluator notes**

There is a dependency on Time stamps (FPT\_STM). If correctness of time is not an issue for this TOE, elimination of this dependency **could** be justified.

#### **C.4.2.3 Operations**

##### **C.4.2.3.1 Selection**

In FAU\_GEN.1.1, the PP/ST author **should** select the level of auditable events called out in the audit subclause of other functional components included in the PP/ST. This level is one of the following: “minimum”, “basic”, “detailed” or “not specified”.

##### **C.4.2.3.2 Assignment**

In FAU\_GEN.1.1, the PP/ST author **should** assign a list of other specifically defined auditable events to be included in the list of auditable events. The assignment **may** comprise none, or events that **could** be auditable events of a functional requirement that are of a higher audit level than requested in b), as well as the events generated through the use of a specified Application Programming Interface (API).

In FAU\_GEN.1.2, the PP/ST author **should** assign, for each of the auditable events included in the PP/ST, either a list of other audit relevant information to be included in audit events records or none.

#### **C.4.3 FAU\_GEN.2 User identity association**

##### **C.4.3.1 User application notes**

This component addresses the requirement of accountability of auditable events at the level of individual user identity. This component **should** be used in addition to FAU\_GEN.1 Audit data generation.

There is a potential conflict between the audit and privacy requirements. For audit purposes, it **may** be desirable to know who performed an action. The user **may** want to keep his/her actions to himself/herself and not be identified by other persons such as a site with job offers. Or it might be required in the Organizational Security Policy that the identity of the users must be protected. In those cases, the objectives for audit and privacy might contradict each other. Therefore, if this requirement is selected and privacy is important, inclusion of the component user pseudonymity might be considered. Requirements on determining the real user name based on its pseudonym are specified in the privacy class.

If the identity of the user has not yet been verified through authentication, in the instance of an invalid login the claimed user identity **should** be recorded. It **should** be considered to indicate when a recorded identity has not been authenticated.

#### **C.5 Security audit analysis (FAU\_SAA)**

##### **C.5.1 User notes**

5690 This family defines requirements for automated means that analyze system activity and audit  
 5691 data looking for possible or real security violations. This analysis **may** work in support of  
 5692 intrusion detection, or automatic response to a potential security violation.

5693 The action to be performed by the TSF on detection of a potential violation is defined in Security  
 5694 audit automatic response (FAU\_ARP) components.

5695 For real-time analysis, audit data **could** be transformed into a useful format for automated  
 5696 treatment, but into a different useful format for delivery to authorized users for review.

## 5697 **C.5.2 FAU\_SAA.1 Potential violation analysis**

### 5698 **C.5.2.1 User application notes**

5699 This component is used to specify the set of auditable events whose occurrence or accumulated  
 5700 occurrence held to indicate a potential violation of the enforcement of the SFRs, and any rules to  
 5701 be used to perform the violation analysis.

### 5702 **C.5.2.2 Operations**

#### 5703 **C.5.2.2.1 Assignment**

5704 In FAU\_SAA.1.2, the PP/ST author **should** identify the subset of defined auditable events whose  
 5705 occurrence or accumulated occurrence need to be detected as an indication of a potential  
 5706 violation of the enforcement of the SFRs.

5707 In FAU\_SAA.1.2, the PP/ST author **should** specify any other rules that the TSF **should** use in its  
 5708 analysis of the audit trail. Those rules **could** include specific requirements to express the needs  
 5709 for the events to occur in a certain period of time. If there are no additional rules that the TSF  
 5710 **should** use in the analysis of the audit trail, this assignment **can** be completed with “none”.

#### EXAMPLE

Period of time: period of the day, duration

## 5711 **C.5.3 FAU\_SAA.2 Profile based anomaly detection**

### 5712 **C.5.3.1 User application notes**

5713 A *profile* is a structure that characterizes the behaviour of users and/or subjects; it represents  
 5714 how the users/subjects interact with the TSF in a variety of ways. Patterns of usage are  
 5715 established with respect to the various types of activity the users/subjects engage in. The ways  
 5716 in which the various types of activity are recorded in the profile are referred to as *profile*  
 5717 *metrics*.

#### EXAMPLE

Patterns of usage: patterns in exceptions raised, patterns in resource utilization (when, which, how), patterns in actions performed.

Profile metrics: resource measures, event counters, timers

5718 Each profile represents the expected patterns of usage performed by members of the *profile*  
 5719 *target group*. This pattern **may** be based on past use (historical patterns) or on normal use for  
 5720 users of similar target groups (expected behaviour). A profile target group refers to one or more  
 5721 users who interact with the TSF. The activity of each member of the profile group is used by the  
 5722 analysis tool in establishing the usage patterns represented in the profile. The following are  
 5723 some examples of profile target groups:

- 5724 a) **Single user account:** one profile per user;
- 5725 b) **Group ID or Group Account:** one profile for all users who possess the same group  
 5726 ID or operate using the same group account;
- 5727 c) **Operating Role:** one profile for all users sharing a given operating role;

5728 d) **System:** one profile for all users of a system.

5729 Each member of a profile target group is assigned an individual *suspicion rating* that represents  
5730 how closely that member's new activity corresponds to the established patterns of usage  
5731 represented in the group profile.

5732 The sophistication of the anomaly detection tool will largely be determined by the number of  
5733 target profile groups required by the PP/ST and the complexity of the required profile metrics.

5734 The PP/ST author **should** enumerate specifically what activity **should** be monitored and/or  
5735 analysed by the TSF. The PP/ST author **should** also identify specifically what information  
5736 pertaining to the activity is necessary to construct the usage profiles.

5737 FAU\_SAA.2 Profile based anomaly detection requires that the TSF maintain profiles of system  
5738 usage. The word maintain implies that the anomaly detector is actively updating the usage  
5739 profile based on new activity performed by the profile target members. It is important here that  
5740 the metrics for representing user activity are defined by the PP/ST author.

EXAMPLE

For example, there **may** be a thousand different actions an individual **may** be capable of performing, but the anomaly detector **may** choose to monitor a subset of that activity.

5741 Anomalous activity gets integrated into the profile just like non-anomalous activity (assuming  
5742 the tool is monitoring those actions). Things that **may** have appeared anomalous four months  
5743 ago, might over time become the norm (and vice-versa) as the user's work duties change. The  
5744 TSF wouldn't be able to capture this notion if it filtered out anomalous activity from the profile  
5745 updating algorithms.

5746 Administrative notification **should** be provided such that the authorized user understands the  
5747 significance of the suspicion rating.

5748 The PP/ST author **should** define how to interpret suspicion ratings and the conditions under  
5749 which anomalous activity is indicated to the Security audit automatic response (FAU\_ARP)  
5750 mechanism.

### 5751 C.5.3.2 Operations

#### 5752 C.5.3.2.1 Assignment

5753 In FAU\_SAA.2.1, the PP/ST author **should** specify the profile target group. A single PP/ST **may**  
5754 include multiple profile target groups.

5755 In FAU\_SAA.2.3, the PP/ST author **should** specify conditions under which anomalous activity is  
5756 reported by the TSF. Conditions **may** include the suspicion rating reaching a certain value, or be  
5757 based on the type of anomalous activity observed.

### 5758 C.5.4 FAU\_SAA.3 Simple attack heuristics

#### 5759 C.5.4.1 User application notes

5760 In practice, it is at best rare when an analysis tool **can** detect with certainty when a security  
5761 violation is imminent. However, there do exist some system events that are so significant that  
5762 they are always worthy of independent review.

EXAMPLE

Example of such events include the deletion of a key TSF security data file (such as the password file) or activity such as a remote user attempting to gain administrative privilege.

5763 These events are referred to as signature events in that their occurrence in isolation from the  
5764 rest of the system activity are indicative of intrusive activity.

5765 The complexity of a given tool will depend greatly on the assignments defined by the PP/ST  
5766 author in identifying the base set of *signature events*.

5767 The PP/ST author **should** enumerate specifically what events **should** be monitored by the TSF in  
 5768 order to perform the analysis. The PP/ST author **should** identify specifically what information  
 5769 pertaining to the event is necessary to determine if the event maps to a signature event.

5770 Administrative notification **should** be provided such that the authorized user understands the  
 5771 significance of the event and the appropriate possible responses.

5772 An effort was made in the specification of these requirements to avoid a dependency on audit  
 5773 data as the sole input for monitoring system activity. This was done in recognition of the  
 5774 existence of previously developed intrusion detection tools that do not perform their analyses  
 5775 of system activity solely through the use of audit data.

EXAMPLE

examples of other input data include network datagrams, resource/accounting data, or combinations of various system data.

5776 The elements of FAU\_SAA.3 Simple attack heuristics do not require that the TSF implementing  
 5777 the immediate attack heuristics be the same TSF whose activity is being monitored. Thus, one  
 5778 **can** develop an intrusion detection component that operates independently of the system  
 5779 whose system activity is being analyzed.

## 5780 C.5.4.2 Operations

### 5781 C.5.4.2.1 Assignment

5782 In FAU\_SAA.3.1, the PP/ST author **should** identify a base subset of system events whose  
 5783 occurrence, in isolation from all other system activity, **may** indicate a violation of the  
 5784 enforcement of the SFRs. These include events that by themselves indicate a clear violation to  
 5785 the enforcement of the SFRs, or whose occurrence is so significant that they warrant actions.

5786 In FAU\_SAA.3.2, the PP/ST author **should** specify the information used to determine system  
 5787 activity. This information is the input data used by the analysis tool to determine the system  
 5788 activity that has occurred on the TOE. This data **may** include audit data, combinations of audit  
 5789 data with other system data, or **may** consist of data other than the audit data. The PP/ST author  
 5790 **should** define precisely what system events and event attributes are being monitored within the  
 5791 input data.

## 5792 C.5.5 FAU\_SAA.4 Complex attack heuristics

### 5793 C.5.5.1 User application notes

5794 In practice, it is at best rare when an analysis tool **can** detect with certainty when a security  
 5795 violation is imminent. However, there do exist some system events that are so significant they  
 5796 are always worthy of independent review.

EXAMPLE

Example of such events include the deletion of a key TSF security data file (such as the password file) or activity such as a remote user attempting to gain administrative privilege.

5797 These events are referred to as signature events in that their occurrence in isolation from the  
 5798 rest of the system activity are indicative of intrusive activity. Event sequences are an ordered  
 5799 set of signature events that might indicate intrusive activity.

5800 The complexity of a given tool will depend greatly on the assignments defined by the PP/ST  
 5801 author in identifying the base set of signature events and event sequences.

5802 The PP/ST author **should** enumerate specifically what events **should** be monitored by the TSF in  
 5803 order to perform the analysis. The PP/ST author **should** identify specifically what information  
 5804 pertaining to the event is necessary to determine if the event maps to a signature event.

5805 Administrative notification **should** be provided such that the authorized user understands the  
 5806 significance of the event and the appropriate possible responses.

An effort was made in the specification of these requirements to avoid a dependency on audit data as the sole input for monitoring system activity. This was done in recognition of the existence of previously developed intrusion detection tools that do not perform their analyses of system activity solely through the use of audit data

**EXAMPLE**

examples of other input data include network datagrams, resource/accounting data, or combinations of various system data

Levelling, therefore, requires the PP/ST author to specify the type of input data used to monitor system activity.

The elements of FAU\_SAA.4 Complex attack heuristics do not require that the TSF implementing the complex attack heuristics be the same TSF whose activity is being monitored. Thus, one **can** develop an intrusion detection component that operates independently of the system whose system activity is being analyzed.

## **C.5.5.2 Operations**

### **C.5.5.2.1 Assignment**

In FAU\_SAA.4.1, the PP/ST author **should** identify a base set of lists of sequences of system events whose occurrence are representative of known penetration scenarios. These event sequences represent known penetration scenarios. Each event represented in the sequence **should** map to a monitored system event, such that as the system events are performed, they are bound (mapped) to the known penetration event sequences.

In FAU\_SAA.4.1, the PP/ST author **should** identify a base subset of system events whose occurrence, in isolation from all other system activity, **may** indicate a violation of the enforcement of the SFRs. These include events that by themselves indicate a clear violation to the SFRs, or whose occurrence is so significant they warrant action.

In FAU\_SAA.4.2, the PP/ST author **should** specify the information used to determine system activity. This information is the input data used by the analysis tool to determine the system activity that has occurred on the TOE. This data **may** include audit data, combinations of audit data with other system data, or **may** consist of data other than the audit data. The PP/ST author **should** define precisely what system events and event attributes are being monitored within the input data.

## **C.6 Security audit review (FAU\_SAR)**

### **C.6.1 User notes**

The Security audit review family defines requirements related to review of the audit information.

These functions **should** allow pre-storage or post-storage audit selection.

**EXAMPLE**

An example of requirement related to review of the audit information is the ability to selectively review:

- the actions of one or more users (such as. identification, authentication, TOE entry, and access control actions);
- the actions performed on a specific object or TOE resource;
- all of a specified set of audited exceptions; or
- actions associated with a specific SFR attribute

The distinction between audit reviews is based on functionality. Audit review (only) encompasses the ability to view audit data. Selectable review is more sophisticated and

requires the ability to select subsets of audit data based on a single criterion or multiple criteria with logical (i.e. and/or) relations and order the audit data before it is reviewed.

## **C.6.2 FAU\_SAR.1 Audit review**

### **C.6.2.1 Rationale**

This component will provide authorized users the capability to obtain and interpret the information. In case of human users this information needs to be in a human understandable presentation. In case of external IT entities, the information needs to be unambiguously represented in an electronic fashion.

### **C.6.2.2 User application notes**

This component is used to specify that users and/or authorized users **can** read the audit records. These audit records will be provided in a manner appropriate to the user. There are different types of users (human users, machine users) that might have different needs.

The content of the audit records that **can** be viewed **can** be specified.

### **C.6.2.3 Operations**

#### **C.6.2.3.1 Assignment**

In FAU\_SAR.1.1, the PP/ST author **should** specify the authorized users that **can** use this capability. If appropriate the PP/ST author **may** include security roles (see FMT\_SMR.1 Security roles).

In FAU\_SAR.1.1, the PP/ST author **should** specify the type of information the specified user is permitted to obtain from the audit records.

#### EXAMPLE

Examples are “all”, “subject identity”, “all information belonging to audit records referencing this user”.

When employing the SFR, FAU\_SAR.1, it is not necessary to repeat, in full detail, the list of audit information first specified in FAU\_GEN.1. Use of terms such as “all” or “all audit information” assist in eliminating ambiguity and the further need for comparative analysis between the two security requirements.

## **C.6.3 FAU\_SAR.2 Restricted audit review**

### **C.6.3.1 User application notes**

This component specifies that any users not identified in FAU\_SAR.1 Audit review will not be able to read the audit records.

## **C.6.4 FAU\_SAR.3 Selectable audit review**

### **C.6.4.1 User application notes**

This component is used to specify that it **should** be possible to perform selection of the audit data to be reviewed. If based on multiple criteria, those criteria **should** be related together with logical (i.e. “and” or “or”) relations, and the tools **should** provide the ability to manipulate audit data

#### EXAMPLE

Means of manipulating audit data include sorting and filtering.

### **C.6.4.2 Operations**

#### **C.6.4.2.1 Assignment**

In FAU\_SAR.3.1, the PP/ST author **should** specify whether capabilities to select and/or order audit data is required from the TSF.



5880 In FAU\_SAR.3.1, the PP/ST author **should** assign the criteria, possibly with logical relations, to  
 5881 be used to select the audit data for review. The logical relations are intended to specify whether  
 5882 the operation **can** be on an individual attribute or a collection of attributes.

EXAMPLE

An example of this assignment could be: "application, user account and/or location".

5883 In this case, the operation **could** be specified using any combination of the three attributes:  
 5884 application, user account and location.

## 5885 **C.7 Security audit event selection (FAU\_SEL)**

### 5886 **C.7.1 User notes**

5887 The Security audit event selection family provides requirements related to the capabilities of  
 5888 identifying which of the possible auditable events are to be audited. The auditable events are  
 5889 defined in the Security audit data generation (FAU\_GEN) family, but those events **should** be  
 5890 defined as being selectable in this component to be audited.

5891 This family ensures that it is possible to keep the audit trail from becoming so large that it  
 5892 becomes useless, by defining the appropriate granularity of the selected security audit events.

### 5893 **C.7.2 FAU\_SEL.1 Selective audit**

#### 5894 **C.7.2.1 User application notes**

5895 This component defines the selection criteria used, and the resulting audited subsets of the set  
 5896 of all auditable events, based on user attributes, subject attributes, object attributes, or event  
 5897 types.

5898 The existence of individual user identities is not assumed for this component. This allows for  
 5899 TOEs such as routers that **may** not support the notion of users.

5900 For a distributed environment, the host identity **could** be used as a selection criterion for events  
 5901 to be audited.

5902 The management function FMT\_MTD.1 Management of TSF data will handle the rights of  
 5903 authorized users to query or modify the selections.

#### 5904 **C.7.2.2 Operations**

##### 5905 **C.7.2.2.1 Selection**

5906 In FAU\_SEL.1.1, the PP/ST author **should** select whether the security attributes upon which  
 5907 audit selectivity is based, is related to object identity, user identity, subject identity, host  
 5908 identity, or event type.

##### 5909 **C.7.2.2.2 Assignment**

5910 In FAU\_SEL.1.1, the PP/ST author **should** specify any additional attributes upon which audit  
 5911 selectivity is based. If there are no additional rules upon which audit selectivity is based, this  
 5912 assignment **can** be completed with "none".

## 5913 **C.8 Security audit data storage (FAU\_STG)**

### 5914 **C.8.1 User notes**

5915 The Security audit data storage family describes requirements for storing audit data for later  
 5916 use, including requirements controlling the loss of audit information due to TOE failure, attack  
 5917 and/or exhaustion of storage space.

### 5918 **C.8.2 FAU\_STG.1 Audit data storage location**

#### 5919 **C.8.2.1 User application notes**

#### 5920 **C.8.2.2 Operations**

5921 **C.8.2.2.1 Selection**5922 In FAU\_STG.1.1 the PP/ST author **should**5923 **C.8.2.2.2 Assignment**5924 In FAU\_STG.1.1 the PP/ST author **should**5925 **C.8.3 FAU\_STG.2 Protected audit data storage**5926 **C.8.3.1 User application notes**

5927 In a distributed environment, as the location of the audit trail is in the TSF, but not necessarily  
 5928 co-located with the function generating the audit data, the PP/ST author **could** request  
 5929 authentication of the originator of the audit record, or non-repudiation of the origin of the  
 5930 record prior storing this record in the audit trail.

5931 The TSF will protect the stored audit data in the audit trail from unauthorized deletion and  
 5932 modification. It is noted that in some TOEs the auditor (role) might not be authorized to delete  
 5933 the audit records for a certain period of time.

5934 **C.8.3.2 Operations**5935 **C.8.3.2.1 Selection**

5936 In FAU\_STG.2.2, the PP/ST author **should** specify whether the TSF **shall** prevent or only be able  
 5937 to detect modifications of the stored audit data in the audit trail. Only one of these options **may**  
 5938 be chosen.

5939 **C.8.4 FAU\_STG.3 Guarantees of audit data availability**5940 **C.8.4.1 User application notes**

5941 This component allows the PP/ST author to specify to which metrics the audit trail **should**  
 5942 conform.

5943 In a distributed environment, as the location of the audit trail is in the TSF, but not necessarily  
 5944 co-located with the function generating the audit data, the PP/ST author **could** request  
 5945 authentication of the originator of the audit record, or non-repudiation of the origin of the  
 5946 record prior storing this record in the audit trail.

5947 **C.8.4.2 Operations**5948 **C.8.4.2.1 Assignment**

5949 In FAU\_STG.3.1, the PP/ST author **should** specify the metric that the TSF must ensure with  
 5950 respect to the stored audit records. This metric limits the data loss by enumerating the number  
 5951 of records that must be kept, or the time that records are guaranteed to be maintained.

**EXAMPLE**

An example of the metric **could** be "100,000" indicating that 100,000 audit records **can** be stored.

5952 **C.8.4.2.2 Selection**

5953 In FAU\_STG.3.1, the PP/ST author **should** specify the condition under which the TSF **shall** still be  
 5954 able to maintain a defined amount of audit data. This condition **can** be any of the following:  
 5955 audit storage exhaustion, failure, attack.

5956 **C.8.5 FAU\_STG.4 Prevention of audit data loss**5957 **C.8.5.1 User application notes**

5958 This component specifies the behaviour of the TOE if the audit trail is full: either audit records  
 5959 are ignored, or the TOE is frozen such that no audited events **can** take place. The requirement  
 5960 also states that no matter how the requirement is instantiated, the authorized user with specific  
 5961 rights to this effect, **can** continue to generate audited events (actions). The reason is that  
 5962 otherwise the authorized user **could** not even reset the TOE. Consideration **should** be given to



the choice of the action to be taken by the TSF in the case of audit storage exhaustion, as ignoring events, which provides better availability of the TOE, will also permit actions to be performed without being recorded and without the user being accountable.

## **C.8.5.2 Operations**

### **C.8.5.2.1 Selection**

In FAU\_STG.4.1, the PP/ST author **should** select whether the TSF **shall** ignore audited actions, or whether it **should** prevent audited actions from happening, or whether the oldest audit records **should** be overwritten when the TSF **can** no longer store audit records. Only one of these options **may** be chosen.

### **C.8.5.2.2 Assignment**

In FAU\_STG.4.1, the PP/ST author **should** specify other actions that **should** be taken in case of audit storage failure, such as informing the authorized user. If there is no other action to be taken in case of audit storage failure, this assignment **can** be completed with “none”.

## **C.8.6 FAU\_STG.5 Action in case of possible audit data loss**

### **C.8.6.1 User application notes**

This component requires that actions will be taken when the audit trail exceeds certain pre-defined limits.

## **C.8.6.2 Operations**

### **C.8.6.2.1 Assignment**

In FAU\_STG.5.1, the PP/ST author **should** indicate the pre-defined limit. If the management functions indicate that this number might be changed by the authorized user, this value is the default value. The PP/ST author might choose to let the authorized user define this limit.

#### **EXAMPLE**

In that case, the assignment can be “an authorized user set limit”.

In FAU\_STG.5.1, the PP/ST author **should** specify actions that **should** be taken in case of imminent audit storage failure indicated by exceeding the threshold. Actions might include informing an authorized user.

## Annex D (normative)

### Class FCO: Communication- application notes

#### D.1 General information

This class describes requirements specifically of interest for TOEs that are used for the transport of information. Families within this class deal with non-repudiation.

In this class, the concept of “information” is used. This information **should** be interpreted as the object being communicated, and **could** contain an electronic mail message, a file, or a set of predefined attribute types.

In the literature, the terms “proof of receipt” and “proof of origin” are commonly used terms. However, it is recognized that the term “proof” might be interpreted in a legal sense to imply a form of mathematical rationale. The components in this class interpret the de-facto use of the word “proof” in the context of “evidence” that the TSF demonstrates the non-repudiated transport of types of information.

#### D.2 Non-repudiation of origin (FCO\_NRO)

##### D.2.1 User notes

Non-repudiation of origin defines requirements to provide evidence to users/subjects about the identity of the originator of some information. The originator cannot successfully deny having sent the information because evidence of origin provides evidence of the binding between the originator and the information sent. The recipient or a third party **can** verify the evidence of origin. This evidence **should** not be forgeable.

###### EXAMPLE 1

Evidence of origin could be a digital signature

If the information or the associated attributes are altered in any way, validation of the evidence of origin might fail. Therefore, a PP/ST author **should** consider including integrity requirements such as FDP\_UIT.1 Data exchange integrity in the PP/ST.

In non-repudiation, there are several different roles involved, each of which **could** be combined in one or more subjects. The first role is a subject that requests evidence of origin (only in FCO\_NRO.1 Selective proof of origin). The second role is the recipient and/or other subjects to which the evidence is provided. The third role is a subject that requests verification of the evidence of origin.

###### EXAMPLE 2

Subject that requests evidence of origin: a recipient or a third party such as an arbiter.

Subject to which the evidence is provided: A notary

The PP/ST author must specify the conditions that must be met to be able to verify the validity of the evidence.

###### EXAMPLE 3

An example of a condition which **could** be specified is where the verification of evidence must occur within 24 hours.

These conditions, therefore, allow the tailoring of the non-repudiation to legal requirements, such as being able to provide evidence for several years.

In most cases, the identity of the recipient will be the identity of the user who received the transmission. In some instances, the PP/ST author does not want the user identity to be exported. In that case, the PP/ST author must consider whether it is appropriate to include this

6025 class, or whether the identity of the transport service provider or the identity of the host **should**  
 6026 be used.

6027 In addition to (or instead of) the user identity, a PP/ST author might be more concerned about  
 6028 the time the information was transmitted.

EXAMPLE

For example, requests for proposals must be transmitted before a certain date in order to be considered.

6029 In such instances, these requirements **can** be customized to provide a timestamp indication  
 6030 (time of origin).

## 6031 **D.2.2 FCO\_NRO.1 Selective proof of origin**

### 6032 **D.2.2.1 Operations**

#### 6033 **D.2.2.1.1 Assignment**

6034 In FCO\_NRO.1.1, the PP/ST author **should** fill in the types of information subject to the evidence  
 6035 of origin function.

EXAMPLE

An example of the type of information is “electronic mail messages”

6036

#### 6037 **D.2.2.1.2 Selection**

6038 In FCO\_NRO.1.1, the PP/ST author **should** specify the user/subject who **can** request evidence of  
 6039 origin.

#### 6040 **D.2.2.1.3 Assignment**

6041 In FCO\_NRO.1.1, the PP/ST author, dependent on the selection, **should** specify the third parties  
 6042 that **can** request evidence of origin.

EXAMPLE

A third party **could** be an arbiter, judge, or legal body.

6043 In FCO\_NRO.1.2, the PP/ST author **should** fill in the list of the attributes that **shall** be linked to  
 6044 the information;

EXAMPLE

Attributes include originator identity, time of origin, and location of origin.

6045 In FCO\_NRO.1.2, the PP/ST author **should** fill in the list of information fields within the  
 6046 information over which the attributes provide evidence of origin, such as the body of a message.

#### 6047 **D.2.2.1.4 Selection**

6048 In FCO\_NRO.1.3, the PP/ST author **should** specify the user/subject who **can** verify the evidence  
 6049 of origin.

#### 6050 **D.2.2.1.5 Assignment**

6051 In FCO\_NRO.1.3, the PP/ST author **should** fill in the list of limitations under which the evidence  
 6052 **can** be verified.

EXAMPLE

An example of a limitation is “the evidence **can** only be verified within a 24-hour time interval.”

6053 An assignment of “immediate” or “indefinite” is acceptable.

6054 In FCO\_NRO.1.3, the PP/ST author, dependent on the selection, **should** specify the third parties  
 6055 that **can** verify the evidence of origin.

## 6056 **D.2.3 FCO\_NRO.2 Enforced proof of origin**

6057 **D.2.3.1 Operations**6058 **D.2.3.1.1 Assignment**

6059 In FCO\_NRO.2.1, the PP/ST author **should** fill in the types of information subject to the evidence  
6060 of origin function.

## EXAMPLE

electronic mail messages.

6061 In FCO\_NRO.2.2, the PP/ST author **should** fill in the list of the attributes that **shall** be linked to  
6062 the information; for example, originator identity, time of origin, and location of origin.

6063 In FCO\_NRO.2.2, the PP/ST author **should** fill in the list of information fields within the  
6064 information over which the attributes provide evidence of origin, such as the body of a message.

6065 **D.2.3.1.2 Selection**

6066 In FCO\_NRO.2.3, the PP/ST author **should** specify the user/subject who **can** verify the evidence  
6067 of origin.

6068 **D.2.3.1.3 Assignment**

6069 In FCO\_NRO.2.3, the PP/ST author **should** fill in the list of limitations under which the evidence  
6070 **can** be verified.

## EXAMPLE

For example, the evidence **can** only be verified within a 24-hour time interval

6071 An assignment of “immediate” or “indefinite” is acceptable.

6072 In FCO\_NRO.2.3, the PP/ST author, dependent on the selection, **should** specify the third parties  
6073 that **can** verify the evidence of origin.

## EXAMPLE

A third party **could** be an arbiter, judge, or legal body.

6074 **D.3 Non-repudiation of receipt (FCO\_NRR)**6075 **D.3.1 User notes**

6076 Non-repudiation of receipt defines requirements to provide evidence to other users/subjects  
6077 that the information was received by the recipient. The recipient cannot successfully deny  
6078 having received the information because evidence of receipt provides evidence of the binding  
6079 between the recipient attributes and the information. The originator or a third party **can** verify  
6080 the evidence of receipt. This evidence **should** not be forgeable.

## EXAMPLE

An example of a receipt is a digital signature

6081 It **should** be noted that the provision of evidence that the information was received does not  
6082 necessarily imply that the information was read or comprehended, but only delivered.

6083 If the information or the associated attributes are altered in any way, validation of the evidence  
6084 of receipt with respect to the original information might fail. Therefore, a PP/ST author **should**  
6085 consider including integrity requirements such as FDP\_UIT.1 Data exchange integrity in the  
6086 PP/ST.

6087 In non-repudiation, there are several different roles involved, each of which **could** be combined  
6088 in one or more subjects. The first role is a subject that requests evidence of receipt (only in  
6089 FCO\_NRR.1 Selective proof of receipt). The second role is the recipient and/or other subjects to  
6090 which the evidence is provided). The third role is a subject that requests verification of the  
6091 evidence of receipt, for example, an originator or a third party such as an arbiter.

## EXAMPLE

A recipient or subject **could** be a notary.

6092

6093 The PP/ST author must specify the conditions that must be met to be able to verify the validity  
6094 of the evidence. An example of a condition which **could** be specified is where the verification of  
6095 evidence must occur within 24 hours. These conditions, therefore, allow the tailoring of the  
6096 non-repudiation to legal requirements, such as being able to provide evidence for several years.

6097 In most cases, the identity of the recipient will be the identity of the user who received the  
6098 transmission. In some instances, the PP/ST author does not want the user identity to be  
6099 exported. In that case, the PP/ST author must consider whether it is appropriate to include this  
6100 class, or whether the identity of the transport service provider or the identity of the host **should**  
6101 be used.

6102 In addition to (or instead of) the user identity, a PP/ST author might be more concerned about  
6103 the time the information was received.

EXAMPLE

When an offer expires at a certain date, orders must be received before a certain date in order to be considered.

6104 In such instances, these requirements **can** be customized to provide a timestamp indication  
6105 (time of receipt).

### 6106 **D.3.2 FCO\_NRR.1 Selective proof of receipt**

#### 6107 **D.3.2.1 Operations**

##### 6108 **D.3.2.1.1 Assignment**

6109 In FCO\_NRR.1.1, the PP/ST author **should** fill in the types of information subject to the evidence  
6110 of receipt function, for example, electronic mail messages.

##### 6111 **D.3.2.1.2 Selection**

6112 In FCO\_NRR.1.1, the PP/ST author **should** specify the user/subject who **can** request evidence of  
6113 receipt.

##### 6114 **D.3.2.1.3 Assignment**

6115 In FCO\_NRR.1.1, the PP/ST author, dependent on the selection, **should** specify the third parties  
6116 that **can** request evidence of receipt.

EXAMPLE

A third party **could** be an arbiter, judge, or legal body.

6117 In FCO\_NRR.1.2, the PP/ST author **should** fill in the list of the attributes that **shall** be linked to  
6118 the information; for example, recipient identity, time of receipt, and location of receipt.

6119 In FCO\_NRR.1.2, the PP/ST author **should** fill in the list of information fields with the fields  
6120 within the information over which the attributes provide evidence of receipt, such as the body a  
6121 message.

##### 6122 **D.3.2.1.4 Selection**

6123 In FCO\_NRR.1.3, the PP/ST author **should** specify the user/subjects who **can** verify the evidence  
6124 of receipt.

##### 6125 **D.3.2.1.5 Assignment**

6126 In FCO\_NRR.1.3, the PP/ST author **should** fill in the list of limitations under which the evidence  
6127 **can** be verified. For example, the evidence **can** only be verified within a 24-hour time interval.  
6128 An assignment of “immediate” or “indefinite” is acceptable.

6129 In FCO\_NRR.1.3, the PP/ST author, dependent on the selection, **should** specify the third parties  
6130 that **can** verify the evidence of receipt.

6131 **D.3.3 FCO\_NRR.2 Enforced proof of receipt**

6132 **D.3.3.1 Operations**

6133 **D.3.3.1.1 Assignment**

6134 In FCO\_NRR.2.1, the PP/ST author **should** fill in the types of information subject to the evidence  
6135 of receipt function,

EXAMPLE

for example, electronic mail messages.

6136 In FCO\_NRR.2.2, the PP/ST author **should** fill in the list of the attributes that **shall** be linked to  
6137 the information;

EXAMPLE

for example, recipient identity, time of receipt, and location of receipt.

6138 In FCO\_NRR.2.2, the PP/ST author **should** fill in the list of information fields with the fields  
6139 within the information over which the attributes provide evidence of receipt, such as the body  
6140 of a message.

6141 **D.3.3.1.2 Selection**

6142 In FCO\_NRR.2.3, the PP/ST author **should** specify the user/subjects who **can** verify the evidence  
6143 of receipt.

6144 **D.3.3.1.3 Assignment**

6145 In FCO\_NRR.2.3, the PP/ST author **should** fill in the list of limitations under which the evidence  
6146 **can** be verified. An assignment of “immediate” or “indefinite” is acceptable.

EXAMPLE

For example, the evidence **can** only be verified within a 24-hour time interval.

6147 In FCO\_NRR.2.3, the PP/ST author, dependent on the selection, **should** specify the third parties  
6148 that **can** verify the evidence of receipt. A third party **could** be an arbiter, judge or legal body.

6149 **~~D.4 Trusted channel (FCO\_TCO)~~**

6150 **~~D.4.1 User notes~~**

6151 **~~D.4.2 FCO\_TCO.1 Trusted communication channel with fixed security properties~~**

6152 **~~D.4.2.1 Operations~~**

6153 **~~D.4.2.1.1 Assignment~~**

6154 ~~In FCO\_TCC.1.1 the PP/ST author **should**~~

6155 ~~In FCO\_TCC.1.4, the PP/ST author **should**~~

6156 **~~D.4.2.1.2 Selection~~**

6157 ~~In FCO\_TCC.1.2, the PP/ST author **should**~~

6158 ~~In FCO\_TCC.1.3, the PP/ST author **should**~~

6159 **~~D.4.3 FCO\_TCO.2 Trusted communication channel with selectable security~~**  
6160 **~~properties~~**

6161 **~~D.4.3.1 Operations~~**

6162 **~~D.4.3.1.1 Assignment~~**

6163 ~~In FCO\_TCC.2.1, the PP/ST author **should**~~

6164 ~~In FCO\_TCC.2.7, the PP/ST author **should**~~

6165 **~~D.4.3.1.2~~ Selection**6166 ~~In FCO\_TCC.2.2, the PP/ST author~~ **should**6167 ~~In FCO\_TCC.2.3, the PP/ST author~~ **should**6168 **~~17.4.5.5~~ In FCO\_TCC.2.4**

6169 The TSF **shall** implement the trusted channel in compliance with the following security standards  
6170 {assignment: *list of security standards or none*} using the following options {selection: {assignment:  
6171 *list of options*}, *none*}.

6172 ~~FCO\_TCC.2.5, the PP/ST author~~ **should**6173 ~~In FCO\_TCC.2.6, the PP/ST author~~ **should**6174 ~~In FCO\_TCC.2.7, the PP/ST author~~ **should**



## Annex E (normative)

### Class FCS: Cryptographic support- application notes

#### E.1 General information

The TSF **may** employ cryptographic functionality to help satisfy several high-level security objectives. These include (but are not limited to): identification and authentication, non-repudiation, trusted path, trusted channel and data separation. This class is used when the TOE implements cryptographic functions, the implementation of which **could** be in hardware, firmware and/or software.

The FCS: Cryptographic support class is composed of four families: Cryptographic key management (FCS\_CKM), Cryptographic operation (FCS\_COP), Random bit generation (FCS\_RBG), and Generation of random numbers (FCS\_RNG). The Cryptographic key management (FCS\_CKM) family addresses the management aspects of cryptographic keys; the Cryptographic operation (FCS\_COP) family is concerned with the operational use of those cryptographic keys; the Random bit generation (FCS\_RBG) family provides requirements for generating random bits; and the Generation of random numbers (FCS\_RNG) is concerned with ensuring that random numbers meet defined quality metrics.

For each cryptographic key generation method implemented by the TOE, if any, the PP/ST author **should** select the FCS\_CKM.1 Cryptographic key generation component.

For each cryptographic key derivation method implemented by the TOE, if any, the PP/ST author **should** select the FCS\_CKM.5 Cryptographic key derivation.

For each cryptographic key distribution method implemented by the TOE, if any, the PP/ST author **should** select the FCS\_CKM.2 Cryptographic key distribution.

For each cryptographic key access method implemented by the TOE, if any, the PP/ST author **should** select the FCS\_CKM.3 Cryptographic key access.

For each cryptographic key destruction method implemented by the TOE, if any, the PP/ST author **should** select the FCS\_CKM.6 Timing and event of cryptographic key destruction component.

For each cryptographic operation (such as digital signature, data encryption, key agreement, secure hash, etc.) performed by the TOE, if any, the PP/ST author **should** select the FCS\_COP.1 Cryptographic operation component.

Cryptographic functionality **may** be used to meet objectives specified in class FCO: Communication, and in families Data authentication (FDP\_DAU), Stored data integrity (FDP\_SDI), Inter-TSF user data confidentiality transfer protection (FDP\_UCT), Inter-TSF user data integrity transfer protection (FDP\_UIT), Specification of secrets (FIA\_SOS), User authentication (FIA\_UAU), to meet a variety of objectives. In the cases where cryptographic functionality is used to meet objectives for other classes, the individual functional components specify the objectives that cryptographic functionality must satisfy. The objectives in class FCS: Cryptographic support **should** be used when cryptographic functionality of the TOE is sought by consumers.

#### E.2 Cryptographic key management (FCS\_CKM)

##### E.2.1 User notes

Cryptographic keys must be managed throughout their lifetime. The typical events in the lifecycle of a cryptographic key include but are not limited to: key generation or derivation, distribution, entry, storage, access, and destruction.



## EXAMPLE

- backup
- escrow
- archive
- recovery

6221 The inclusion of other stages is dependent on the key management strategy being implemented,  
6222 as the TOE is not always involved in all of the key life-cycle phases.

## EXAMPLE

The TOE **may** only generate and distribute cryptographic keys.

6223 This family is intended to support the cryptographic key lifecycle and consequently defines  
6224 requirements for the following activities: cryptographic key generation, cryptographic key  
6225 derivation, cryptographic key distribution, cryptographic key access, and cryptographic key  
6226 destruction. This family **should** be included whenever there are functional requirements for the  
6227 management of cryptographic keys.

6228 If Security audit data generation (FAU\_GEN) is included in the PP/ST then, in the context of the  
6229 events being audited:

6230 a) The object attributes **may** include the assigned user for the cryptographic key, the  
6231 user role, the cryptographic operation that the cryptographic key is to be used for,  
6232 the cryptographic key identifier and the cryptographic key validity period.

6233 b) The object value **may** include the values of cryptographic key(s) and parameters  
6234 excluding any sensitive information (such as secret or private cryptographic keys).

6235 Typically, random numbers are used to generate cryptographic keys. If this is the case, then  
6236 FCS\_CKM.1 Cryptographic key generation **should** be used instead of the component FIA\_SOS.2  
6237 TSF Generation of secrets. In cases where random number generation is required for purposes  
6238 other than for the generation of cryptographic keys, the component FIA\_SOS.2 TSF Generation  
6239 of secrets **should** be used.

## 6240 E.2.2 FCS\_CKM.1 Cryptographic key generation

### 6241 E.2.2.1 User application notes

6242 This component requires the cryptographic key sizes and method used to generate  
6243 cryptographic keys to be specified, this **can may** be in accordance with an assigned standard. It  
6244 **should** be used to specify the cryptographic key sizes and the method used to generate the  
6245 cryptographic keys. Only one instance of the component is needed for the same method and  
6246 multiple key sizes. The key size **could may** be common or different for the various entities and  
6247 **could may** be either the input to or the output from the method.

## EXAMPLE

An example of a method is an algorithm.

### 6248 E.2.2.2 Operations

#### 6249 E.2.2.2.1 Assignment

6250 In FCS\_CKM.1.1, the PP/ST author **should** specify the cryptographic key generation algorithm to  
6251 be used.

6252 In FCS\_CKM.1.1, the PP/ST author **should** specify the cryptographic key sizes to be used. The  
6253 key sizes specified **should** be appropriate for the algorithm and its intended use.

6254 In FCS\_CKM.1.1, the PP/ST author **should** specify the assigned standard that documents the  
6255 method used to generate cryptographic keys. The assigned standard **may** comprise none, one or

6256 more actual standards publications, for example, from international, national, industry or  
6257 organizational standards.

### 6258 **E.2.3 FCS\_CKM.2 Cryptographic key distribution**

#### 6259 **E.2.3.1 User application notes**

6260 This component requires the method used to distribute cryptographic keys to be specified, this  
6261 ~~can~~ **may** be in accordance with an assigned standard. See ISO/IEC 15408-1 for information on  
6262 using standards in PPs and STs.

#### 6263 **E.2.3.2 Operations**

##### 6264 **E.2.3.2.1 Assignment**

6265 In FCS\_CKM.2.1 the PP/ST author **should** specify the cryptographic key distribution method to  
6266 be used.

6267 In FCS\_CKM.2.1 the PP/ST author **should** specify the assigned standard that documents the  
6268 method used to distribute cryptographic keys. The assigned standard **may** comprise none, one  
6269 or more actual standards publications, for example, from international, national, industry or  
6270 organizational standards.

### 6271 **E.2.4 FCS\_CKM.3 Cryptographic key access**

#### 6272 **E.2.4.1 User application notes**

6273 This component requires the method used to access cryptographic keys be specified, this ~~can~~  
6274 **may** be in accordance with an assigned standard.

#### 6275 **E.2.4.2 Operations**

##### 6276 **E.2.4.2.1 Assignment**

6277 In FCS\_CKM.2.1, the PP/ST author **should** specify the type of cryptographic key access being  
6278 used.

#### EXAMPLE

Examples of types of cryptographic key access include (but are not limited to) cryptographic key backup, cryptographic key archival, cryptographic key escrow, and cryptographic key recovery.

6279 In FCS\_CKM.2.1, the PP/ST author **should** specify the cryptographic key access method to be  
6280 used.

6281 In FCS\_CKM.2.1, the PP/ST author **should** specify the assigned standard that documents the  
6282 method used to access cryptographic keys. The assigned standard **may** comprise none, one or  
6283 more actual standards publications, for example, from international, national, industry or  
6284 organizational standards.

### 6285 **E.2.5 FCS\_CKM.5 Cryptographic key derivation**

#### 6286 **E.2.5.1 User application notes**

6287 Table E.1 **should** be used when completing and potentially iterating the FCS-CKM.5 component.  
6288 Each row, which can be identified using the “Identifier”, provides a set of recommended  
6289 selections and assignments for completing FCS-CKM.5 for each commonly used key type.

6290 **Table E.1 — Recommended selections and assignments for key derivation**

Identifier	key type	input parameters	key derivation algorithm	key sizes	list of standards
KeyDrv1	[assignment: key name]	Direct Generation from a Random Bit Generator as	KDF in Counter Mode using [selection: CMAC-AES-	[selection: 128, 256] bits	NIST SP 800-108 (Section 5.1) [KDF in Counter Mode]

Identifier	key type	input parameters	key derivation algorithm	key sizes	list of standards
		specified in FCS_RBG_EXT.1	128, CMAC-AES-256, HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512] as the PRF		[selection: ISO/IEC9797-1(Clauses B.6, B.7), NIST SP800-38B) [CMAC]  ISO/IEC 18033-3:2010 (Subclause 5.2) [AES], ISO/IEC 9797-2:2011 (Clause 7 MAC Algorithm 2 (HMAC)), FIPS 198-1, ISO10118-3, (Clause 10, 11); FIPS180-4, (Section 6) [SHA]]
KeyDrv2	[assignment: key name]	Direct Generation from a Random Bit Generator as specified in FCS_RBG_EXT.1	KDF in Feedback Mode using [selection: CMAC-AES-128, CMAC-AES-192, CMAC-AES-256, HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512] as the PRF	[selection: 128, 256] bits	NIST SP 800-108 (Section 5.2) [KDF in Feedback Mode]  [selection: ISO/IEC9797-1 (Subclause 7.6), NIST SP800-38B) [CMAC]  ISO/IEC 18033-3:2010 (Subclause 5.2) [AES], ISO/IEC 9797-2:2011 (Clause 7 MAC Algorithm 2 (HMAC)), FIPS 198-1, ISO10118-3, (Clause 10, 11); FIPS180-4, (Section 6) [SHA]]
KeyDrv3	[assignment: key name]	Direct Generation from a Random Bit Generator as specified in FCS_RBG_EXT.1	KDF in Double-Pipeline Iteration Mode using [selection: CMAC-AES-128, CMAC-AES-256, HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512] as the PRF	[selection: 128, 256] bits	NIST SP 800-108 (Section 5.3) [KDF in Double-Pipeline Iteration Mode]  [selection: ISO/IEC9797-1 (Subclause 7.6), NIST SP800-38B) [CMAC]  ISO/IEC 18033-3:2010 (subclause 5.2) [AES], ISO/IEC 9797-2:2011 (Clause 7 MAC Algorithm 2 (HMAC)), FIPS 198-1, ISO/IEC 10118-3, (Clause 10, 11); FIPS180-4, (Section 6) [SHA]]
KeyDrv4	Authorization Factor Submask	Password Salt: using a salt as specified in FCS_SLT_EXT.1	PBKDF using HMAC- [selection: SHA-1, SHA-256, SHA-512] as the PRF, with	[selection: 128, 256] bits	NIST SP 800-132

Identifier	key type	input parameters	key derivation algorithm	key sizes	list of standards
			[assignment: positive integer of 1000 or more] iterations		
KeyDrv5	[assignment: key name]	Intermediary keys	[selection: exclusive OR (XOR), SHA-256, SHA-512]	[selection: 128, 256] bits	[selection: ISO 10118-3, (Clause 10, 11); FIPS180-4, (Section 6) [SHA]]

6291

6292 NOTE For identifier KeyDrv4, The key size to be used in the HMAC falls into a range between L1 and L2 defined  
 6293 in ISO/IEC 10118 for the appropriate hash function (for example for SHA-256 L1 = 512, L2 = 256) where  $L2 \leq k \leq L1$ .

6294 **Editors' Note:**

6295 Is there a specific part of ISO/IEC 10118 that is applicable here? Also, if the parameters in the standard  
 6296 could possibly be updated then we should specify the date of the ISO/IEC 10118 edition that applies here.

6297 Similarly, we **may** need to specify dates/revisions for each of the standards given in the table.

**EXAMPLE**

To derive a component or SFR from the FCS\_CKM.5 component for Intermediary keys, the row identified as KeyDrv 5 in Table E.1 is used.

Using this information, the following component is generated:

The TSF shall derive cryptographic keys [assignment: *key type*] from [Intermediary keys] in accordance with a specified cryptographic key derivation algorithm [**selection: *exclusive OR (XOR), SHA-256, SHA-512***] and specified cryptographic key sizes [**selection: *128, 256 bits***] that meet the following: [**selection: *ISO 10118-3, (Clause 10, 11); FIPS180-4, (Section 6) [SHA]***].

This component can then be used in PPs or completed and used as an SFR in PPs and STs, as appropriate.

6298

6299 **Editors' Note**

6300 The Editors have attempted to provide this example. **Please review!**

### 6301 E.2.5.2 Evaluator notes

6302 Evaluators should refer to ISO/IEC 15408:20XX Annex A.4.8 for information in regard to the  
 6303 evaluation of standards specified in FCS\_CKM.5.

### 6304 E.2.5.3 Operations

#### 6305 E.2.5.3.1 Assignment

6306 See E.2.5.1.

#### 6307 E.2.5.3.2 Selection

6308 See E.2.5.1.

### 6309 E.2.6 FCS\_CKM.6 Timing and event of cryptographic key destruction

#### 6310 E.2.6.1 User application notes

6311 This component requires the list of keys, including any keying material and the method used to  
 6312 destroy cryptographic keys to be specified, this can be in accordance with an assigned standard.

6313 NOTE Key material includes keys and initialization vectors necessary to establish and maintain cryptographic  
 6314 keying relationships

6315 **E.2.6.2 Operations**6316 **E.2.6.2.1 Assignment**6317 **E.2.6.2.2 Selection**6318 **E.3 Cryptographic operation (FCS\_COP)**6319 **E.3.1 User notes**

6320 A cryptographic operation **may** have cryptographic mode(s) of operation associated with it. If  
 6321 this is the case, then the cryptographic mode(s) must be specified.

**EXAMPLE**

Examples of cryptographic modes of operation are cipher block chaining, output feedback mode, electronic code book mode, and cipher feedback mode.

6322 Cryptographic operations **may** be used to support one or more TOE security services. The  
 6323 Cryptographic operation (FCS\_COP) component **may** need to be iterated more than once  
 6324 depending on:

- 6325 a) the user application for which the security service is being used,
- 6326 b) the use of different cryptographic algorithms and/or cryptographic key sizes,
- 6327 c) the type or sensitivity of the data being operated on.

6328 If Security audit data generation (FAU\_GEN) Security audit data generation is included in the  
 6329 PP/ST then, in the context of the cryptographic operation events being audited:

- 6330 a) The types of cryptographic operation **may** include digital signature generation  
 6331 and/or verification, cryptographic checksum generation for integrity and/or for  
 6332 verification of checksum, secure hash (message digest) computation, data  
 6333 encryption and/or decryption, cryptographic key encryption and/or decryption,  
 6334 cryptographic key agreement, and random number generation.
- 6335 b) The subject attributes **may** include subject role(s) and user(s) associated with the  
 6336 subject.
- 6337 c) The object attributes **may** include the assigned user for the cryptographic key, user  
 6338 role, cryptographic operation the cryptographic key is to be used for, cryptographic  
 6339 key identifier, and the cryptographic key validity period.

6340 **E.3.2 FCS\_COP.1 Cryptographic operation**6341 **E.3.2.1 User application notes**

6342 This component requires the cryptographic algorithm and key size used to perform specified  
 6343 cryptographic operation(s) which **can** be based on an assigned standard.

6344 **E.3.2.2 Operations**6345 **E.3.2.2.1 Assignment**

6346 In FCS\_COP.1.1, the PP/ST author ~~should~~ **shall** specify the cryptographic operations being  
 6347 performed. Typical cryptographic operations include digital signature generation and/or  
 6348 verification, cryptographic checksum generation for integrity and/or for verification of  
 6349 checksum, secure hash (message digest) computation, data encryption and/or decryption,  
 6350 cryptographic key encryption and/or decryption, cryptographic key agreement, and random  
 6351 number generation. The cryptographic operation **may** be performed on user data or TSF data.

6352 In FCS\_COP.1.1, the PP/ST author **should** specify the cryptographic algorithm to be used.

**EXAMPLE**

Examples of typical cryptographic algorithms include, but are not limited to, DES, RSA and IDEA.

6353 In FCS\_COP.1.1, the PP/ST author **should** specify the cryptographic key sizes to be used. The key  
6354 sizes specified **should** be appropriate for the algorithm and its intended use.

6355 In FCS\_COP.1.1, the PP/ST author **should** specify the assigned standard that documents how the  
6356 identified cryptographic operation(s) are performed. The assigned standard **may** comprise  
6357 none, one or more actual standards publications, these **may** include standards from  
6358 international, national, industry or organizational standards.

## 6359 **E.4 Random bit generation (FCS\_RBG)**

### 6360 **E.4.1 User notes**

### 6361 **E.4.2 FCS\_RBG.1 Random bit generation (RBG)**

#### 6362 **E.4.2.1 User application notes**

6363 For FCS\_RBG.1 These dependencies **shall** always be met.

6364 NOTE ISO/IEC 15408-1:20XX 7.3 item c) allowing a justification to be provided if a dependency is not met is not  
6365 allowed for this component.

6366 In the RBG State Update Table the ST author **must** include a row for initialization (Source1).  
6367 Other rows are optional, depending on the noise sources supported by the TSF. The identifier  
6368 values identify the specific source, so there should be a row for every unique source, and if the  
6369 same source is used for more than one update type then the same identifier is given.

6370 If reseeding is not feasible, the TSF will unstantiate RBGs (and instantiate a new RBG), rather  
6371 than produce output that is of insufficient quality. The listed standards should specify the  
6372 reseed interval, and procedure for uninstantiating and reseeding. The 'Condition' selection  
6373 allows the PP Author to require application-specific conditions for reseeding.

6374 "Unstantiate" means that the internal state of the DRBG is no longer available for use.

6375 In the 'Condition' selection, "on demand" means, that an interface to reseed is presented as a  
6376 TSFI

#### EXAMPLE

An example of a n interface is an API call.

6377 Health tests for the RBG are specified in FPT\_TST.1.

#### 6378 **E.4.2.2 Operations**

##### 6379 **E.4.2.2.1 Selection**

##### 6380 **E.4.2.2.2 Assignment**

### 6381 **E.4.3 FCS\_RBG.2 Random bit generation (external seeding)**

#### 6382 **E.4.3.1 User application notes**

6383 For this component, the interface to obtain the entropy noise source can be used multiple times  
6384 to provide input. For instance, if the input length is 128 bits, it could be used twice to gather 256  
6385 bits. In this instance, the 128 bits would not be provided to the DRBG, since the DRBG can only  
6386 be instantiated once, rather a function would gather the 128 bits twice and provide the DRBG  
6387 with 256 bits of entropy noise source.

6388 This component does not describe requirements on seed quality: it is the responsibility of the  
6389 operational environment to define their requirement in this regard and to ensure that it is met  
6390 by the external source.

6391 Guidance in the introduction to PP/ST authors should address protection from modification and  
6392 disclosure of the value from the external noise source, as well as the leaking of any pertinent  
6393 information (e.g., internal state) regarding the RBG.

6394 **Editors' Note**

6395 Please provide an exact reference to what is meant by “Guidance in the introduction to PP/ST authors”.  
 6396 Does it mean the “Introduction Section” of the PP/ST ? In that case a reference would be See ISO/IEC  
 6397 15408-1:20XX, B.2.2.1

#### 6398 **E.4.3.2 Operations**

##### 6399 **E.4.3.2.1 Selection**

##### 6400 **E.4.3.2.2 Assignment**

#### 6401 **E.4.4 FCS\_RBG.3 Random bit generation (internal seeding – single source)**

##### 6402 **E.4.4.1 User application notes**

6403 If an ST Author wishes to use multiple internal noise sources, they iterate this requirement for  
 6404 each noise source being used by the TSF.

6405 Hardware-based noise sources are sources whose primary function is noise generation, such as  
 6406 ring oscillators, diodes, and thermal noise. While software is used to collect the noise from these  
 6407 hardware sources, these are not software-based. Software-based noise sources are those  
 6408 sources that have some other primary function and the noise is a byproduct of their normal  
 6409 operation. Examples of software-based noise sources are user or system-based events, reading  
 6410 the least significant bits from an event timer, etc.

6411 Hardware-based noise sources may be stochastically modelled, in which case the amount of  
 6412 entropy is well understood. Software-based noise sources are usually less well understood and  
 6413 therefore will typically take a more conservative approach, gathering larger numbers of bits  
 6414 than required and then performing a compression function to derive the final output. Software-  
 6415 based noise sources often rely on an entropy estimator.

##### 6416 **E.4.4.2 Operations**

##### 6417 **E.4.4.2.1 Selection**

##### 6418 **E.4.4.2.2 Assignment**

#### 6419 **E.4.5 FCS\_RBG.4 Random bit generation**

##### 6420 **E.4.5.1 User application notes**

##### 6421 **E.4.5.2 Operations**

##### 6422 **E.4.5.2.1 Selection**

##### 6423 **E.4.5.2.2 Assignment**

#### 6424 **E.4.6 FCS\_RBG.5 Random bit generation**

##### 6425 **E.4.6.1 User application notes**

##### 6426 **E.4.6.2 Operations**

##### 6427 **E.4.6.2.1 Selection**

##### 6428 **E.4.6.2.2 Assignment**

#### 6429 **E.4.7 FCS\_RBG.6 Random bit generation service**

##### 6430 **E.4.7.1 User application notes**

##### 6431 **E.4.7.2 Operations**

##### 6432 **E.4.7.2.1 Selection**

##### 6433 **E.4.7.2.2 Assignment**

#### 6434 **E.5 Generation of random numbers (FCS\_RNG)**

6435 **Editors' note**

6436 | Editors are waiting for contribution from the CCDB Crypto Working Group

6437 **E.5.1 User notes**

6438 **E.5.2 FCS\_RNG.1 Random number generation**

6439 **E.5.2.1 User application notes**

6440 **E.5.2.2 Operations**

6441 **E.5.2.2.1 Selection**

6442 In FCS\_RNG.1 .1 the PP/ST author **should**

6443 **E.5.2.2.2 Assignment**

6444 In FCS\_RNG.1 .1 the PP/ST author **should**



## Annex F (normative)

### Class FDP: User data protection- application notes

#### F.1 General information

This class contains families specifying requirements related to protecting user data. This class differs from FIA and FPT in that FDP: User data protection specifies components to protect user data, FIA specifies components to protect attributes associated with the user, and FPT specifies components to protect TSF information.

The class does not contain explicit requirements for traditional Mandatory Access Controls (MAC) or traditional Discretionary Access Controls (DAC); however, such requirements **may** be constructed using components from this class.

FDP: User data protection does not explicitly deal with confidentiality, integrity, or availability, as all three are most often intertwined in the policy and mechanisms. However, the TOE security policy must adequately cover these three objectives in the PP/ST.

A final aspect of this class is that it specifies access control in terms of “operations”. An operation is defined as a specific type of access on a specific object. It depends on the level of abstraction of the PP/ST author whether these operations are described as “read” and/or “write” operations, or as more complex operations such as “update the database”.

The access control policies are policies that control access to the information container. The attributes represent attributes of the container. Once the information is out of the container, the accessor is free to modify that information, including writing the information into a different container with different attributes. By contrast, an information flow policies controls access to the information, independent of the container. The attributes of the information, which **may** be associated with the attributes of the container (or **may** not, as in the case of a multi-level database) stay with the information as it moves. The accessor does not have the ability, in the absence of an explicit authorization, to change the attributes of the information.

This class is not meant to be a complete taxonomy of IT access policies, as others **can** be imagined. Those policies included here are simply those for which current experience with actual systems provides a basis for specifying requirements. There **may** be other forms of intent that are not captured in the definitions here.

#### EXAMPLE

For example, a goal of having user-imposed (and user-defined) controls on information flow (such as an automated implementation of the NO FOREIGN handling caveat).

Such concepts **could** be handled as refinements of, or extensions to the FDP: User data protection components.

Finally, it is important when looking at the components in FDP: User data protection to remember that these components are requirements for functions that **may** be implemented by a mechanism that also serves or **could** serve another purpose.

#### EXAMPLE

it is possible to build an access control policy (Access control policy (FDP\_ACC)) that uses labels (FDP\_1FF.1 Simple security attributes) as the basis of the access control mechanism.

A set of SFRs **may** encompass many security function policies (SFPs), each to be identified by the two policy-oriented components Access control policy (FDP\_ACC), and Information flow control policy (FDP\_IFC). These policies will typically take confidentiality, integrity, and availability aspects into consideration as required, to satisfy the TOE requirements. Care **should**

6485 be taken to ensure that all objects are covered by at least one SFP and that there are no conflicts  
6486 arising from implementing the multiple SFPs.

6487 When building a PP/ST using components from the FDP: User data protection class, the  
6488 following information provides guidance on where to look and what to select from the class.

6489 The requirements in the FDP: User data protection class are defined in terms of a set of SFRs  
6490 that will implement a SFP. Since a TOE **may** implement multiple SFPs simultaneously, the PP/ST  
6491 author must specify the name for each SFP, so it **can** be referenced in other families. This name  
6492 will then be used in each component selected to indicate that it is being used as part of the  
6493 definition of requirements for that SFP. This allows the author to easily indicate the scope for  
6494 operations such as objects covered, operations covered, authorized users, etc.

6495 Each instantiation of a component **can** apply to only one SFP. Therefore, if an SFP is specified in  
6496 a component then this SFP will apply to all the elements in this component. The components  
6497 **may** be instantiated multiple times within a PP/ST to account for different policies if so desired.

6498 The key to selecting components from this family is to have a well-defined set of TOE security  
6499 objectives to enable proper selection of the components from the two policy components;  
6500 Access control policy (FDP\_ACC) and Information flow control policy (FDP\_IFC). In Access  
6501 control policy (FDP\_ACC) and Information flow control policy (FDP\_IFC) respectively, all access  
6502 control policies and all information flow control policies are named. Furthermore, the scope of  
6503 control of these components in terms of the subjects, objects and operations covered by this  
6504 security functionality. The names of these policies are meant to be used throughout the  
6505 remainder of the functional components that have an operation that calls for an assignment or  
6506 selection of an “access control SFP” or an “information flow control SFP”. The rules that define  
6507 the functionality of the named access control and information flow control SFPs will be defined  
6508 in the Access control functions (FDP\_ACF) and Information flow control functions (FDP\_IfF)  
6509 families (respectively).

6510 The following steps are guidance on how this class is applied in the construction of a PP/ST:

- 6511 a) Identify the policies to be enforced from the Access control policy (FDP\_ACC), and  
6512 Information flow control policy (FDP\_IFC) families. These families define scope of  
6513 control for the policy, granularity of control and **may** identify some rules to go with  
6514 the policy.
- 6515 b) Identify the components and perform any applicable operations in the policy  
6516 components. The assignment operations **may** be performed generally (such as with  
6517 a statement “All files”) or specifically (“The files “A”, “B”, etc.) depending upon the  
6518 level of detail known.
- 6519 c) Identify any applicable function components from the Access control functions  
6520 (FDP\_ACF) and Information flow control functions (FDP\_IfF) families to address  
6521 the named policy families from Access control policy (FDP\_ACC) and Information  
6522 flow control policy (FDP\_IFC). Perform the operations to make the components  
6523 define the rules to be enforced by the named policies. This **should** make the  
6524 components fit the requirements of the selected function envisioned or to be built.
- 6525 d) Identify who will have the ability to control and change security attributes under  
6526 the function, such as only a security administrator, only the owner of the object, etc.  
6527 Select the appropriate components from FMT: Security management and perform  
6528 the operations. Refinements **may** be useful here to identify missing features, such  
6529 as that some or all changes must be done via trusted path.
- 6530 e) Identify any appropriate components from the FMT: Security management for  
6531 initial values for new objects and subjects.
- 6532 f) Identify any applicable rollback components from the Rollback (FDP\_ROL) family.
- 6533 g) Identify any applicable residual information protection requirements from the  
6534 Residual information protection (FDP\_RIP) family.

- 6535 h) Identify any applicable import or export components, and how security attributes  
6536 **should** be handled during import and export, from the Import from outside of the  
6537 TOE (FDP\_ITC) and Export from the TOE (FDP\_ETC) families.
- 6538 i) Identify any applicable internal TOE communication components from the Internal  
6539 TOE transfer (FDP\_ITT) family.
- 6540 j) Identify any requirements for integrity protection of stored information from the  
6541 Stored data integrity (FDP\_SDI).
- 6542 k) Identify any applicable inter-TSF communication components from the Inter-TSF  
6543 user data confidentiality transfer protection (FDP\_UCT) or Inter-TSF user data  
6544 integrity transfer protection (FDP\_UIT) families.

## 6545 **F.2 Access control policy (FDP\_ACC)**

### 6546 **F.2.1 User notes**

6547 This family is based upon the concept of arbitrary controls on the interaction of subjects and  
6548 objects. The scope and purpose of the controls is based upon the attributes of the accessor  
6549 (subject), the attributes of the container being accessed (object), the actions (operations) and  
6550 any associated access control rules.

6551 The components in this family are capable of identifying the access control SFPs (by name) to  
6552 be enforced by the traditional Discretionary Access Control (DAC) mechanisms. It further  
6553 defines the subjects, objects and operations that are covered by identified access control SFPs.  
6554 The rules that define the functionality of an access control SFP will be defined by other families,  
6555 such as Access control functions (FDP\_ACF) and Export from the TOE (FDP\_ETC). The names of  
6556 the access control SFPs defined in Access control policy (FDP\_ACC) are meant to be used  
6557 throughout the remainder of the functional components that have an operation that calls for an  
6558 assignment or selection of an “access control SFP.”

6559 The access control SFP covers a set of triplets: subject, object, and operations. Therefore, a  
6560 subject **can** be covered by multiple access control SFPs but only with respect to a different  
6561 operation or a different object. Of course, the same applies to objects and operations.

6562 A critical aspect of an access control function that enforces an access control SFP is the ability  
6563 for users to modify the attributes involved in access control decisions. The Access control policy  
6564 (FDP\_ACC) family does not address these aspects. Some of these requirements are left  
6565 undefined, but **can** be added as refinements, while others are covered elsewhere in other  
6566 families and classes such as FMT: Security management.

6567 There are no audit requirements in Access control policy (FDP\_ACC) as this family specifies  
6568 access control SFP requirements. Audit requirements will be found in families specifying  
6569 functions to satisfy the access control SFPs identified in this family.

6570 This family provides a PP/ST author the capability to specify several policies, for example, a  
6571 fixed access control SFP to be applied to one scope of control, and a flexible access control SFP  
6572 to be defined for a different scope of control. To specify more than one access control policy, the  
6573 components from this family **can** be iterated multiple times in a PP/ST to different subsets of  
6574 operations and objects. This will accommodate TOEs that contain multiple policies, each  
6575 addressing a particular set of operations and objects. In other words, the PP/ST author **should**  
6576 specify the required information in the ACC component for each of the access control SFPs that  
6577 the TSF will enforce. For example, a TOE incorporating three access control SFPs, each covering  
6578 only a subset of the objects, subjects, and operations within the TOE, will contain one  
6579 FDP\_ACC.1 Subset access control component for each of the three access-control SFPs,  
6580 necessitating a total of three FDP\_ACC.1 Subset access control components.

### 6581 **F.2.2 FDP\_ACC.1 Subset access control**

#### 6582 **F.2.2.1 User application notes**

6583 The terms object and subject refer to generic elements in the TOE. For a policy to be  
 6584 implementable, the entities must be clearly identified. For a PP, the objects and operations  
 6585 might be expressed as types such as: named objects, data repositories, observe accesses, etc.  
 6586 For a specific TOE these generic terms (subject, object) must be refined.

EXAMPLE

files, registers, ports, daemons, open calls, etc.

6587 This component specifies that the policy cover some well-defined set of operations on some  
 6588 subset of the objects. It places no constraints on any operations outside the set - including  
 6589 operations on objects for which other operations are controlled.

## 6590 **F.2.2.2 Operations**

### 6591 **F.2.2.2.1 Assignment**

6592 In FDP\_ACC.1.1, the PP/ST author **should** specify a uniquely named access control SFP to be  
 6593 enforced by the TSF.

6594 In FDP\_ACC.1.1, the PP/ST author **should** specify the list of subjects, objects, and operations  
 6595 among subjects and objects covered by the SFP.

## 6596 **F.2.3 FDP\_ACC.2 Complete access control**

### 6597 **F.2.3.1 User application notes**

6598 This component requires that all possible operations on objects, that are included in the SFP,  
 6599 are covered by an access control SFP.

6600 The PP/ST author must demonstrate that each combination of objects and subjects is covered  
 6601 by an access control SFP.

## 6602 **F.2.3.2 Operations**

### 6603 **F.2.3.2.1 Assignment**

6604 In FDP\_ACC.2.1, the PP/ST author **should** specify a uniquely named access control SFP to be  
 6605 enforced by the TSF.

6606 In FDP\_ACC.2.1, the PP/ST author **should** specify the list of subjects and objects covered by the  
 6607 SFP. All operations among those subjects and objects will be covered by the SFP.

## 6608 **F.3 Access control functions (FDP\_ACF)**

### 6609 **F.3.1 User notes**

6610 This family describes the rules for the specific functions that **can** implement an access control  
 6611 policy named in Access control policy (FDP\_ACC) which also specifies the scope of control of the  
 6612 policy.

6613 This family provides a PP/ST author the capability to describe the rules for access control. This  
 6614 results in a TOE where the access to objects will not change. An example of such an object is  
 6615 "Message of the Day", which is readable by all, and changeable only by the authorized  
 6616 administrator. This family also provides the PP/ST author with the ability to describe rules that  
 6617 provide for exceptions to the general access control rules. Such exceptions would either  
 6618 explicitly allow or deny authorization to access an object.

6619 There are no explicit components to specify other possible functions such as two-person  
 6620 control, sequence rules for operations, or exclusion controls. However, these mechanisms, as  
 6621 well as traditional DAC mechanisms, **can** be represented with the existing components, by  
 6622 careful drafting of the access control rules.

6623 A variety of acceptable access control functionality **may** be specified in this family.

## EXAMPLE

- Access control lists (ACLs)
- Time-based access control specifications
- Origin-based access control specifications
- Owner-controlled access control attributes

6624 **F.3.2 FDP\_ACF.1 Security attribute based access control**6625 **F.3.2.1 User application notes**

6626 This component provides requirements for a mechanism that mediates access control based on  
 6627 security attributes associated with subjects and objects. Each object and subject has a set of  
 6628 associated attributes, such as location, time of creation, access rights such as Access Control  
 6629 Lists (ACLs)). This component allows the PP/ST author to specify the attributes that will be  
 6630 used for the access control mediation. This component allows access control rules, using these  
 6631 attributes, to be specified.

## EXAMPLE

Examples of the attributes that a PP/ST author might assign are:

An identity attribute may be associated with users, subjects, or objects to be used for mediation. Examples of such attributes might be the name of the program image used in the creation of the subject, or a security attribute assigned to the program image.

A time attribute can be used to specify that access will be authorized during certain times of the day, during certain days of the week, or during a certain calendar year.

A location attribute **could** specify whether the location is the location of the request for the operation, the location where the operation will be carried out, or both. It **could** be based upon internal tables to translate the logical interfaces of the TSF into locations such as through terminal locations, CPU locations, etc.

A grouping attribute allows a single group of users to be associated with an operation for the purposes of access control. If required, the refinement operation should be used to specify the maximum number of definable groups, the maximum membership of a group, and the maximum number of groups to which a user can concurrently be associated.

6632 This component also provides requirements for the access control security functions to be able  
 6633 to explicitly authorize or deny access to an object based upon security attributes. This **could** be  
 6634 used to provide privilege, access rights, or access authorizations within the TOE. Such  
 6635 privileges, rights, or authorizations **could** apply to users, subjects (representing users or  
 6636 applications), and objects.

6637 **F.3.2.2 Operations**6638 **F.3.2.2.1 Assignment**

6639 In FDP\_ACF.1.1, the PP/ST author **should** specify an access control SFP name that the TSF is to  
 6640 enforce. The name of the access control SFP, and the scope of control for that policy are defined  
 6641 in components from Access control policy (FDP\_ACC).

6642 In FDP\_ACF.1.1, the PP/ST author **should** specify, for each controlled subject and object, the  
 6643 security attributes and/or named groups of security attributes that the function will use in the  
 6644 specification of the rules. For example, such attributes **may** be things such as the user identity,  
 6645 subject identity, role, time of day, location, ACLs, or any other attribute specified by the PP/ST  
 6646 author. Named groups of security attributes **can** be specified to provide a convenient means to  
 6647 refer to multiple security attributes. Named groups **could** provide a useful way to associate  
 6648 "roles" defined in Security management roles (FMT\_SMR), and all of their relevant attributes,  
 6649 with subjects. In other words, each role **could** relate to a named group of attributes.

6650 In FDP\_ACF.1.2, the PP/ST author **should** specify the SFP rules governing access among  
 6651 controlled subjects and controlled objects using controlled operations on controlled objects.  
 6652 These rules specify when access is granted or denied. It **can** specify general access control  
 6653 functions or granular access control functions.



**EXAMPLE**

General access control functions: typical permission bits

Granular access control: Access Control Lists (ACL)

6654 In FDP\_ACF.1.3, the PP/ST author **should** specify the rules, based on security attributes, that  
 6655 explicitly authorize access of subjects to objects that will be used to explicitly authorize access.  
 6656 These rules are in addition to those specified in FDP\_ACF.1.1. They are included in FDP\_ACF.1.3  
 6657 as they are intended to contain exceptions to the rules in FDP\_ACF.1.1. An example of rules to  
 6658 explicitly authorize access is based on a privilege vector associated with a subject that always  
 6659 grants access to objects covered by the access control SFP that has been specified. If such a  
 6660 capability is not desired, then the PP/ST author **should** specify “none”.

6661 In FDP\_ACF.1.4, the PP/ST author **should** specify the rules, based on security attributes, that  
 6662 explicitly deny access of subjects to objects. These rules are in addition to those specified in  
 6663 FDP\_ACF.1.1 . They are included in FDP\_ACF.1.4 as they are intended to contain exceptions to  
 6664 the rules in FDP\_ACF.1.1 . An example of rules to explicitly deny access is based on a privilege  
 6665 vector associated with a subject that always denies access to objects covered by the access  
 6666 control SFP that has been specified. If such a capability is not desired, then the PP/ST author  
 6667 **should** specify “none”.

## 6668 **F.4Data authentication (FDP\_DAU)**

### 6669 **F.4.1 User notes**

6670 This family describes specific functions that **can** be used to authenticate “static” data.

6671 Components in this family are to be used when there is a requirement for “static” data  
 6672 authentication, i.e. where data is to be signed but not transmitted.

6673 Note the Non-repudiation of origin (FCO\_NRO) family provides for non-repudiation of origin of information  
 6674 received during a data exchange.

### 6675 **F.4.2 FDP\_DAU.1 Basic Data Authentication**

#### 6676 **F.4.2.1 User application notes**

6677 This component **may** be satisfied by one-way hash functions to generate a hash value for a  
 6678 definitive document that **may** be used as verification of the validity or authenticity of its  
 6679 information content.

**EXAMPLE**

cryptographic checksum, fingerprint, message digest

#### 6680 **F.4.2.2 Operations**

##### 6681 **F.4.2.2.1 Assignment**

6682 In FDP\_DAU.1.1, the PP/ST author **should** specify the list of objects or information types for  
 6683 which the TSF **shall** be capable of generating data authentication evidence.

6684 In FDP\_DAU.1.2, the PP/ST author **should** specify the list of subjects that will have the ability to  
 6685 verify data authentication evidence for the objects identified in the previous element. The list of  
 6686 subjects **could** be very specific, if the subjects are known, or it **could** be more generic and refer  
 6687 to a “type” of subject such as an identified role.

### 6688 **F.4.3 FDP\_DAU.2 Data Authentication with Identity of Guarantor**

#### 6689 **F.4.3.1 User application notes**

6690 This component additionally requires the ability to verify the identity of the user that provided  
 6691 the guarantee of authenticity

<p>EXAMPLE</p> <p>a trusted third party.</p>
--

## 6692 F.4.3.2 Operations

### 6693 F.4.3.2.1 Assignment

6694 In FDP\_DAU.2.1, the PP/ST author **should** specify the list of objects or information types for  
 6695 which the TSF **shall** be capable of generating data authentication evidence.

6696 In FDP\_DAU.2.2, the PP/ST author **should** specify the list of subjects that will have the ability to  
 6697 verify data authentication evidence for the objects identified in the previous element as well as  
 6698 the identity of the user that created the data authentication evidence.

## 6699 F.5 Export from the TOE (FDP\_ETC)

### 6700 F.5.1 User notes

6701 This family defines functions for TSF-mediated exporting of user data from the TOE such that its  
 6702 security attributes either **can** be explicitly preserved or **can** be ignored once it has been  
 6703 exported. Consistency of these security attributes are addressed by Inter-TSF TSF data  
 6704 consistency (FPT\_TDC).

6705 Export from the TOE (FDP\_ETC) is concerned with limitations on export and association of  
 6706 security attributes with the exported user data.

6707 This family, and the corresponding Import family Import from outside of the TOE (FDP\_ITC),  
 6708 address how the TOE deals with user data transferred into and outside its control. In principle,  
 6709 this family is concerned with the TSF-mediated exporting of user data and its related security  
 6710 attributes.

6711 A variety of activities might be involved here:

- 6712 a) exporting of user data without any security attributes;
- 6713 b) exporting user data including security attributes where the two are associated with
- 6714 one another and the security attributes unambiguously represent the exported
- 6715 user data.

6716 If there are multiple SFPs (access control and/or information flow control) then it **may** be  
 6717 appropriate to iterate these components once for each named SFP.

### 6718 F.5.2 FDP\_ETC.1 Export of user data without security attributes

#### 6719 F.5.2.1 User application notes

6720 This component is used to specify the TSF-mediated exporting of user data without the export  
 6721 of its security attributes.

#### 6722 F.5.2.2 Operations

##### 6723 F.5.2.2.1 Assignment

6724 In FDP\_ETC.1.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
 6725 flow control SFP(s) that will be enforced when exporting user data. The user data that this  
 6726 function exports is scoped by the assignment of these SFPs.

### 6727 F.5.3 FDP\_ETC.2 Export of user data with security attributes

#### 6728 F.5.3.1 User application notes

6729 The user data is exported together with its security attributes. The security attributes are  
 6730 unambiguously associated with the user data. There are several ways of achieving this  
 6731 association. One way that this **can** be achieved is by physically collocating the user data and the  
 6732 security attributes.

## EXAMPLE

On the same external media

or by using cryptographic techniques such as secure signatures to associate the attributes and the user data. Inter-TSF trusted channel (FTP\_ITC) **could** be used to assure that the attributes are correctly received at the other trusted IT product while Inter-TSF TSF data consistency (FPT\_TDC) **can** be used to make sure that those attributes are properly interpreted. Furthermore, Trusted path (FTP\_TRP) **could** be used to make sure that the export is being initiated by the proper user.

### F.5.3.2 Operations

#### F.5.3.2.1 Assignment

In FDP\_ETC.2.1, the PP/ST author **should** specify the access control SFP(s) and/or information flow control SFP(s) that will be enforced when exporting user data. The user data that this function exports is scoped by the assignment of these SFPs.

In FDP\_ETC.2.4, the PP/ST author **should** specify any additional exportation control rules or “none” if there are no additional exportation control rules. These rules will be enforced by the TSF in addition to the access control SFPs and/or information flow control SFPs selected in FDP\_ETC.2.1.

## F.6 Information flow control policy (FDP\_IFC)

### F.6.1 User notes

This family covers the identification of information flow control SFPs; and, for each, specifies the scope of control of the SFP.

The components in this family are capable of identifying the information flow control SFPs to be enforced by the traditional Mandatory Access Control mechanisms that would be found in a TOE. However, they go beyond just the traditional MAC mechanisms and **can** be used to identify and describe non-interference policies and state-transitions. It further defines the subjects under control of the policy, the information under control of the policy, and operations which cause controlled information to flow to and from controlled subjects for each information flow control SFP in the TOE. The information flow control SFP will be defined by other families such as Information flow control functions (FDP\_IFF) and Export from the TOE (FDP\_ETC). The information flow control SFPs named here in Information flow control policy (FDP\_IFC) are meant to be used throughout the remainder of the functional components that have an operation that calls for an assignment or selection of an “information flow control SFP.”

These components are quite flexible. They allow the domain of flow control to be specified and there is no requirement that the mechanism be based upon labels. The different elements of the information flow control components also permit different degrees of exception to the policy.

Each SFP covers a set of triplets: subject, information, and operations that cause information to flow to and from subjects. Some information flow control policies **may** be at a very low level of detail and explicitly describe subjects in terms of processes within an operating system. Other information flow control policies **may** be at a high level and describe subjects in the generic sense of users or input/output channels. If the information flow control policy is at too high a level of detail, it **may** not clearly define the desired IT security functions. In such cases, it is more appropriate to include such descriptions of information flow control policies as objectives. Then the desired IT security functions **can** be specified as supportive of those objectives.

In the second component (FDP\_IFC.2 Complete information flow control), each information flow control SFP will cover all possible operations that cause information covered by that SFP to flow to and from subjects covered by that SFP. Furthermore, all information flows will need to be covered by a SFP. Therefore, for each action that causes information to flow, there will be a set of rules that define whether the action is allowed. If there are multiple SFPs that are applicable



for a given information flow, all involved SFPs must allow this flow before it is permitted to take place.

An information flow control SFP covers a well-defined set of operations. The SFPs coverage **may** be “complete” with respect to some information flows, or it **may** address only some of the operations that affect the information flow.

An access control SFP controls access to the objects that contain information. An information flow control SFP controls access to the information, independent of its container. The attributes of the information, which **may** be associated with the attributes of the container (or **may** not, as in the case of a multi-level database) stay with the information as it flows. The accessor does not have the ability, in the absence of an explicit authorization, to change the attributes of the information.

Information flows and operations **can** be expressed at multiple levels. In the case of a ST, the information flows and operations might be specified at a system-specific level: TCP/IP packets flowing through a firewall based upon known IP addresses. For a PP, the information flows and operations might be expressed as types: email, data repositories, observe accesses, etc.

The components in this family **can** be applied multiple times in a PP/ST to different subsets of operations and objects. This will accommodate TOEs that contain multiple policies, each addressing a particular set of objects, subjects, and operations.

## **F.6.2 FDP\_IFC.1 Subset information flow control**

### **F.6.2.1 User application notes**

This component requires that an information flow control policy apply to a subset of the possible operations in the TOE.

### **F.6.2.2 Operations**

#### **F.6.2.2.1 Assignment**

In FDP\_IFC.1.1, the PP/ST author **should** specify a uniquely named information flow control SFP to be enforced by the TSF.

In FDP\_IFC.1.1, the PP/ST author **should** specify the list of subjects, information, and operations which cause controlled information to flow to and from controlled subjects covered by the SFP. As mentioned above, the list of subjects **could** be at various levels of detail depending on the needs of the PP/ST author.

#### **EXAMPLE**

It **could** specify users, machines, or processes.

Information **could** refer to data such as email or network protocols, or more specific objects similar to those specified under an access control policy. If the information that is specified is contained within an object that is subject to an access control policy, then both the access control policy and information flow control policy must be enforced before the specified information **could** flow to or from the object.

## **F.6.3 FDP\_IFC.2 Complete information flow control**

### **F.6.3.1 User application notes**

This component requires that all possible operations that cause information to flow to and from subjects included in the SFP, are covered by an information flow control SFP.

The PP/ST author must demonstrate that each combination of information flows and subjects is covered by an information flow control SFP.

### **F.6.3.2 Operations**

#### **F.6.3.2.1 Assignment**

6822 In FDP\_IFC.2.1, the PP/ST author **should** specify a uniquely named information flow control SFP  
6823 to be enforced by the TSF.

6824 In FDP\_IFC.2.1, the PP/ST author **should** specify the list of subjects and information that will be  
6825 covered by the SFP. All operations that cause that information to flow to and from subjects will  
6826 be covered by the SFP. As mentioned above, the list of subjects **could** be at various levels of  
6827 detail depending on the needs of the PP/ST author.

EXAMPLE

It **could** specify users, machines, or processes.

6828 Information **could** refer to data such as email or network protocols, or more specific objects  
6829 similar to those specified under an access control policy. If the information that is specified is  
6830 contained within an object that is subject to an access control policy, then both the access  
6831 control policy and information flow control policy must be enforced before the specified  
6832 information **could** flow to or from the object.

## 6833 **F.7 Information flow control functions (FDP\_IFF)**

### 6834 **F.7.1 User notes**

6835 This family describes the rules for the specific functions that **can** implement the information  
6836 flow control SFPs named in Information flow control policy (FDP\_IFC), which also specifies the  
6837 scope of control of the policies. It consists of two “trees:” one addressing the common  
6838 information flow control function issues, and a second addressing illicit information flows (i.e.  
6839 covert channels) with respect to one or more information flow control SFPs. This division arises  
6840 because the issues concerning illicit information flows are, in some sense, orthogonal to the rest  
6841 of an SFP. Illicit information flows are flows in violation of policy; thus, they are not a policy  
6842 issue.

6843 In order to implement strong protection against disclosure or modification in the face of  
6844 untrusted software, controls on information flow are required. Access controls alone are not  
6845 sufficient because they only control access to containers, allowing the information they contain  
6846 to flow, without controls, throughout a system.

6847 In this family, the phrase “types of illicit information flows” is used. This phrase **may** be used to  
6848 refer to the categorization of flows as “Storage Channels” or “Timing Channels”, or it **can** refer to  
6849 improved categorizations reflective of the needs of a PP/ST author.

6850 The flexibility of these components allows the definition of a privilege policy within FDP\_IFF.1  
6851 Simple security attributes and FDP\_IFF.2 Hierarchical security attributes to allow the controlled  
6852 bypass of all or part of a particular SFP. If there is a need for a predefined approach to SFP  
6853 bypass, the PP/ST author **should** consider incorporating a privilege policy.

### 6854 **F.7.2 FDP\_IFF.1 Simple security attributes**

#### 6855 **F.7.2.1 User application notes**

6856 This component requires security attributes on information, and on subjects that cause that  
6857 information to flow and subjects that act as recipients of that information. The attributes of the  
6858 containers of the information **should** also be considered if it is desired that they **should** play a  
6859 part in information flow control decisions or if they are covered by an access control policy.  
6860 This component specifies the key rules that are enforced and describes how security attributes  
6861 are derived.

6862 This component does not specify the details of how a security attribute is assigned (i.e. user  
6863 versus process). Flexibility in policy is provided by having assignments that allow specification  
6864 of additional policy and function requirements, as necessary.

6865 This component also provides requirements for the information flow control functions to be  
6866 able to explicitly authorize and deny an information flow based upon security attributes. This

6867 **could** be used to implement a privilege policy that covers exceptions to the basic policy defined  
6868 in this component.

## 6869 **F.7.2.2 Operations**

### 6870 **F.7.2.2.1 Assignment**

6871 In FDP\_IFF.1.1, the PP/ST author **should** specify the information flow control SFPs enforced by  
6872 the TSF. The name of the information flow control SFP, and the scope of control for that policy  
6873 are defined in components from Information flow control policy (FDP\_IFC).

6874 In FDP\_IFF.1.1, the PP/ST author **should** specify, for each type of controlled subject and  
6875 information, the security attributes that are relevant to the specification of the SFP rules.

#### EXAMPLE

For example, such security attributes **may** be things such the subject identifier, subject sensitivity label, subject clearance label, information sensitivity label, etc.

6876 The types of security attributes **should** be sufficient to support the environmental needs.

6877 In FDP\_IFF.1.2, the PP/ST author **should** specify for each operation, the security attribute-based  
6878 relationship that must hold between subject and information security attributes that the TSF  
6879 will enforce.

6880 In FDP\_IFF.1.3, the PP/ST author **should** specify any additional information flow control SFP  
6881 rules that the TSF is to enforce. This includes all rules of the SFP that are either not based on the  
6882 security attributes of the information and the subject or rules that automatically modify the  
6883 security attributes of information or subjects as a result of an access operation. An example for  
6884 the first case is a rule of the SFP controlling a threshold value for specific types of information.  
6885 This would for example be the case when the information flow SFP contains rules on access to  
6886 statistical data where a subject is only allowed to access this type of information up to a specific  
6887 number of accesses. An example for the second case would be a rule stating under which  
6888 conditions and how the security attributes of a subject or object change as the result of an  
6889 access operation. Some information flow policies for example **may** limit the number of access  
6890 operations to information with specific security attributes. If there are no additional rules then  
6891 the PP/ST author **should** specify “none”.

6892 In FDP\_IFF.1.4, the PP/ST author **should** specify the rules, based on security attributes, that  
6893 explicitly authorize information flows. These rules are in addition to those specified in the  
6894 preceding elements. They are included in FDP\_IFF.1.4 as they are intended to contain  
6895 exceptions to the rules in the preceding elements.

#### EXAMPLE

An example of rules to explicitly authorize information flows is based on a privilege vector associated with a subject that always grants the subject the ability to cause an information flow for information that is covered by the SFP that has been specified.

6896 If such a capability is not desired, then the PP/ST author **should** specify “none”.

6897 In FDP\_IFF.1.5, the PP/ST author **should** specify the rules, based on security attributes, that  
6898 explicitly deny information flows. These rules are in addition to those specified in the preceding  
6899 elements. They are included in FDP\_IFF.1.5 as they are intended to contain exceptions to the  
6900 rules in the preceding elements. An example of rules to explicitly deny information flows is  
6901 based on a privilege vector associated with a subject that always denies the subject the ability  
6902 to cause an information flow for information that is covered by the SFP that has been specified.  
6903 If such a capability is not desired, then the PP/ST author **should** specify “none”.

## 6904 **F.7.3 FDP\_IFF.2 Hierarchical security attributes**

### 6905 **F.7.3.1 User application notes**

6906 This component requires that the named information flow control SFP uses hierarchical  
6907 security attributes that form a lattice.

6908 It is important to note that the hierarchical relationship requirements identified in FDP\_IFF.2.4  
6909 need only apply to the information flow control security attributes for the information flow  
6910 control SFPs that have been identified in FDP\_IFF.2.1. This component is not meant to apply to  
6911 other SFPs such as access control SFPs.

6912 FDP\_IFF.2.6 phrases the requirements for the set of security attributes to form a lattice. A  
6913 number of information flow policies defined in the literature and implemented in IT products  
6914 are based on a set of security attributes that form a lattice. FDP\_IFF.2.6 is specifically included  
6915 to address this type of information flow policies.

6916 If it is the case that multiple information flow control SFPs are to be specified, and that each of  
6917 these SFPs will have their own security attributes that are not related to one another, then the  
6918 PP/ST author **should** iterate this component once for each of those SFPs. Otherwise a conflict  
6919 might arise with the sub-items of FDP\_IFF.2.4 since the required relationships will not exist.

### 6920 **F.7.3.2 Operations**

#### 6921 **F.7.3.2.1 Assignment**

6922 In FDP\_IFF.2.1, the PP/ST author **should** specify the information flow control SFPs enforced by  
6923 the TSF. The name of the information flow control SFP, and the scope of control for that policy  
6924 are defined in components from Information flow control policy (FDP\_IFC).

6925 In FDP\_IFF.2.1, the PP/ST author **should** specify, for each type of controlled subject and  
6926 information, the security attributes that are relevant to the specification of the SFP rules. For  
6927 example, such security attributes **may** be things such the subject identifier, subject sensitivity  
6928 label, subject clearance label, information sensitivity label, etc. The types of security attributes  
6929 **should** be sufficient to support the environmental needs.

6930 In FDP\_IFF.2.2, the PP/ST author **should** specify for each operation, the security attribute-based  
6931 relationship that must hold between subject and information security attributes that the TSF  
6932 will enforce. These relationships **should** be based upon the ordering relationships between the  
6933 security attributes.

6934 In FDP\_IFF.2.3, the PP/ST author **should** specify any additional information flow control SFP  
6935 rules that the TSF is to enforce. This includes all rules of the SFP that are either not based on the  
6936 security attributes of the information and the subject or rules that automatically modify the  
6937 security attributes of information or subjects as a result of an access operation. An example for  
6938 the first case is a rule of the SFP controlling a threshold value for specific types of information.

#### EXAMPLE

This would for example be the case when the information flow SFP contains rules on access to statistical data where a subject is only allowed to access this type of information up to a specific number of accesses. An example for the second case would be a rule stating under which conditions and how the security attributes of a subject or object change as the result of an access operation.

6939 Some information flow policies **may** limit the number of access operations to information with  
6940 specific security attributes. If there are no additional rules then the PP/ST author **should** specify  
6941 “none”.

6942 In FDP\_IFF.2.4, the PP/ST author **should** specify the rules, based on security attributes, that  
6943 explicitly authorize information flows. These rules are in addition to those specified in the  
6944 preceding elements. They are included in FDP\_IFF.2.4 as they are intended to contain  
6945 exceptions to the rules in the preceding elements.

#### EXAMPLE

An example of rules to explicitly authorize information flows is based on a privilege vector associated with a subject that always grants the subject the ability to cause an information flow for information that is covered by the SFP that has been specified.

If such a capability is not desired, then the PP/ST author **should** specify “none”.

In FDP\_IFF.2.5, the PP/ST author **should** specify the rules, based on security attributes, that explicitly deny information flows. These rules are in addition to those specified in the preceding elements. They are included in FDP\_IFF.2.5 as they are intended to contain exceptions to the rules in the preceding elements. An example of rules to explicitly deny information flows is based on a privilege vector associated with a subject that always denies the subject the ability to cause an information flow for information that is covered by the SFP that has been specified. If such a capability is not desired, then the PP/ST author **should** specify “none”.

#### **F.7.4 FDP\_IFF.3 Limited illicit information flows**

##### **F.7.4.1 User application notes**

This component **should** be used when at least one of the SFPs that requires control of illicit information flows does not require elimination of flows.

For the specified illicit information flows, certain maximum capacities **should** be provided. In addition, a PP/ST author has the ability to specify whether the illicit information flows must be audited.

##### **F.7.4.2 Operations**

###### **F.7.4.2.1 Assignment**

In FDP\_IFF.3.1, the PP/ST author **should** specify the information flow control SFPs enforced by the TSF. The name of the information flow control SFP, and the scope of control for that policy are defined in components from Information flow control policy (FDP\_IFC).

In FDP\_IFF.3.1, the PP/ST author **should** specify the types of illicit information flows that are subject to a maximum capacity limitation.

In FDP\_IFF.3.1, the PP/ST author **should** specify the maximum capacity permitted for any identified illicit information flows.

#### **F.7.5 FDP\_IFF.4 Partial elimination of illicit information flows**

##### **F.7.5.1 User application notes**

This component **should** be used when all the SFPs that requires control of illicit information flows require elimination of some (but not necessarily all) illicit information flows.

##### **F.7.5.2 Operations**

###### **F.7.5.2.1 Assignment**

In FDP\_IFF.4.1, the PP/ST author **should** specify the information flow control SFPs enforced by the TSF. The name of the information flow control SFP, and the scope of control for that policy are defined in components from Information flow control policy (FDP\_IFC).

In FDP\_IFF.4.1, the PP/ST author **should** specify the types of illicit information flows which are subject to a maximum capacity limitation.

In FDP\_IFF.4.1, the PP/ST author **should** specify the maximum capacity permitted for any identified illicit information flows.

In FDP\_IFF.4.2, the PP/ST author **should** specify the types of illicit information flows to be eliminated. This list **may** not be empty as this component requires that some illicit information flows are to be eliminated.

#### **F.7.6 FDP\_IFF.5 No illicit information flows**

##### **F.7.6.1 User application notes**

This component **should** be used when the SFPs that require control of illicit information flows require elimination of all illicit information flows. However, the PP/ST author **should** carefully consider the potential impact that eliminating all illicit information flows might have on the



6991 normal functional operation of the TOE. Many practical applications have shown that there is an  
 6992 indirect relationship between illicit information flows and normal functionality within a TOE  
 6993 and eliminating all illicit information flows **may** result in less than desired functionality.

## 6994 **F.7.6.2 Operations**

### 6995 **F.7.6.2.1 Assignment**

6996 In FDP\_IFF.5.1, the PP/ST author **should** specify the information flow control SFP for which  
 6997 illicit information flows are to be eliminated. The name of the information flow control SFP, and  
 6998 the scope of control for that policy are defined in components from Information flow control  
 6999 policy (FDP\_IFC).

## 7000 **F.7.7 FDP\_IFF.6 Illicit information flow monitoring**

### 7001 **F.7.7.1 User application notes**

7002 This component **should** be used when it is desired that the TSF provide the ability to monitor  
 7003 the use of illicit information flows that exceed a specified capacity. If it is desired that such flows  
 7004 be audited, then this component **could** serve as the source of audit events to be used by  
 7005 components from the Security audit data generation (FAU\_GEN) family.

## 7006 **F.7.7.2 Operations**

### 7007 **F.7.7.2.1 Assignment**

7008 In FDP\_IFF.6.1, the PP/ST author **should** specify the information flow control SFPs enforced by  
 7009 the TSF. The name of the information flow control SFP, and the scope of control for that policy  
 7010 are defined in components from Information flow control policy (FDP\_IFC).

7011 In FDP\_IFF.6.1, the PP/ST author **should** specify the types of illicit information flows that will be  
 7012 monitored for exceeding a maximum capacity.

7013 In FDP\_IFF.6.1, the PP/ST author **should** specify the maximum capacity above which illicit  
 7014 information flows will be monitored by the TSF.

## 7015 **F.8 Information retention control (FDP\_IRC)**

### 7016 **F.8.1 User notes**

### 7017 **F.8.2 FDP\_IRC.1 Subset information control**

#### 7018 **F.8.2.1 User application notes**

#### 7019 **F.8.2.2 Operations**

##### 7020 **F.8.2.2.1 Assignment**

7021 In FDP\_IRC.1.1, the PP/ST author **should**

### 7022 **F.8.3 FDP\_IRC.2 Complete information control**

#### 7023 **F.8.3.1 User application notes**

#### 7024 **F.8.3.2 Operations**

##### 7025 **F.8.3.2.1 Assignment**

7026 In FDP\_IRC.2.1, the PP/ST author **should**

## 7027 **F.9 Import from outside of the TOE (FDP\_ITC)**

### 7028 **F.9.1 User notes**

7029 This family defines mechanisms for TSF-mediated importing of user data from outside the TOE  
 7030 into the TOE such that the user data security attributes **can** be preserved. Consistency of these  
 7031 security attributes are addressed by Inter-TSF TSF data consistency (FPT\_TDC).

7032 Import from outside of the TOE (FDP\_ITC) is concerned with limitations on import, user  
7033 specification of security attributes, and association of security attributes with the user data.

7034 This family, and the corresponding export family Export from the TOE (FDP\_ETC), address how  
7035 the TOE deals with user data outside its control. This family is concerned with assigning and  
7036 abstraction of the user data security attributes.

EXAMPLE

A variety of activities might be involved here:

- a) importing user data from an unformatted medium (such as., tape, scanner, video or audio signal), without including any security attributes, and physically marking the medium to indicate its contents;
- b) importing user data, including security attributes, from a medium and verifying that the object security attributes are appropriate;
- c) importing user data, including security attributes, from a medium using a cryptographic sealing technique to protect the association of user data and security attributes.

7037 This family is not concerned with the determination of whether the user data **may** be imported.  
7038 It is concerned with the values of the security attributes to associate with the imported user  
7039 data.

7040 There are two possibilities for the import of user data: either the user data is unambiguously  
7041 associated with reliable object security attributes (values and meaning of the security attributes  
7042 is not modified), or no reliable security attributes (or no security attributes at all) are available  
7043 from the import source. This family addresses both cases.

7044 If there are reliable security attributes available, they **may** have been associated with the user  
7045 data by physical means (the security attributes are on the same media), or by logical means (the  
7046 security attributes are distributed differently but include unique object identification).

EXAMPLE

cryptographic checksum

7047 This family is concerned with TSF-mediated importing of user data and maintaining the  
7048 association of security attributes as required by the SFP. Other families are concerned with  
7049 other import aspects such as consistency, trusted channels, and integrity that are beyond the  
7050 scope of this family. Furthermore, Import from outside of the TOE (FDP\_ITC) is only concerned  
7051 with the interface to the import medium. Export from the TOE (FDP\_ETC) is responsible for the  
7052 other end point of the medium (the source).

7053 Some of the well-known import requirements are:

- 7054 a) importing of user data without any security attributes;
- 7055 b) importing of user data including security attributes where the two are associated  
7056 with one another and the security attributes unambiguously represent the  
7057 information being imported.

7058 These import requirements **may** be handled by the TSF with or without human intervention,  
7059 depending on the IT limitations and the organizational security policy. For example, if user data  
7060 is received on a “confidential” channel, the security attributes of the objects will be set to  
7061 “confidential”.

7062 If there are multiple SFPs (access control and/or information flow control) then it **may** be  
7063 appropriate to iterate these components once for each named SFP.

## 7064 **F.9.2 FDP\_ITC.1 Import of user data without security attributes**

### 7065 **F.9.2.1 User application notes**

7066 This component is used to specify the import of user data that does not have reliable (or any)  
7067 security attributes associated with it. This function requires that the security attributes for the  
7068 imported user data be initialized within the TSF. It **could** also be the case that the PP/ST author

7069 specifies the rules for import. It **may** be appropriate, in some environments, to require that  
 7070 these attributes be supplied via a trusted path or a trusted channel mechanism.

## 7071 **F.9.2.2 Operations**

### 7072 **F.9.2.2.1 Assignment**

7073 In FDP\_ITC.1.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
 7074 flow control SFP(s) that will be enforced when importing user data from outside of the TOE.  
 7075 The user data that this function imports is scoped by the assignment of these SFPs.

7076 In FDP\_ITC.1.3, the PP/ST author **should** specify any additional importation control rules or  
 7077 “none” if there are no additional importation control rules. These rules will be enforced by the  
 7078 TSF in addition to the access control SFPs and/or information flow control SFPs selected in  
 7079 FDP\_ITC.1.1.

## 7080 **F.9.3 FDP\_ITC.2 Import of user data with security attributes**

### 7081 **F.9.3.1 User application notes**

7082 This component is used to specify the import of user data that has reliable security attributes  
 7083 associated with it. This function relies upon the security attributes that are accurately and  
 7084 unambiguously associated with the objects on the import medium. Once imported, those  
 7085 objects will have those same attributes. This requires Inter-TSF TSF data consistency  
 7086 (FPT\_TDC) to ensure the consistency of the data. It **could** also be the case that the PP/ST author  
 7087 specifies the rules for import.

## 7088 **F.9.3.2 Operations**

### 7089 **F.9.3.2.1 Assignment**

7090 In FDP\_ITC.2.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
 7091 flow control SFP(s) that will be enforced when importing user data from outside of the TOE.  
 7092 The user data that this function imports is scoped by the assignment of these SFPs.

7093 In FDP\_ITC.2.5, the PP/ST author **should** specify any additional importation control rules or  
 7094 “none” if there are no additional importation control rules. These rules will be enforced by the  
 7095 TSF in addition to the access control SFPs and/or information flow control SFPs selected in  
 7096 FDP\_ITC.2.1.

## 7097 **F.10 Internal TOE transfer (FDP\_ITT)**

### 7098 **F.10.1 User notes**

7099 This family provides requirements that address protection of user data when it is transferred  
 7100 between parts of a TOE across an internal channel. This **may** be contrasted with the Inter-TSF  
 7101 user data confidentiality transfer protection (FDP\_UCT) and Inter-TSF user data integrity  
 7102 transfer protection (FDP\_UIT) family, which provide protection for user data when it is  
 7103 transferred between distinct TSFs across an external channel, and Export from the TOE  
 7104 (FDP\_ETC) and Import from outside of the TOE (FDP\_ITC), which address TSF-mediated  
 7105 transfer of data to or from outside the TOE.

7106 The requirements in this family allow a PP/ST author to specify the desired security for user  
 7107 data while in transit within the TOE. This security **could** be protection against disclosure,  
 7108 modification, or loss of availability.

7109 The determination of the degree of physical separation above which this family **should** apply  
 7110 depends on the intended environment of use. In a hostile environment, there **may** be risks  
 7111 arising from transfers between parts of the TOE separated by only a system bus. In more benign  
 7112 environments, the transfers **may** be across more traditional network media.

7113 If there are multiple SFPs (access control and/or information flow control) then it **may** be  
 7114 appropriate to iterate these components once for each named SFP.



## 7115 **F.10.2 FDP\_ITT.1 Basic internal transfer protection**

### 7116 **F.10.2.1 Operations**

#### 7117 **F.10.2.1.1 Assignment**

7118 In FDP\_ITT.1.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
7119 flow control SFP(s) covering the information being transferred.

#### 7120 **F.10.2.1.2 Selection**

7121 In FDP\_ITT.1.1, the PP/ST author **should** specify the types of transmission errors that the TSF  
7122 **should** prevent occurring for user data while in transport. The options are disclosure,  
7123 modification, loss of use.

## 7124 **F.10.3 FDP\_ITT.2 Transmission separation by attribute**

### 7125 **F.10.3.1 User application notes**

7126 This component **could**, for example, be used to provide different forms of protection to  
7127 information with different clearance levels.

7128 One of the ways to achieve separation of data when it is transmitted is through the use of  
7129 separate logical or physical channels.

### 7130 **F.10.3.2 Operations**

#### 7131 **F.10.3.2.1 Assignment**

7132 In FDP\_ITT.2.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
7133 flow control SFP(s) covering the information being transferred.

#### 7134 **F.10.3.2.2 Selection**

7135 In FDP\_ITT.2.1, the PP/ST author **should** specify the types of transmission errors that the TSF  
7136 **should** prevent occurring for user data while in transport. The options are disclosure,  
7137 modification, loss of use.

#### 7138 **F.10.3.2.3 Assignment**

7139 In FDP\_ITT.2.2, the PP/ST author **should** specify the security attributes, the values of which the  
7140 TSF will use to determine when to separate data that is being transmitted between physically-  
7141 separated parts of the TOE. An example is that user data associated with the identity of one  
7142 owner is transmitted separately from the user data associated with the identity of a different  
7143 owner. In this case, the value of the identity of the owner of the data is what is used to  
7144 determine when to separate the data for transmission.

## 7145 **F.10.4 FDP\_ITT.3 Integrity monitoring**

### 7146 **F.10.4.1 User application notes**

7147 This component is used in combination with either FDP\_ITT.1 Basic internal transfer protection  
7148 or FDP\_ITT.2 Transmission separation by attribute. It ensures that the TSF checks received user  
7149 data (and their attributes) for integrity. FDP\_ITT.1 Basic internal transfer protection or  
7150 FDP\_ITT.2 Transmission separation by attribute will provide the data in a manner such that it is  
7151 protected from modification (so that FDP\_ITT.3 Integrity monitoring **can** detect any  
7152 modifications).

7153 The PP/ST author has to specify the types of errors that must be detected. The PP/ST author  
7154 **should** consider: modification of data, substitution of data, unrecoverable ordering change of  
7155 data, replay of data, incomplete data, in addition to other integrity errors.

7156 The PP/ST author must specify the actions that the TSF **should** take on detection of a failure.

## EXAMPLE

For example: ignore the user data, request the data again, inform the authorized administrator, reroute traffic for other lines.

7157 **F.10.4.2 Operations**7158 **F.10.4.2.1 Assignment**

7159 In FDP\_ITT.3.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
7160 flow control SFP(s) covering the information being transferred and monitored for integrity  
7161 errors.

7162 In FDP\_ITT.3.1, the PP/ST author **should** specify the type of possible integrity errors to be  
7163 monitored during transmission of the user data.

7164 In FDP\_ITT.3.2, the PP/ST author **should** specify the action to be taken by the TSF when an  
7165 integrity error is encountered.

## EXAMPLE

An example is that the TSF should request the resubmission of the user data. The SFP(s) specified in FDP\_ITT.3.1 will be enforced as the actions are taken by the TSF.

7166 **F.10.5 FDP\_ITT.4 Attribute-based integrity monitoring**7167 **F.10.5.1 User application notes**

7168 This component is used in combination with FDP\_ITT.2 Transmission separation by attribute. It  
7169 ensures that the TSF checks received user data, that has been transmitted by separate channels  
7170 (based on values of specified security attributes), for integrity. It allows the PP/ST author to  
7171 specify actions to be taken upon detection of an integrity error.

## EXAMPLE

This component **could** be used to provide different integrity error detection and action for information at different integrity levels.

7172 The PP/ST author has to specify the types of errors that must be detected. The PP/ST author  
7173 **should** consider: modification of data, substitution of data, unrecoverable ordering change of  
7174 data, replay of data, incomplete data, in addition to other integrity errors.

7175 The PP/ST author **should** specify the attributes (and associated transmission channels) that  
7176 necessitate integrity error monitoring.

7177 The PP/ST author must specify the actions that the TSF **should** take on detection of a failure.

## EXAMPLE

For example: ignore the user data, request the data again, inform the authorized administrator, reroute traffic for other lines.

7178 **F.10.5.2 Operations**7179 **F.10.5.2.1 Assignment**

7180 In FDP\_ITT.4.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
7181 flow control SFP(s) covering the information being transferred and monitored for integrity  
7182 errors.

7183 In FDP\_ITT.4.1, the PP/ST author **should** specify the type of possible integrity errors to be  
7184 monitored during transmission of the user data.

7185 In FDP\_ITT.4.1, the PP/ST author **should** specify a list of security attributes that require  
7186 separate transmission channels. This list is used to determine which user data to monitor for  
7187 integrity errors., based on its security attributes and its transmission channel. This element is  
7188 directly related to FDP\_ITT.2 Transmission separation by attribute.

7189 In FDP\_ITT.4.2, the PP/ST author **should** specify the action to be taken by the TSF when an  
 7190 integrity error is encountered. An example might be that the TSF **should** request the  
 7191 resubmission of the user data. The SFP(s) specified in FDP\_ITT.4.1 will be enforced as the  
 7192 actions are taken by the TSF.

## 7193 **F.11 Residual information protection (FDP\_RIP)**

### 7194 **F.11.1 User notes**

7195 Residual information protection ensures that TSF-controlled resources when de-allocated from  
 7196 an object and before they are reallocated to another object are treated by the TSF in a way that  
 7197 it is not possible to reconstruct all or part of the data contained in the resource before it was de-  
 7198 allocated.

7199 A TOE usually has a number of functions that potentially de-allocate resources from an object  
 7200 and potentially re-allocate those resources to objects. Some, but not all of those resources **may**  
 7201 have been used to store critical data from the previous use of the resource and for those  
 7202 resources FDP\_RIP requires that they are prepared for reuse. Object reuse applies to explicit  
 7203 requests of a subject or user to release resources as well as implicit actions of the TSF that  
 7204 result in the de-allocation and subsequent re-allocation of resources to different objects.

#### EXAMPLE

Examples of explicit requests are the deletion or truncation of a file or the release of an area of main memory.  
 Examples of implicit actions of the TSF are the de-allocation and re-allocation of cache regions.

7205 The requirement for object reuse is related to the content of the resource belonging to an  
 7206 object, not all information about the resource or object that **may** be stored elsewhere in the TSF.  
 7207 As an example, to satisfy the FDP\_RIP requirement for files as objects requires that all sectors  
 7208 that make up the file need to be prepared for re-use.

7209 It also applies to resources that are serially reused by different subjects within the system. For  
 7210 example, most operating systems typically rely upon hardware registers (resources) to support  
 7211 processes within the system. As processes are swapped from a “run” state to a “sleep” state  
 7212 (and vice versa), these registers are serially reused by different subjects. While this “swapping”  
 7213 action **may** not be considered an allocation or deallocation of a resource, Residual information  
 7214 protection (FDP\_RIP) **could** apply to such events and resources.

7215 Residual information protection (FDP\_RIP) typically controls access to information that is not  
 7216 part of any currently defined or accessible object; however, in certain cases this **may** not be  
 7217 true. For example, object “A” is a file and object “B” is the disk upon which that file resides. If  
 7218 object “A” is deleted, the information from object “A” is under the control of Residual  
 7219 information protection (FDP\_RIP) even though it is still part of object “B”.

7220 It is important to note that Residual information protection (FDP\_RIP) applies only to on-line  
 7221 objects and not off-line objects such as those backed-up on tapes. For example, if a file is deleted  
 7222 in the TOE, Residual information protection (FDP\_RIP) **can** be instantiated to require that no  
 7223 residual information exists upon deallocation; however, the TSF cannot extend this  
 7224 enforcement to that same file that exists on the off-line back-up. Therefore, that same file is still  
 7225 available. If this is a concern, then the PP/ST author **should** make sure that the proper  
 7226 environmental objectives are in place to support operational user guidance to address off-line  
 7227 objects.

7228 Residual information protection (FDP\_RIP) and Rollback (FDP\_ROL) **can** conflict when Residual  
 7229 information protection (FDP\_RIP) is instantiated to require that residual information be cleared  
 7230 at the time the application releases the object to the TSF (i.e. upon deallocation). Therefore, the  
 7231 Residual information protection (FDP\_RIP) selection of “deallocation” **should** not be used with  
 7232 Rollback (FDP\_ROL) since there would be no information to roll back. The other selection,  
 7233 “unavailability upon allocation”, **may** be used with Rollback (FDP\_ROL), but there is the risk that  
 7234 the resource which held the information has been allocated to a new object before the roll back  
 7235 took place. If that were to occur, then the roll back would not be possible.

7236 There are no audit requirements in Residual information protection (FDP\_RIP) because this is  
 7237 not a user-invokable function. Auditing of allocated or deallocated resources would be auditable  
 7238 as part of the access control SFP or the information flow control SFP operations.

7239 This family **should** apply to the objects specified in the access control SFP(s) or the information  
 7240 flow control SFP(s) as specified by the PP/ST author.

## 7241 **F.11.2 FDP\_RIP.1 Subset residual information protection**

### 7242 **F.11.2.1 User application notes**

7243 This component requires that, for a subset of the objects in the TOE, the TSF will ensure that  
 7244 there is no available residual information contained in a resource allocated to those objects or  
 7245 deallocated from those objects.

### 7246 **F.11.2.2 Operations**

#### 7247 **F.11.2.2.1 Selection**

7248 In FDP\_RIP.1.1, the PP/ST author **should** specify the event, allocation of the resource to or  
 7249 deallocation of the resource from, that invokes the residual information protection function.

#### 7250 **F.11.2.2.2 Assignment**

7251 In FDP\_RIP.1.1, the PP/ST author **should** specify the list of objects subject to residual  
 7252 information protection.

## 7253 **F.11.3 FDP\_RIP.2 Full residual information protection**

### 7254 **F.11.3.1 User application notes**

7255 This component requires that for all objects in the TOE, the TSF will ensure that there is no  
 7256 available residual information contained in a resource allocated to those objects or deallocated  
 7257 from those objects.

### 7258 **F.11.3.2 Operations**

#### 7259 **F.11.3.2.1 Selection**

7260 In FDP\_RIP.2.1, the PP/ST author **should** specify the event, allocation of the resource to or  
 7261 deallocation of the resource from, that invokes the residual information protection function.

## 7262 **F.12 Rollback (FDP\_ROL)**

### 7263 **F.12.1 User notes**

7264 This family addresses the need to return to a well-defined valid state, such as the need of a user  
 7265 to undo modifications to a file or to undo transactions in case of an incomplete series of  
 7266 transaction as in the case of databases.

7267 This family is intended to assist a user in returning to a well-defined valid state after the user  
 7268 undoes the last set of actions, or, in distributed databases, the return of all of the distributed  
 7269 copies of the databases to the state before an operation failed.

7270 Residual information protection (FDP\_RIP) and Rollback (FDP\_ROL) conflict when Residual  
 7271 information protection (FDP\_RIP) enforces that the contents will be made unavailable at the  
 7272 time that a resource is deallocated from an object. Therefore, this use of Residual information  
 7273 protection (FDP\_RIP) cannot be combined with Rollback (FDP\_ROL) as there would be no  
 7274 information to roll back. Residual information protection (FDP\_RIP) **can** be used only with  
 7275 Rollback (FDP\_ROL) when it enforces that the contents will be unavailable at the time that a  
 7276 resource is allocated to an object. This is because the Rollback (FDP\_ROL) mechanism will have  
 7277 an opportunity to access the previous information that **may** still be present in the TOE in order  
 7278 to successfully roll back the operation.

7279 The rollback requirement is bounded by certain limits.

**EXAMPLE**

For example, a text editor typically only allows you roll back up to a certain number of commands. Another example would be backups. If backup tapes are rotated, after a tape is reused, the information **can** no longer be retrieved. This also poses a bound on the rollback requirement.

7280 **F.12.2 FDP\_ROL.1 Basic rollback**7281 **F.12.2.1 User application notes**

7282 This component allows a user or subject to undo a set of operations on a predefined set of  
7283 objects. The undo is only possible within certain limits, for example up to a number of  
7284 characters or up to a time limit.

7285 **F.12.2.2 Operations**7286 **F.12.2.2.1 Assignment**

7287 In FDP\_ROL.1.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
7288 flow control SFP(s) that will be enforced when performing rollback operations. This is  
7289 necessary to make sure that roll back is not used to circumvent the specified SFPs.

7290 In FDP\_ROL.1.1, the PP/ST author **should** specify the list of operations that **can** be rolled back.

7291 In FDP\_ROL.1.1, the PP/ST author **should** specify the information and/or list of objects that are  
7292 subjected to the rollback policy.

7293 In FDP\_ROL.1.2, the PP/ST author **should** specify the boundary limit to which rollback  
7294 operations **may** be performed. The boundary **may** be specified as a predefined period of time,

**EXAMPLE**

operations **may** be undone which were performed within the past two minutes. Other possible boundaries **may** be defined as the maximum number of operations allowable or the size of a buffer.

7295 **F.12.3 FDP\_ROL.2 Advanced rollback**7296 **F.12.3.1 User application notes**

7297 This component enforces that the TSF provide the capability to rollback all operations;  
7298 however, the user **can** choose to rollback only a part of them.

7299 **F.12.3.2 Operations**7300 **F.12.3.2.1 Assignment**

7301 In FDP\_ROL.2.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
7302 flow control SFP(s) that will be enforced when performing rollback operations. This is  
7303 necessary to make sure that roll back is not used to circumvent the specified SFPs.

7304 In FDP\_ROL.2.1, the PP/ST author **should** specify the list of objects that are subjected to the  
7305 rollback policy.

7306 In FDP\_ROL.2.2, the PP/ST author **should** specify the boundary limit to which rollback  
7307 operations **may** be performed. The boundary **may** be specified as a predefined period of time,

**EXAMPLE**

for example, operations **may** be undone which were performed within the past two minutes.

7308 Other possible boundaries **may** be defined as the maximum number of operations allowable or  
7309 the size of a buffer.

7310 **F.13 Stored data confidentiality (FDP\_SDC)**7311 **F.13.1 User notes**7312 **F.13.2 FDP\_SDC.1 Stored data confidentiality**

7313 **F.13.2.1 User application notes**7314 **F.13.2.2 Operations**7315 **F.13.2.2.1 Assignment**7316 In FDP\_SDC.1.1 the PP/ST author **should**7317 **F.13.3 FDP\_SDC.2 Protection of data on disk**7318 **F.13.3.1 User application notes**

7319 Data characteristics **could be** data length (shorter or longer than a threshold), data type (binary,  
 7320 text, image, sound, video), data representation (binary, vector, character, frame) leading to the  
 7321 specification of a dedicated [selection: cryptographic, [assignment: other method]].

7322 **F.13.3.2 Evaluator application notes**

7323 dependencies to FCS\_COP.1 could be non-satisfied in practice if alternative method to  
 7324 cryptography is used in dedicated cases.

7325 **F.13.3.3 Operations**7326 **F.13.3.3.1 Assignment**7327 **F.14 Stored data integrity (FDP\_SDI)**7328 **F.14.1 User notes**

7329 This family provides requirements that address protection of user data while it is stored within  
 7330 containers controlled by the TSF.

7331 Hardware glitches or errors **may** affect data stored in memory. This family provides  
 7332 requirements to detect these unintentional errors. The integrity of user data while stored on  
 7333 storage devices controlled by the TSF are also addressed by this family.

7334 To prevent a subject from modifying the data, the Information flow control functions (FDP\_IFF)  
 7335 or Access control functions (FDP\_ACF) families are required (rather than this family).

7336 This family differs from Internal TOE transfer (FDP\_ITT) that protects the user data from  
 7337 integrity errors while being transferred within the TOE.

7338 **F.14.2 FDP\_SDI.1 Stored data integrity monitoring**7339 **F.14.2.1 User application notes**

7340 This component monitors data stored on media for integrity errors. The PP/ST author **can**  
 7341 specify different kinds of user data attributes that will be used as the basis for monitoring.

7342 **F.14.2.2 Operations**7343 **F.14.2.2.1 Assignment**7344 In FDP\_SDI.1.1, the PP/ST author **should** specify the integrity errors that the TSF will detect.

7345 In FDP\_SDI.1.1, the PP/ST author **should** specify the user data attributes that will be used as the  
 7346 basis for the monitoring.

7347 **F.14.3 FDP\_SDI.2 Stored data integrity monitoring and action**7348 **F.14.3.1 User application notes**

7349 This component monitors data stored on media for integrity errors. The PP/ST author **can**  
 7350 specify which action **should** be taken in case an integrity error is detected.

7351 **F.14.3.2 Operations**7352 **F.14.3.2.1 Assignment**7353 In FDP\_SDI.2.1, the PP/ST author **should** specify the integrity errors that the TSF will detect.



7354 In FDP\_SDI.2.1, the PP/ST author **should** specify the user data attributes that will be used as the  
7355 basis for the monitoring.

7356 In FDP\_SDI.2.2, the PP/ST author **should** specify the actions to be taken in case an integrity  
7357 error is detected.

## 7358 **F.15 Inter-TSF user data confidentiality transfer protection (FDP\_UCT)**

### 7359 **F.15.1 User notes**

7360 This family defines the requirements for ensuring the confidentiality of user data when it is  
7361 transferred using an external channel between the TOE and another trusted IT product.  
7362 Confidentiality is enforced by preventing unauthorized disclosure of user data in transit  
7363 between the two end points. The end points **may** be a TSF or a user.

7364 This family provides a requirement for the protection of user data during transit. In contrast,  
7365 Confidentiality of exported TSF data (FPT\_ITC) handles TSF data.

### 7366 **F.15.2 FDP\_UCT.1 Basic data exchange confidentiality**

#### 7367 **F.15.2.1 User application notes**

7368 Depending on the access control or information flow policies the TSF is required to send or  
7369 receive user data in a manner such that the confidentiality of the user data is protected.

#### 7370 **F.15.2.2 Operations**

##### 7371 **F.15.2.2.1 Assignment**

7372 In FDP\_UCT.1.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
7373 flow control SFP(s) that will be enforced when exchanging user data. The specified policies will  
7374 be enforced to make decisions about who **can** exchange data and which data **can** be exchanged.

##### 7375 **F.15.2.2.2 Selection**

7376 In FDP\_UCT.1.1, the PP/ST author **should** specify whether this element applies to a mechanism  
7377 that transmits or receives user data.

## 7378 **F.16 Inter-TSF user data integrity transfer protection (FDP\_UIT)**

### 7379 **F.16.1 User notes**

7380 This family defines the requirements for providing integrity for user data in transit between the  
7381 TSF and another trusted IT product and recovering from detectable errors. At a minimum, this  
7382 family monitors the integrity of user data for modifications. Furthermore, this family supports  
7383 different ways of correcting detected integrity errors.

7384 This family defines the requirements for providing integrity for user data in transit; while  
7385 Integrity of exported TSF data (FPT\_ITI) handles TSF data.

7386 Inter-TSF user data integrity transfer protection (FDP\_UIT) and Inter-TSF user data  
7387 confidentiality transfer protection (FDP\_UCT) are duals of each other, as Inter-TSF user data  
7388 confidentiality transfer protection (FDP\_UCT) addresses user data confidentiality. Therefore,  
7389 the same mechanism that implements Inter-TSF user data integrity transfer protection  
7390 (FDP\_UIT) **could** possibly be used to implement other families such as Inter-TSF user data  
7391 confidentiality transfer protection (FDP\_UCT) and Import from outside of the TOE (FDP\_ITC).

### 7392 **F.16.2 FDP\_UIT.1 Data exchange integrity**

#### 7393 **F.16.2.1 User application notes**

7394 Depending on the access control or information flow policies the TSF is required to send or  
7395 receive user data in a manner such that modification of the user data is detected. There is no  
7396 requirement for a TSF mechanism to attempt to recover from the modification.

#### 7397 **F.16.2.2 Operations**

7398 **F.16.2.2.1 Assignment**

7399 In FDP\_UIT.1.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
 7400 flow control SFP(s) that will be enforced on the transmitted data or on the received data. The  
 7401 specified policies will be enforced to make decisions about who **can** transmit or who **can** receive  
 7402 data, and which data **can** be transmitted or received.

7403 **F.16.2.2.2 Selection**

7404 In FDP\_UIT.1.1, the PP/ST author **should** specify whether this element applies to a TSF that is  
 7405 transmitting or receiving objects.

7406 In FDP\_UIT.1.1, the PP/ST author **should** specify whether the data **should** be protected from  
 7407 modification, deletion, insertion, or replay.

7408 In FDP\_UIT.1.2, the PP/ST author **should** specify whether the errors of the type: modification,  
 7409 deletion, insertion, or replay are detected.

7410 **F.16.3 FDP\_UIT.2 Source data exchange recovery**7411 **F.16.3.1 User application notes**

7412 This component provides the ability to recover from a set of identified transmission errors, if  
 7413 required, with the help of the other trusted IT product. As the other trusted IT product is  
 7414 outside the TOE, the TSF cannot control its behaviour. However, it **can** provide functions that  
 7415 have the ability to cooperate with the other trusted IT product for the purposes of recovery.

**EXAMPLE**

For example, the TSF **could** include functions that depend upon the source trusted IT product to re-send the data in the event that an error is detected.

7416 This component deals with the ability of the TSF to handle such an error recovery.

7417 **F.16.3.2 Operations**7418 **F.16.3.2.1 Assignment**

7419 In FDP\_UIT.2.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
 7420 flow control SFP(s) that will be enforced when recovering user data. The specified policies will  
 7421 be enforced to make decisions about which data **can** be recovered and how it **can** be recovered.

7422 In FDP\_UIT.2.1, the PP/ST author **should** specify the list of integrity errors from which the TSF,  
 7423 with the help of the source trusted IT product, is be able to recover the original user data.

7424 **F.16.4 FDP\_UIT.3 Destination data exchange recovery**7425 **F.16.4.1 User application notes**

7426 This component provides the ability to recover from a set of identified transmission errors. It  
 7427 accomplishes this task without help from the source trusted IT product. For example, if certain  
 7428 errors are detected, the transmission protocol must be robust enough to allow the TSF to  
 7429 recover from the error based on checksums and other information available within that  
 7430 protocol.

7431 **F.16.4.2 Operations**7432 **F.16.4.2.1 Assignment**

7433 In FDP\_UIT.3.1, the PP/ST author **should** specify the access control SFP(s) and/or information  
 7434 flow control SFP(s) that will be enforced when recovering user data. The specified policies will  
 7435 be enforced to make decisions about which data **can** be recovered and how it **can** be recovered.

7436 In FDP\_UIT.3.1, the PP/ST author **should** specify the list of integrity errors from which the  
 7437 receiving TSF, alone, is able to recover the original user data.



## Annex G (normative)

### Class FIA: Identification and authentication- application notes

#### G.1 General information

A common security requirement is to unambiguously identify the person and/or entity performing functions in a TOE. This involves not only establishing the claimed identity of each user, but also verifying that each user is indeed who he/she claims to be. This is achieved by requiring users to provide the TSF with some information that is known by the TSF to be associated with the user in question.

Families in this class address the requirements for functions to establish and verify a claimed user identity. Identification and Authentication is required to ensure that users are associated with the proper security attributes

##### EXAMPLE

Security attributes include identity, groups, roles, security, or integrity levels.

The unambiguous identification of authorized users and the correct association of security attributes with users and subjects is critical to the enforcement of the security policies.

The Authentication failures (FIA\_AFL) family addresses defining limits on repeated unsuccessful authentication attempts.

The Authentication proof of identity (FIA\_API) family...

The User attribute definition (FIA\_ATD) family address the definition of user attributes that are used in the enforcement of the SFRs.

The Specification of secrets (FIA\_SOS) family addresses the generation and verification of secrets that satisfy a defined metric.

The User authentication (FIA\_UAU) family addresses verifying the identity of a user.

The User identification (FIA\_UID) family addresses determining the identity of a user.

The User-subject binding (FIA\_USB) family addresses the correct association of security attributes for each authorized user.

#### G.2 Authentication failures (FIA\_AFL)

##### G.2.1 User notes

This family addresses requirements for defining values for authentication attempts and TSF actions in cases of authentication attempt failure. Parameters include, but are not limited to, the number of attempts and time thresholds.

The session establishment process is the interaction with the user to perform the session establishment independent of the actual implementation. If the number of unsuccessful authentication attempts exceeds the indicated threshold, either the user account or the terminal (or both) will be locked. If the user account is disabled, the user cannot log-on to the system. If the terminal is disabled, the terminal (or the address that the terminal has) cannot be used for any log-on. Both of these situations continue until the condition for re-establishment is satisfied.

##### G.2.2 FIA\_AFL.1 Authentication failure handling

###### G.2.2.1 User application notes

7478 The PP/ST author **may** define the number of unsuccessful authentication attempts or **may**  
 7479 choose to let the TOE developer or the authorized user to define this number. The unsuccessful  
 7480 authentication attempts need not be consecutive, but rather related to an authentication event.  
 7481 Such an authentication event **could** be the count from the last successful session establishment  
 7482 at a given terminal.

7483 The PP/ST author **could** specify a list of actions that the TSF **shall** take in the case of  
 7484 authentication failure. An authorized administrator **could** also be allowed to manage the events,  
 7485 if deemed opportune by the PP/ST author. These actions **could** be, among other things, terminal  
 7486 deactivation, user account deactivation, or administrator alarm. The conditions under which the  
 7487 situation will be restored to normal must be specified on the action.

7488 In order to prevent denial of service, TOEs usually ensure that there is at least one user account  
 7489 that cannot be disabled.

7490 Further actions for the TSF **can** be stated by the PP/ST author, including rules for re-enabling  
 7491 the user session establishment process, or sending an alarm to the administrator.

EXAMPLE

Examples of these actions are: until a specified time has lapsed, until the authorized administrator re-enables the terminal/account, a time related to failed previous attempts (every time the attempt fails, the disabling time is doubled).

## 7492 G.2.2.2 Operations

### 7493 G.2.2.2.1 Selection

7494 In FIA\_AFL.1 Authentication failure handling, the PP/ST author **should** select either the  
 7495 assignment of a positive integer, or the phrase “an administrator configurable positive integer”  
 7496 specifying the range of acceptable values.

### 7497 G.2.2.2.2 Assignment

7498 In FIA\_AFL.1 Authentication failure handling, the PP/ST author **should** specify the  
 7499 authentication events. Examples of these authentication events are: the unsuccessful  
 7500 authentication attempts since the last successful authentication for the indicated user identity,  
 7501 the unsuccessful authentication attempts since the last successful authentication for the current  
 7502 terminal, the number of unsuccessful authentication attempts in the last 10 minutes. At least  
 7503 one authentication event must be specified.

7504 In FIA\_AFL.1 Authentication failure handling, if the assignment of a positive integer is selected,  
 7505 the PP/ST author **should** specify the default number (positive integer) of unsuccessful  
 7506 authentication attempts that, when met or surpassed, will trigger the events.

7507 In FIA\_AFL.1 Authentication failure handling, if an administrator configurable positive integer is  
 7508 selected, the PP/ST author **should** specify the range of acceptable values from which the  
 7509 administrator of the TOE **may** configure the number of unsuccessful authentication attempts.  
 7510 The number of authentication attempts **should** be less than or equal to the upper bound and  
 7511 greater or equal to the lower bound values.

### 7512 G.2.2.2.3 Selection

7513 In FIA\_AFL.1.2, the PP/ST author **should** select whether the event of meeting or surpassing the  
 7514 defined number of unsuccessful authentication attempts **shall** trigger an action by the TSF.

### 7515 G.2.2.2.4 Assignment

7516 In FIA\_AFL.1.2, the PP/ST author **should** specify the actions to be taken in case the threshold is  
 7517 met or surpassed, as selected. These actions **could** be disabling of an account for 5 minutes,  
 7518 disabling the terminal for an increasing amount of time (2 to the power of the number of  
 7519 unsuccessful attempts in seconds), or disabling of the account until unlocked by the  
 7520 administrator and simultaneously informing the administrator. The actions **should** specify the

7521 measures and if applicable the duration of the measure (or the conditions under which the  
7522 measure will be ended).

### 7523 **G.3 Authentication proof of identity (FIA\_API)**

#### 7524 **G.3.1 User notes**

7525 The other families of the Class FIA describe only the authentication verification of users'  
7526 identity performed by the TOE and do not describe the functionality of the user to prove their  
7527 identity. The following paragraph defines the extended family FIA\_API from point of view of a  
7528 TOE proving its identity.

#### 7529 **G.3.2 FIA\_API.1 Authentication proof of identity**

7530 **Editor's Note:**

7531 **Editors request contributions for the application notes for this family.**

##### 7532 **G.3.2.1 User application notes**

##### 7533 **G.3.2.2 Operations**

##### 7534 **G.3.2.2.1 Assignment**

### 7535 **G.4 User attribute definition (FIA\_ATD)**

#### 7536 **G.4.1 User notes**

7537 All authorized users **may** have a set of security attributes, other than the user's identity, that are  
7538 used to enforce the SFRs. This family defines the requirements for associating user security  
7539 attributes with users as needed to support the TSF in making security decisions.

7540 There are dependencies on the individual security policy (SFP) definitions. These individual  
7541 definitions **should** contain the listing of attributes that are necessary for policy enforcement.

#### 7542 **G.4.2 FIA\_ATD.1 User attribute definition**

##### 7543 **G.4.2.1 User application notes**

7544 This component specifies the security attributes that **should** be maintained at the level of the  
7545 user. This means that the security attributes listed are assigned to and **can** be changed at the  
7546 level of the user. In other words, changing a security attribute in this list associated with a user  
7547 **should** have no impact on the security attributes of any other user.

7548 In case security attributes belong to a group of users (such as Capability List for a group), the  
7549 user will need to have a reference (as security attribute) to the relevant group.

##### 7550 **G.4.2.2 Operations**

##### 7551 **G.4.2.2.1 Assignment**

7552 In FIA\_ATD.1.1, the PP/ST author **should** specify the security attributes that are associated to an  
7553 individual user.

#### **EXAMPLE**

An example of such a list is {"clearance", "group identifier", "rights"}.

### 7554 **G.5 Specification of secrets (FIA\_SOS)**

#### 7555 **G.5.1 User notes**

7556 This family defines requirements for mechanisms that enforce defined quality metrics on  
7557 provided secrets and generate secrets to satisfy the defined metric. Examples of such  
7558 mechanisms **may** include automated checking of user supplied passwords, or automated  
7559 password generation.

7560 A secret **can** be generated outside the TOE

EXAMPLE

selected by the user and introduced in the TOE.

7561 In such cases, the FIA\_SOS.1 Verification of secrets component **can** be used to ensure that the  
7562 external generated secret adheres to certain standards, for example a minimum size, not  
7563 present in a dictionary, and/or not previously used.

7564 Secrets **can** also be generated by the TOE. In those cases, the FIA\_SOS.2 TSF Generation of  
7565 secrets component **can** be used to require the TOE to ensure that the secrets that will adhere to  
7566 some specified metrics.

7567 Secrets contain the authentication data provided by the user for an authentication mechanism  
7568 that is based on knowledge the user possesses. When cryptographic keys are employed, the  
7569 class FCS: Cryptographic support **should** be used instead of this family.

## 7570 **G.5.2 FIA\_SOS.1 Verification of secrets**

### 7571 **G.5.2.1 User application notes**

7572 Secrets **can** be generated by the user. This component ensures that those user generated secrets  
7573 **can** be verified to meet a certain quality metric.

### 7574 **G.5.2.2 Operations**

#### 7575 **G.5.2.2.1 Assignment**

7576 In FIA\_SOS.1.1, the PP/ST author **should** provide a defined quality metric. The quality metric  
7577 specification **can** be as simple as a description of the quality checks to be performed, or as  
7578 formal as a reference to a government published standard that defines the quality metrics that  
7579 secrets must meet.

EXAMPLE

quality metrics **could** include a description of the alphanumeric structure of acceptable secrets and/or the space size that acceptable secrets must meet.

## 7580 **G.5.3 FIA\_SOS.2 TSF Generation of secrets**

### 7581 **G.5.3.1 User application notes**

7582 This component allows the TSF to generate secrets for specific functions such as authentication  
7583 by means of passwords.

7584 When a pseudo-random number generator is used in a secret generation algorithm, it **should**  
7585 accept as input random data that would provide output that has a high degree of  
7586 unpredictability. This random data (seed) **can** be derived from a number of available  
7587 parameters such as a system clock, system registers, date, time, etc. The parameters **should** be  
7588 selected to ensure that the number of unique seeds that **can** be generated from these inputs  
7589 **should** be at least equal to the minimum number of secrets that must be generated.

### 7590 **G.5.3.2 Operations**

#### 7591 **G.5.3.2.1 Assignment**

7592 In FIA\_SOS.2.1, the PP/ST author **should** provide a defined quality metric. The quality metric  
7593 specification **can** be as simple as a description of the quality checks to be performed or as  
7594 formal as a reference to a government published standard that defines the quality metrics that  
7595 secrets must meet.

EXAMPLE

quality metrics **could** include a description of the alphanumeric structure of acceptable secrets and/or the space size that acceptable secrets must meet.

7596 In FIA\_SOS.2.2, the PP/ST author **should** provide a list of TSF functions for which the TSF  
 7597 generated secrets must be used. An example of such a function **could** include a password-based  
 7598 authentication mechanism.

## 7599 **G.6 User authentication (FIA\_UAU)**

### 7600 **G.6.1 User notes**

7601 This family defines the types of user authentication mechanisms supported by the TSF. This  
 7602 family defines the required attributes on which the user authentication mechanisms must be  
 7603 based.

### 7604 **G.6.2 FIA\_UAU.1 Timing of authentication**

#### 7605 **G.6.2.1 User application notes**

7606 This component requires that the PP/ST author define the TSF-mediated actions that **can** be  
 7607 performed by the TSF on behalf of the user before the claimed identity of the user is  
 7608 authenticated. The TSF-mediated actions **should** have no security concerns with users  
 7609 incorrectly identifying themselves prior to being authenticated. For all other TSF-mediated  
 7610 actions not in the list, the user must be authenticated before the action **can** be performed by the  
 7611 TSF on behalf of the user.

7612 This component cannot control whether the actions **can** also be performed before the  
 7613 identification took place. This requires the use of either FIA\_UID.1 Timing of identification or  
 7614 FIA\_UID.2 User identification before any action with the appropriate assignments.

#### 7615 **G.6.2.2 Operations**

##### 7616 **G.6.2.2.1 Assignment**

7617 In FIA\_UAU.1.1, the PP/ST author **should** specify a list of TSF-mediated actions that **can** be  
 7618 performed by the TSF on behalf of a user before the claimed identity of the user is  
 7619 authenticated. This list cannot be empty. If no actions are appropriate, component FIA\_UAU.2  
 7620 User authentication before any action **should** be used instead.

#### EXAMPLE

Such an action might include the request for help on the login procedure.

### 7621 **G.6.3 FIA\_UAU.2 User authentication before any action**

#### 7622 **G.6.3.1 User application notes**

7623 This component requires that a user is authenticated before any other TSF-mediated action **can**  
 7624 take place on behalf of that user.

### 7625 **G.6.4 FIA\_UAU.3 Unforgeable authentication**

#### 7626 **G.6.4.1 User application notes**

7627 This component addresses requirements for mechanisms that provide protection of  
 7628 authentication data. Authentication data that is copied from another user, or is in some way  
 7629 constructed **should** be detected and/or rejected. These mechanisms provide confidence that  
 7630 users authenticated by the TSF are actually who they claim to be.

7631 This component **may** be useful only with authentication mechanisms that are based on  
 7632 authentication data that cannot be shared. It is impossible for a TSF to detect or prevent the  
 7633 sharing of passwords outside the control of the TSF.

#### EXAMPLE

An example of authentication data that cannot be shared is biometrics

7634 **Editors' Note**

7635 **Is this a good example? Editors' consider replay attacks could be "sharing" biometrics.**

#### 7636 **G.6.4.2 Operations**

##### 7637 **G.6.4.2.1 Selection**

7638 In FIA\_UAU.3.1, the PP/ST author **should** specify whether the TSF will detect, prevent, or detect  
7639 and prevent forging of authentication data.

7640 In FIA\_UAU.3.2, the PP/ST author **should** specify whether the TSF will detect, prevent, or detect  
7641 and prevent copying of authentication data.

#### 7642 **G.6.5 FIA\_UAU.4 Single-use authentication mechanisms**

##### 7643 **G.6.5.1 User application notes**

7644 This component addresses requirements for authentication mechanisms based on single-use  
7645 authentication data. Single-use authentication data **can** be something the user has or knows, but  
7646 not something the user is.

###### **EXAMPLE**

Single-use authentication data include single-use passwords, encrypted time-stamps, and/or random numbers from a secret lookup table.

7647 The PP/ST author **can** specify to which authentication mechanism(s) this requirement applies.

##### 7648 **G.6.5.2 Operations**

###### 7649 **G.6.5.2.1 Assignment**

7650 In FIA\_UAU.4.1, the PP/ST author **should** specify the list of authentication mechanisms to which  
7651 this requirement applies. This assignment **can** be "all authentication mechanisms". An example  
7652 of this assignment **could** be "the authentication mechanism employed to authenticate people on  
7653 the external network".

#### 7654 **G.6.6 FIA\_UAU.5 Multiple authentication mechanisms**

##### 7655 **G.6.6.1 User application notes**

7656 The use of this component allows specification of requirements for more than one  
7657 authentication mechanism to be used within a TOE. For each distinct mechanism, applicable  
7658 requirements must be chosen from the FIA: Identification and authentication class to be applied  
7659 to each mechanism. It is possible that the same component **could** be selected multiple times in  
7660 order to reflect different requirements for the different use of the authentication mechanism.

7661 The management functions in the class FMT **may** provide maintenance capabilities for the set of  
7662 authentication mechanisms, as well as the rules that determine whether the authentication was  
7663 successful.

7664 To allow anonymous users to interact with the TOE, a "none" authentication mechanism **can** be  
7665 incorporated. The use of such access **should** be clearly explained in the rules of FIA\_UAU.5.2.

##### 7666 **G.6.6.2 Operations**

###### 7667 **G.6.6.2.1 Assignment**

7668 In FIA\_UAU.5.1, the PP/ST author **should** define the available authentication mechanisms.

###### **EXAMPLE**

Such a list **could** be: "none, password mechanism, biometric (retinal scan), S/key mechanism".

7669 In FIA\_UAU.5.2, the PP/ST author **should** specify the rules that describe how the authentication  
7670 mechanisms provide authentication and when each is to be used. This means that for each  
7671 situation the set of mechanisms that might be used for authenticating the user must be  
7672 described.

**EXAMPLE**

A list of such rules is: “if the user has special privileges a password mechanism and a biometric mechanism both **shall** be used, with success only if both succeed; for all other users a password mechanism **shall** be used.”

The PP/ST author might give the boundaries within which the authorized administrator **may** specify specific rules. An example of a rule is: “the user **shall** always be authenticated by means of a token; the administrator might specify additional authentication mechanisms that also must be used.” The PP/ST author also might choose not to specify any boundaries but leave the authentication mechanisms and their rules completely up to the authorized administrator.

**G.6.7 FIA\_UAU.6 Re-authenticating****G.6.7.1 User application notes**

This component addresses potential needs to re-authenticate users at defined points in time. These **may** include user requests for the TSF to perform security relevant actions, as well as requests from non-TSF entities for re-authentication.

**EXAMPLE**

A server application requesting that the TSF re-authenticate the client it is serving.

**G.6.7.2 Operations****G.6.7.2.1 Assignment**

In FIA\_UAU.6.1, the PP/ST author **should** specify the list of conditions requiring re-authentication. This list **could** include a specified user inactivity period that has elapsed, the user requesting a change in active security attributes, or the user requesting the TSF to perform some security critical function.

The PP/ST author might give the boundaries within which the re-authentication **should** occur and leave the specifics to the authorized administrator.

**EXAMPLE**

“the user **shall** always be re-authenticated at least once a day; the administrator might specify that the re-authentication **should** happen more often but not more often than once every 10 minutes.”

**G.6.8 FIA\_UAU.7 Protected authentication feedback****G.6.8.1 User application notes**

This component addresses the feedback on the authentication process that will be provided to the user. In some systems, the feedback consists of indicating how many characters have been typed but not showing the characters themselves, in other systems even this information might not be appropriate.

This component requires that the authentication data is not provided as-is back to the user. In a workstation environment, it **could** display a “dummy” for each password character provided, and not the original character.

**Example**

A “dummy” **could** be a star “\*” character.

**G.6.8.2 Operations****G.6.8.2.1 Assignment**

In FIA\_UAU.7 Protected authentication feedback, the PP/ST author **should** specify the feedback related to the authentication process that will be provided to the user.



**EXAMPLE**

A feedback assignment **could** be “the number of characters typed”, another type of feedback is “the authentication mechanism that failed the authentication”.

7705 **G.7 User identification (FIA\_UID)**

7706 **G.7.1 User notes**

7707 This family defines the conditions under which users are required to identify themselves before  
7708 performing any other actions that are to be mediated by the TSF and that require user  
7709 identification.

7710 **G.7.2 FIA\_UID.1 Timing of identification**

7711 **G.7.2.1 User application notes**

7712 This component poses requirements for the user to be identified. The PP/ST author **can** indicate  
7713 specific actions that **can** be performed before the identification takes place.

7714 If FIA\_UID.1 Timing of identification is used, the TSF-mediated actions mentioned in FIA\_UID.1  
7715 Timing of identification **should** also appear in this FIA\_UAU.1 Timing of authentication.

7716 **G.7.2.2 Operations**

7717 **G.7.2.2.1 Assignment**

7718 In FIA\_UID.1.1, the PP/ST author **should** specify a list of TSF-mediated actions that **can** be  
7719 performed by the TSF on behalf of a user before the user has to identify itself. If no actions are  
7720 appropriate, component FIA\_UID.2 User identification before any action **should** be used instead.  
7721 An example of such an action might include the request for help on the login procedure.

7722 **G.7.3 FIA\_UID.2 User identification before any action**

7723 **G.7.3.1 User application notes**

7724 In this component users will be identified. A user is not allowed by the TSF to perform any  
7725 action before being identified.

7726 **G.8 User-subject binding (FIA\_USB)**

7727 **G.8.1 User notes**

7728 An authenticated user, in order to use the TOE, typically activates a subject. The user's security  
7729 attributes are associated (totally or partially) with this subject. This family defines  
7730 requirements to create and maintain the association of the user's security attributes to a subject  
7731 acting on the user's behalf.

7732 **G.8.2 FIA\_USB.1 User-subject binding**

7733 **G.8.2.1 User application notes**

7734 It is intended that a subject is acting on behalf of the user who caused the subject to come into  
7735 being or to be activated to perform a certain task.

7736 Therefore, when a subject is created, that subject is acting on behalf of the user who initiated  
7737 the creation. In cases where anonymity is used, the subject is still acting on behalf of a user, but  
7738 the identity of that user is unknown. A special category of subjects is those subjects that serve  
7739 multiple users. In such cases the user that created this subject is assumed to be the “owner”.

**EXAMPLE**

An example of a user is a server process.

7740 **G.8.2.2 Operations**

7741 **G.8.2.2.1 Assignment**



- 7742 In FIA\_USB.1.1, the PP/ST author **should** specify a list of the user security attributes that are to  
7743 be bound to subjects.
- 7744 In FIA\_USB.1.2, the PP/ST author **should** specify any rules that are to apply upon initial  
7745 association of attributes with subjects, or “none”.
- 7746 In FIA\_USB.1.3, the PP/ST author **should** specify any rules that are to apply when changes are  
7747 made to the user security attributes associated with subjects acting on behalf of users, or  
7748 “none”.

## Annex H (normative)

### Class FMT: Security management- application notes

#### H.1 General information

This class specifies the management of several aspects of the TSF: security attributes, TSF data and functions in the TSF. The different management roles and their interaction, such as separation of capability, **can** also be specified.

In an environment where the TOE is made up of multiple physically separated parts, the timing issues with respect to propagation of security attributes, TSF data, and function modification become very complex, especially if the information is required to be replicated across the parts of the TOE. This **should** be considered when selecting components such as FMT\_REV.1 Revocation, or FMT\_SAE.1 Time-limited authorization, where the behaviour might be impaired. In such situations, use of components from Internal TOE TSF data replication consistency (FPT\_TRC) is advisable.

#### H.2 Limited capabilities and availability (FMT\_LIM)

##### H.2.1 User notes

The functional requirements FMT\_LIM.1 and FMT\_LIM.2 assume that there are two types of mechanisms (limitation of capabilities and limitation of availability) which together **shall** provide protection in order to enforce the policy. This also allows that

- a) the TSF is provided without restrictions in the product in its user environment but its capabilities are so limited that the policy is enforced or conversely
- b) the TSF is designed with high functionality but is removed or disabled in the product in its user environment.

The combination of both requirements **shall** enforce the policy.

##### H.2.2 FMT\_LIM.1 Limited capabilities

###### H.2.2.1 User application notes

###### H.2.2.2 Operations

###### H.2.2.2.1 Selection

In FMT\_LIM.1.1, the PP/ST author **should** select whether the role **can** determine the behaviour of, disable, enable, and/or modify the behaviour of the security functions.

###### H.2.2.2.2 Assignment

In FMT\_LIM.1.1, the PP/ST author **should** specify the functions that **can** be modified by the identified roles. Examples include auditing and time determination.

In FMT\_LIM.1.1, the PP/ST author **should** specify the roles that are allowed to modify the functions in the TSF. The possible roles are specified in FMT\_SMR.1 Security roles.

##### H.2.3 FMT\_LIM.2 Limited availability

###### H.2.3.1 User application notes

###### H.2.3.2 Operations

###### H.2.3.2.1 Assignment

#### H.3 Management of functions in TSF (FMT\_MOF)

### H.3.1 User notes

The TSF management functions enable authorized users to set up and control the secure operation of the TOE. These administrative functions typically fall into a number of different categories:

- a) Management functions that relate to access control, accountability and authentication controls enforced by the TOE. For example, definition and update of user security characteristics or definition and update of auditing system controls, definition and update of per-user policy attributes, definition of known system access control labels, and control and management of user groups.

#### EXAMPLE

User security characteristics: unique identifiers associated with user names, user accounts, system entry parameters

Auditing system controls: selection of audit events, management of audit trails, audit trail analysis, and audit report generation

User policy attributes: user clearance

- b) Management functions that relate to controls over availability. For example, definition and update of availability parameters or resource quotas.
- c) Management functions that relate to general installation and configuration. For example, TOE configuration, manual recovery, installation of TOE security fixes (if any), repair and reinstallation of hardware.
- d) Management functions that relate to routine control and maintenance of TOE resources. For example, enabling and disabling peripheral devices, mounting of removable storage media, backup, and recovery.

NOTE These functions need to be present in a TOE based on the families included in the PP or ST. It is the responsibility of the PP/ST author to ensure that adequate functions will be provided to manage the TOE in a secure fashion.

The TSF might contain functions that **can** be controlled by an administrator. For example, the auditing functions **could** be switched off, the time synchronization **could** be switchable, and/or the authentication mechanism **could** be modifiable.

### H.3.2 FMT\_MOF.1 Management of security functions behaviour

#### H.3.2.1 User application notes

This component allows identified roles to manage the security functions of the TSF. This might entail obtaining the current status of a security function, disabling, or enabling the security function, or modifying the behaviour of the security function.

#### EXAMPLE

modifying the behaviour of the security functions is changing of authentication mechanisms.

#### H.3.2.2 Operations

##### H.3.2.2.1 Selection

In FMT\_MOF.1.1, the PP/ST author **should** select whether the role **can** determine the behaviour of, disable, enable, and/or modify the behaviour of the security functions.

##### H.3.2.2.2 Assignment

In FMT\_MOF.1.1, the PP/ST author **should** specify the functions that **can** be modified by the identified roles. Examples include auditing and time determination.

In FMT\_MOF.1.1, the PP/ST author **should** specify the roles that are allowed to modify the functions in the TSF. The possible roles are specified in FMT\_SMR.1 Security roles.

## 7827 **H.4 Management of security attributes (FMT\_MSA)**

### 7828 **H.4.1 User notes**

7829 This family defines the requirements on the management of security attributes.

7830 Security attributes affect the behaviour of the TSF.

#### EXAMPLE

Examples of security attributes are the groups to which a user belongs, the roles he/she might assume, the priority of a process (subject), and the rights belonging to a role or a user.

7831 These security attributes might need to be managed by the user, a subject, a specific authorized  
7832 user (a user with explicitly given rights for this management) or inherit values according to a  
7833 given policy/set of rules.

7834 It is noted that the right to assign rights to users is itself a security attribute and/or potentially  
7835 subject to management by FMT\_MSA.1 Management of security attributes.

7836 FMT\_MSA.2 Secure security attributes **can** be used to ensure that any accepted combination of  
7837 security attributes is within a secure state. The definition of what “secure” means is left to the  
7838 TOE guidance.

7839 In some instances, subjects, objects, or user accounts are created. If no explicit values for the  
7840 related security attributes are given, default values need to be used. FMT\_MSA.1 Management of  
7841 security attributes **can** be used to specify that these default values **can** be managed.

### 7842 **H.4.2 FMT\_MSA.1 Management of security attributes**

#### 7843 **H.4.2.1 User application notes**

7844 This component allows users acting in certain roles to manage identified security attributes.  
7845 The users are assigned to a role within the component FMT\_SMR.1 Security roles.

7846 The default value of a parameter is the value the parameter takes when it is instantiated  
7847 without specifically assigned values. An initial value is provided during the instantiation  
7848 (creation) of a parameter and overrides the default value.

#### 7849 **H.4.2.2 Operations**

##### 7850 **H.4.2.2.1 Assignment**

7851 In FMT\_MSA.1.1, the PP/ST author **should** list the access control SFP(s) or the information flow  
7852 control SFP(s) for which the security attributes are applicable.

##### 7853 **H.4.2.2.2 Selection**

7854 In FMT\_MSA.1.1, the PP/ST author **should** specify the operations that **can** be applied to the  
7855 identified security attributes. The PP/ST author **can** specify that the role **can** modify the default  
7856 value (change\_default), query, modify the security attribute, delete the security attributes  
7857 entirely or define their own operation.

##### 7858 **H.4.2.2.3 Assignment**

7859 In FMT\_MSA.1.1, the PP/ST author **should** specify the security attributes that **can** be operated  
7860 on by the identified roles. It is possible for the PP/ST author to specify that the default value  
7861 such as default access-rights **can** be managed.

#### EXAMPLE

Examples of these security attributes are user-clearance, priority of service level, access control list, default access rights.

7862 In FMT\_MSA.1.1, the PP/ST author **should** specify the roles that are allowed to operate on the  
7863 security attributes. The possible roles are specified in FMT\_SMR.1 Security roles.

7864 In FMT\_MSA.1.1, if selected, the PP/ST author **should** specify which other operations the role  
7865 **could** perform.

EXAMPLE

An example of such an operation **could** be “create”.

## 7866 H.4.3 FMT\_MSA.2 Secure security attributes

### 7867 H.4.3.1 User application notes

7868 This component contains requirements on the values that **can** be assigned to security attributes.  
7869 The assigned values **should** be such that the TOE will remain in a secure state.

7870 The definition of what “secure” means is not answered in this component but is left to the  
7871 development of the TOE and the resulting information in the guidance. An example **could** be  
7872 that if a user account is created, it **should** have a non-trivial password.

### 7873 H.4.3.2 Operations

#### 7874 H.4.3.2.1 Assignment

7875 In FMT\_MSA.2.1, the PP/ST author **should** specify the list of security attributes that require only  
7876 secure values to be provided.

## 7877 H.4.4 FMT\_MSA.3 Static attribute initialization

### 7878 H.4.4.1 User application notes

7879 This component requires that the TSF provide default values for relevant object security  
7880 attributes, which **can** be overridden by an initial value. It **may** still be possible for a new object  
7881 to have different security attributes at creation if a mechanism exists to specify the permissions  
7882 at time of creation.

### 7883 H.4.4.2 Operations

#### 7884 H.4.4.2.1 Assignment

7885 In FMT\_MSA.3.1, the PP/ST author **should** list the access control SFP or the information flow  
7886 control SFP for which the security attributes are applicable.

#### 7887 H.4.4.2.2 Selection

7888 In FMT\_MSA.3.1, the PP/ST author **should** select whether the default property of the access  
7889 control attribute will be restrictive, permissive, or another property. Only one of these options  
7890 **may** be chosen.

#### 7891 H.4.4.2.3 Assignment

7892 In FMT\_MSA.3.1, if the PP/ST author selects another property, the PP/ST author **should** specify  
7893 the desired characteristics of the default values.

7894 In FMT\_MSA.3.2, the PP/ST author **should** specify the roles that are allowed to modify the  
7895 values of the security attributes. The possible roles are specified in FMT\_SMR.1 Security roles.

## 7896 H.4.5 FMT\_MSA.4 Security attribute value inheritance

### 7897 H.4.5.1 User application notes

7898 This component requires specification of the set of rules through which the security attribute  
7899 inherits values and the conditions to be met for these rules to be applied.

### 7900 H.4.5.2 Operations

#### 7901 H.4.5.2.1 Assignment

7902 In FMT\_MSA.4.1, the PP/ST author specifies the rules governing the value that will be inherited  
7903 by the specified security attribute, including the conditions that are to be met for the rules to be  
7904 applied.

**EXAMPLE**

For example, if a new file or directory is created (in a multilevel filesystem), its label is the label at which the user is logged in at the time it is created.

7905 **H.5 Management of TSF data (FMT\_MTD)**7906 **H.5.1 User notes**

7907 This component imposes requirements on the management of TSF data. Examples of TSF data  
7908 are the current time and the audit trail.

**EXAMPLE**

this family allows the specification of whom **can** read, delete, or create the audit trail.

7909 **H.5.2 FMT\_MTD.1 Management of TSF data**7910 **H.5.2.1 User application notes**

7911 This component allows users with a certain role to manage values of TSF data. The users are  
7912 assigned to a role within the component FMT\_SMR.1 Security roles.

7913 The default value of a parameter is the values the parameter takes when it is instantiated  
7914 without specifically assigned values. An initial value is provided during the instantiation  
7915 (creation) of a parameter and overrides the default value.

7916 **H.5.2.2 Operations**7917 **H.5.2.2.1 Selection**

7918 In FMT\_MTD.1.1, the PP/ST author **should** specify the operations that **can** be applied to the  
7919 identified TSF data. The PP/ST author **can** specify that the role **can** modify the default value  
7920 (change\_default), clear, query or modify the TSF data, or delete the TSF data entirely. If so  
7921 desired the PP/ST author **could** specify any type of operation. To clarify “clear TSF data” means  
7922 that the content of the TSF data is removed, but that the entity that stores the TSF data remains  
7923 in the TOE.

7924 **H.5.2.2.2 Assignment**

7925 In FMT\_MTD.1.1, the PP/ST author **should** specify the TSF data that **can** be operated on by the  
7926 identified roles. It is possible for the PP/ST author to specify that the default value **can** be  
7927 managed.

7928 In FMT\_MTD.1.1, the PP/ST author **should** specify the roles that are allowed to operate on the  
7929 TSF data. The possible roles are specified in FMT\_SMR.1 Security roles.

7930 In FMT\_MTD.1.1, if selected, the PP/ST author **should** specify which other operations the role  
7931 **could** perform. An example **could** be “create”.

7932 **H.5.3 FMT\_MTD.2 Management of limits on TSF data**7933 **H.5.3.1 User application notes**

7934 This component specifies limits on TSF data, and actions to be taken if these limits are  
7935 exceeded. This component will allow limits on the size of the audit trail to be defined, and  
7936 specification of the actions to be taken when these limits are exceeded.

7937 **H.5.3.2 Operations**7938 **H.5.3.2.1 Assignment**

7939 In FMT\_MTD.2.1, the PP/ST author **should** specify the TSF data that **can** have limits, and the  
7940 value of those limits. An example of such TSF data is the number of users logged-in.

7941 In FMT\_MTD.2.1, the PP/ST author **should** specify the roles that are allowed to modify the limits  
 7942 on the TSF data and the actions to be taken. The possible roles are specified in FMT\_SMR.1  
 7943 Security roles.

7944 In FMT\_MTD.2.2, the PP/ST author **should** specify the actions to be taken if the specified limit  
 7945 on the specified TSF data is exceeded.

EXAMPLE

An example of such TSF action is that the authorized user is informed and an audit record is generated.

## 7946 **H.5.4 FMT\_MTD.3 Secure TSF data**

### 7947 **H.5.4.1 User application notes**

7948 This component covers requirements on the values that **can** be assigned to TSF data. The  
 7949 assigned values **should** be such that the TOE will remain in a secure state.

7950 The definition of what “secure” means is not answered in this component but is left to the  
 7951 development of the TOE and the resulting information in the guidance.

### 7952 **H.5.4.2 Operations**

#### 7953 **H.5.4.2.1 Assignment**

7954 In FMT\_MTD.3.1, the PP/ST author **should** specify what TSF data require only secure values to  
 7955 be accepted.

## 7956 **H.6 Revocation (FMT\_REV)**

### 7957 **H.6.1 User notes**

7958 This family addresses revocation of security attributes for a variety of entities within a TOE.

### 7959 **H.6.2 FMT\_REV.1 Revocation**

#### 7960 **H.6.2.1 User application notes**

7961 This component specifies requirements on the revocation of rights. It requires the specification  
 7962 of the revocation rules. Examples are:

- 7963 a) Revocation will take place on the next login of the user;
- 7964 b) Revocation will take place on the next attempt to open the file;
- 7965 c) Revocation will take place within a fixed time. This might mean that all open
- 7966 connections are re-evaluated every x minutes.

#### 7967 **H.6.2.2 Operations**

##### 7968 **H.6.2.2.1 Assignment**

7969 In FMT\_REV.1.1, the PP/ST author **should** specify which security attributes are to be revoked  
 7970 when a change is made to the associated object/subject/user/other resource.

##### 7971 **H.6.2.2.2 Selection**

7972 In FMT\_REV.1.1, the PP/ST author **should** specify whether the ability to revoke security  
 7973 attributes from users, subjects, objects, or any additional resources **shall** be provided by the  
 7974 TSF.

##### 7975 **H.6.2.2.3 Assignment**

7976 In FMT\_REV.1.1, the PP/ST author **should** specify the roles that are allowed to modify the  
 7977 functions in the TSF. The possible roles are specified in FMT\_SMR.1 Security roles.

7978 In FMT\_REV.1.1, the PP/ST author **should**, if additional resources is selected, specify whether  
 7979 the ability to revoke their security attributes **shall** be provided by the TSF.



7980 In FMT\_REV.1.2, the PP/ST author **should** specify the revocation rules. Examples of these rules  
 7981 **could** include: “prior to the next operation on the associated resource”, or “for all new subject  
 7982 creations”.

## 7983 **H.7 Security attribute expiration (FMT\_SAE)**

### 7984 **H.7.1 User notes**

7985 This family addresses the capability to enforce time limits for the validity of security attributes.  
 7986 This family **can** be applied to specify expiration requirements for access control attributes,  
 7987 identification and authentication attributes, certificates, audit attributes, etc.

#### EXAMPLE

An example of a certificate is key certificates such as ANSI X509.

### 7988 **H.7.2 FMT\_SAE.1 Time-limited authorization**

#### 7989 **H.7.2.1 Operations**

##### 7990 **H.7.2.1.1 Assignment**

7991 In FMT\_SAE.1.1, the PP/ST author **should** provide the list of security attributes for which  
 7992 expiration is to be supported.

#### EXAMPLE

An example of such an attribute might be a user's security clearance.

7993 In FMT\_SAE.1.1, the PP/ST author **should** specify the roles that are allowed to modify the  
 7994 security attributes in the TSF. The possible roles are specified in FMT\_SMR.1 Security roles.

7995 In FMT\_SAE.1.2, the PP/ST author **should** provide a list of actions to be taken for each security  
 7996 attribute when it expires. An example might be that the user's security clearance, when it  
 7997 expires, is set to the lowest allowable clearance on the TOE. If immediate revocation is desired  
 7998 by the PP/ST, the action “immediate revocation” **should** be specified.

## 7999 **H.8 Specification of Management Functions (FMT\_SMF)**

### 8000 **H.8.1 User notes**

8001 This family allows the specification of the management functions to be provided by the TOE.  
 8002 Each security management function that is listed in fulfilling the assignment is either security  
 8003 attribute management, TSF data management, or security function management.

### 8004 **H.8.2 FMT\_SMF.1 Specification of Management Functions**

#### 8005 **H.8.2.1 User application notes**

8006 This component specifies the management functions to be provided.

8007 PP/ST authors **should** consult the “Management” subclauses for components included in their  
 8008 PP/ST to provide a basis for the management functions to be listed via this component.

#### 8009 **H.8.2.2 Operations**

##### 8010 **H.8.2.2.1 Assignment**

8011 In FMT\_SMF.1.1, the PP/ST author **should** specify the management functions to be provided by  
 8012 the TSF, either security attribute management, TSF data management, or security function  
 8013 management.

## 8014 **H.9 Security management roles (FMT\_SMR)**

### 8015 **H.9.1 User notes**



8016 This family reduces the likelihood of damage resulting from users abusing their authority by  
 8017 taking actions outside their assigned functional responsibilities. It also addresses the threat that  
 8018 inadequate mechanisms have been provided to securely administer the TSF.

8019 This family requires that information be maintained to identify whether a user is authorized to  
 8020 use a particular security-relevant administrative function.

8021 Some management actions **can** be performed by users, others only by designated people within  
 8022 the organization. This family allows the definition of different roles, such as owner, auditor,  
 8023 administrator, daily-management.

8024 The roles as used in this family are security related roles. Each role **can** encompass an extensive  
 8025 set of capabilities or **can** be a single right. This family defines the roles. The capabilities of the  
 8026 role are defined in Limited capabilities and availability (FMT\_LIM), Management of security  
 8027 attributes (FMT\_MSA) and Management of TSF data (FMT\_MTD).

EXAMPLE

Set of capabilities: root in UNIX

Single right: right to read a single object such as the helpfile.

8028 Some type of roles might be mutually exclusive.

EXAMPLE

the daily-management might be able to define and activate users but might not be able to remove users (which is reserved for the administrator (role)).

8029 This class will allow policies such as two-person control to be specified.

## 8030 **H.9.2 FMT\_SMR.1 Security roles**

### 8031 **H.9.2.1 User application notes**

8032 This component specifies the different roles that the TSF **should** recognize. Often the system  
 8033 distinguishes between the owner of an entity, an administrator, and other users.

### 8034 **H.9.2.2 Operations**

#### 8035 **H.9.2.2.1 Assignment**

8036 In FMT\_SMR.1.1, the PP/ST author **should** specify the roles that are recognized by the system.  
 8037 These are the roles that users **could** occupy with respect to security. Examples are: owner,  
 8038 auditor, and administrator.

## 8039 **H.9.3 FMT\_SMR.2 Restrictions on security roles**

### 8040 **H.9.3.1 User application notes**

8041 This component specifies the different roles that the TSF **should** recognize, and conditions on  
 8042 how those roles **could** be managed. Often the system distinguishes between the owner of an  
 8043 entity, an administrator, and other users.

8044 The conditions on those roles specify the interrelationship between the different roles, as well  
 8045 as restrictions on when the role **can** be assumed by a user.

### 8046 **H.9.3.2 Operations**

#### 8047 **H.9.3.2.1 Assignment**

8048 In FMT\_SMR.2.1, the PP/ST author **should** specify the roles that are recognized by the system.  
 8049 These are the roles that users **could** occupy with respect to security. Examples are: owner,  
 8050 auditor, administrator.

8051 In FMT\_SMR.2.3, the PP/ST author **should** specify the conditions that govern role assignment.  
 8052 Examples of these conditions are: "an account cannot have both the auditor and administrator  
 8053 role" or "a user with the assistant role must also have the owner role".

8054 **H.9.4 FMT\_SMR.3 Assuming roles**

8055 **H.9.4.1 User application notes**

8056 This component specifies that an explicit request must be given to assume the specific role.

8057 **H.9.4.2 Operations**

8058 **H.9.4.2.1 Assignment**

8059 In FMT\_SMR.3.1, the PP/ST author **should** specify the roles that require an explicit request to be  
8060 assumed.

EXAMPLE

auditor and administrator.

## Annex I (normative)

### Class FPR: Privacy- application notes

#### I.1 General information

This class describes the requirements that **could** be levied to satisfy the users' privacy needs, while still allowing the system flexibility as far as possible to maintain sufficient control over the operation of the system.

In the components of this class there is flexibility as to whether or not authorized users are covered by the required security functionality.

##### EXAMPLE

a PP/ST author might consider it appropriate not to require protection of the privacy of users against a suitably authorized user.

This class, together with other classes (such as those concerned with audit, access control, trusted path, and non-repudiation) provides the flexibility to specify the desired privacy behaviour. On the other hand, the requirements in this class might impose limitations on the use of the components of other classes, such as FIA: Identification and authentication or FAU: Security audit.

##### EXAMPLE

If authorized users are not allowed to see the user identity (perhaps because of Anonymity or Pseudonymity), it will obviously not be possible to hold individual users accountable for any security relevant actions they perform that are covered by the privacy requirements. However, it **may** still be possible to include audit requirements in a PP/ST, where the fact that a particular security relevant event has occurred is more important than knowing who was responsible for it.

Additional information is provided in the application notes for class FAU: Security audit, where it is explained that the definition of "identity" in the context of auditing **can** also be an alias or other information that **could** identify a user.

This class describes four families: Anonymity, Pseudonymity, Unlinkability and Unobservability. Anonymity, Pseudonymity and Unlinkability have a complex interrelationship. When choosing a family, the choice **should** depend on the threats identified. For some types of privacy threats, pseudonymity will be more appropriate than anonymity.

##### EXAMPLE

If there is a requirement for auditing.

In addition, some types of privacy threats are best countered by a combination of components from several families.

All families assume that a user does not explicitly perform an action that discloses the user's own identity.

##### EXAMPLE

The TSF is not expected to screen the user name in electronic messages or databases.

All families in this class have components that **can** be scoped through operations. These operations allow the PP/ST author to state the cooperating users/subjects to which the TSF must be resistant.

**EXAMPLE**

An instantiation of anonymity **could** be: “The TSF shall ensure that the users and/or subjects are unable to determine the user identity bound to the teleconsulting application”.

It is noted that the TSF should not only provide this protection against individual users, but also against users cooperating to obtain the information.

8090 **I.2 Anonymity (FPR\_ANO)**8091 **I.2.1 User notes**

8092 Anonymity ensures that a subject **may** use a resource or service without disclosing its user  
8093 identity.

8094 The intention of this family is to specify that a user or subject might take action without  
8095 releasing its user identity to others such as users, subjects, or objects. The family provides the  
8096 PP/ST author with a means to identify the set of users that cannot see the identity of someone  
8097 performing certain actions.

8098 Therefore, if a subject, using anonymity, performs an action, another subject will not be able to  
8099 determine either the identity or even a reference to the identity of the user employing the  
8100 subject. The focus of the anonymity is on the protection of the user's identity, not on the  
8101 protection of the subject identity; hence, the identity of the subject is not protected from  
8102 disclosure.

8103 Although the identity of the subject is not released to other subjects or users, the TSF is not  
8104 explicitly prohibited from obtaining the users identity. In case the TSF is not allowed to know  
8105 the identity of the user, FPR\_ANO.2 Anonymity without soliciting information **could** be invoked.  
8106 In that case, the TSF **should** not request the user information.

8107 The interpretation of “determine” **should** be taken in the broadest sense of the word.

8108 The Components leveling and description distinguishes between the users and an authorized  
8109 user. An authorized user is often excluded from the component, and therefore allowed to  
8110 retrieve a user's identity. However, there is no specific requirement that an authorized user  
8111 must be able to have the capability to determine the user's identity. For ultimate privacy, the  
8112 components would be used to say that no user or authorized user **can** see the identity of anyone  
8113 performing any action.

8114 Although some systems will provide anonymity for all services that are provided, other systems  
8115 provide anonymity for certain subjects/operations. To provide this flexibility, an operation is  
8116 included where the scope of the requirement is defined. If the PP/ST author wants to address  
8117 all subjects/operations, the words “all subjects and all operations” **could** be provided.

8118 Possible applications include the ability to make enquiries of a confidential nature to public  
8119 databases, respond to electronic polls, or make anonymous payments or donations.

**EXAMPLE**

Potential hostile users or subjects include providers, system operators, communication partners and users, who smuggle malicious parts (including malware) into systems. All of these users can investigate usage patterns (such as which users used which services) and misuse this information.

8120 **I.2.2 FPR\_ANO.1 Anonymity**8121 **I.2.2.1 User application notes**

8122 This component ensures that the identity of a user is protected from disclosure. There **may** be  
8123 instances, however, that a given authorized user **can** determine who performed certain actions.  
8124 This component gives the flexibility to capture either a limited or total privacy policy.

8125 **I.2.2.2 Operations**8126 **I.2.2.2.1 Assignment**

8127 In FPR\_ANO.1.1, the PP/ST author **should** specify the set of users and/or subjects against which  
 8128 the TSF must provide protection. For example, even if the PP/ST author specifies a single user  
 8129 or subject role, the TSF must not only provide protection against each individual user or subject  
 8130 but must protect with respect to cooperating users and/or subjects.

EXAMPLE

A set of users **could** be a group of users which can operate under the same role or can all use the same process(es).

8131 In FPR\_ANO.1.1, the PP/ST author **should** identify the list of subjects and/or operations and/or  
 8132 objects where the real user name of the subject **should** be protected.

EXAMPLE

“the voting application”.

### 8133 **I.2.3 FPR\_ANO.2 Anonymity without soliciting information**

#### 8134 **I.2.3.1 User application notes**

8135 This component is used to ensure that the TSF is not allowed to know the identity of the user.

#### 8136 **I.2.3.2 Operations**

##### 8137 **I.2.3.2.1 Assignment**

8138 In FPR\_ANO.2.1, the PP/ST author **should** specify the set of users and/or subjects against which  
 8139 the TSF must provide protection. For example, even if the PP/ST author specifies a single user  
 8140 or subject role, the TSF must not only provide protection against each individual user or subject  
 8141 but must protect with respect to cooperating users and/or subjects.

EXAMPLE

A set of users **could** be a group of users which can operate under the same role or can all use the same process(es).

8142 In FPR\_ANO.2.1, the PP/ST author **should** identify the list of subjects and/or operations and/or  
 8143 objects where the real user name of the subject **should** be protected.

EXAMPLE

“the voting application”.

8144 In FPR\_ANO.2.2, the PP/ST author **should** identify the list of services which are subject to the  
 8145 anonymity requirement, for example, “the accessing of job descriptions”.

8146 In FPR\_ANO.2.2, the PP/ST author **should** identify the list of subjects from which the real user  
 8147 name of the subject **should** be protected when the specified services are provided.

### 8148 **I.3 Pseudonymity (FPR\_PSE)**

#### 8149 **I.3.1 User notes**

8150 Pseudonymity ensures that a user **may** use a resource or service without disclosing its identity  
 8151 but **can** still be accountable for that use. The user **can** be accountable by directly being related to  
 8152 a reference (alias) held by the TSF, or by providing an alias that will be used for processing  
 8153 purposes, such as an account number.

8154 In several respects, pseudonymity resembles anonymity. Both pseudonymity and anonymity  
 8155 protect the identity of the user, but in pseudonymity a reference to the user's identity is  
 8156 maintained for accountability or other purposes.

8157 The component FPR\_PSE.1 Pseudonymity does not specify the requirements on the reference to  
 8158 the user's identity. For the purpose of specifying requirements on this reference two sets of  
 8159 requirements are presented: FPR\_PSE.2 Reversible pseudonymity and FPR\_PSE.3 Alias  
 8160 pseudonymity.

8161 A way to use the reference is by being able to obtain the original user identity.

## EXAMPLE

In a digital cash environment, it would be advantageous to be able to trace the user's identity when a check has been issued multiple times (i.e. fraud).

8162 In general, the user's identity needs to be retrieved under specific conditions. The PP/ST author  
8163 might want to incorporate FPR\_PSE.2 Reversible pseudonymity to describe those services.

8164 Another usage of the reference is as an alias for a user.

## EXAMPLE

A user who does not wish to be identified, **can** provide an account to which the resource utilization **should** be charged. In such cases, the reference to the user identity is an alias for the user, where other users or subjects **can** use the alias for performing their functions without ever obtaining the user's identity (for example, statistical operations on use of the system). In this case, the PP/ST author might wish to incorporate FPR\_PSE.3 Alias pseudonymity to specify the rules to which the reference must conform.

8165 Using these constructs above, digital money **can** be created using FPR\_PSE.2 Reversible  
8166 pseudonymity specifying that the user identity will be protected and, if so specified in the  
8167 condition, that there be a requirement to trace the user identity if the digital money is spent  
8168 twice. When the user is honest, the user identity is protected; if the user tries to cheat, the user  
8169 identity **can** be traced.

8170 A different kind of system **could** be a digital credit card, where the user will provide a  
8171 pseudonym that indicates an account from which the cash **can** be subtracted. In such cases, for  
8172 example, FPR\_PSE.3 Alias pseudonymity **could** be used. This component would specify that the  
8173 user identity will be protected and, furthermore, that the same user will only get assigned  
8174 values for which he/she has provided money (if so specified in the conditions).

8175 It **should** be realized that the more stringent components potentially cannot be combined with  
8176 other requirements, such as identification and authentication or audit. The interpretation of  
8177 "determine the identity" **should** be taken in the broadest sense of the word. The information is  
8178 not provided by the TSF during the operation, nor **can** the entity determine the subject or the  
8179 owner of the subject that invoked the operation, nor will the TSF record information, available  
8180 to the users or subjects, which might release the user identity in the future.

8181 The intent is that the TSF not reveal any information that would compromise the identity of the  
8182 user,

## EXAMPLE

The identity of subjects acting on the user's behalf.

8183 The information that is considered to be sensitive depends on the effort an attacker is capable  
8184 of spending.

8185 Possible applications include the ability to charge a caller for premium rate telephone services  
8186 without disclosing his or her identity, or to be charged for the anonymous use of an electronic  
8187 payment system.

## EXAMPLE

Potential hostile users include providers, system operators, communication partners and users, who smuggle malicious parts (including malware) into systems. All of these attackers **can** investigate which users used which services and misuse this information. Additionally, to Anonymity services, Pseudonymity Services contains methods for authorization without identification, especially for anonymous payment ("Digital Cash"). This helps providers to obtain their payment in a secure way while maintaining customer anonymity.

## 8188 I.3.2 FPR\_PSE.1 Pseudonymity

### 8189 I.3.2.1 User application notes

8190 This component provides the user protection against disclosure of identity to other users. The  
8191 user will remain accountable for its actions.

### 8192 I.3.2.2 Operations

**I.3.2.2.1 Assignment**

In FPR\_PSE.1.1, the PP/ST author **should** specify the set of users and/or subjects against which the TSF must provide protection. For example, even if the PP/ST author specifies a single user or subject role, the TSF must not only provide protection against each individual user or subject but must protect with respect to cooperating users and/or subjects.

**EXAMPLE**

A set of users **could** be a group of users which **can** operate under the same role or **can** all use the same process(es).

In FPR\_PSE.1.1, the PP/ST author **should** identify the list of subjects and/or operations and/or objects where the real user name of the subject **should** be protected.

**EXAMPLE**

“the accessing of job offers”.

Note “objects” includes any other attributes that might enable another user or subject to derive the actual identity of the user.

In FPR\_PSE.1.2, the PP/ST author **should** identify the (one or more) number of aliases the TSF is able to provide.

In FPR\_PSE.1.2, the PP/ST author **should** identify the list of subjects to whom the TSF is able to provide an alias.

**I.3.2.2.2 Selection**

In FPR\_PSE.1.3, the PP/ST author **should** specify whether the user alias is generated by the TSF or supplied by the user. Only one of these options **may** be chosen.

**I.3.2.2.3 Assignment**

In FPR\_PSE.1.3, the PP/ST author **should** identify the metric to which the TSF-generated or user-generated alias **should** conform.

**I.3.3 FPR\_PSE.2 Reversible pseudonymity****I.3.3.1 User application notes**

In this component, the TSF **shall** ensure that under specified conditions the user identity related to a provided reference **can** be determined.

In FPR\_PSE.1 Pseudonymity the TSF **shall** provide an alias instead of the user identity. When the specified conditions are satisfied, the user identity to which the alias belong **can** be determined.

**EXAMPLE**

Such a condition in an electronic cash environment is: “The TSF shall provide the notary a capability to determine the user identity based on the provided alias only under the conditions that a check has been issued twice.”

**I.3.3.2 Operations****I.3.3.2.1 Assignment**

In FPR\_PSE.2.1, the PP/ST author **should** specify the set of users and/or subjects against which the TSF must provide protection.

**EXAMPLE**

Even if the PP/ST author specifies a single user or subject role, the TSF must not only provide protection against each individual user or subject but must protect with respect to cooperating users and/or subjects. A set of users, for example, **could** be a group of users which **can** operate under the same role or **can** all use the same process(es).

In FPR\_PSE.2.1, the PP/ST author **should** identify the list of subjects and/or operations and/or objects where the real user name of the subject **should** be protected.



## EXAMPLE

“the accessing of job offers”.

8225

8226 NOTE “objects” includes any other attributes that might enable another user or subject to derive the actual  
8227 identity of the user.

8228 In FPR\_PSE.2.2, the PP/ST author **should** identify the (one or more) number of aliases the TSF,  
8229 is able to provide.

8230 In FPR\_PSE.2.2, the PP/ST author **should** identify the list of subjects to whom the TSF is able to  
8231 provide an alias.

8232 **I.3.3.2.2 Selection**

8233 In FPR\_PSE.2.3, the PP/ST author **should** specify whether the user alias is generated by the TSF  
8234 or supplied by the user. Only one of these options **may** be chosen.

8235 **I.3.3.2.3 Assignment**

8236 In FPR\_PSE.2.3, the PP/ST author **should** identify the metric to which the TSF-generated or  
8237 user-generated alias **should** conform.

8238 **I.3.3.2.4 Selection**

8239 In FPR\_PSE.2.4, the PP/ST author **should** select whether the authorized user and/or trusted  
8240 subjects **can** determine the real user name.

8241 **I.3.3.2.5 Assignment**

8242 In FPR\_PSE.2.4, the PP/ST author **should** identify the list of conditions under which the trusted  
8243 subjects and authorized user **can** determine the real user name based on the provided  
8244 reference. These conditions **can** be conditions such as time of day, or they **can** be administrative  
8245 such as on a court order.

8246 In FPR\_PSE.2.4, the PP/ST author **should** identify the list of trusted subjects that **can** obtain the  
8247 real user name under a specified condition.

## EXAMPLE

a notary or special authorized user.

8248 **I.3.4 FPR\_PSE.3 Alias pseudonymity**8249 **I.3.4.1 User application notes**

8250 In this component, the TSF **shall** ensure that the provided reference meets certain construction  
8251 rules, and thereby **can** be used in a secure way by potentially insecure subjects.

8252 If a user wants to use disk resources without disclosing its identity, pseudonymity **can** be used.  
8253 However, every time the user accesses the system, the same alias must be used. Such conditions  
8254 **can** be specified in this component.

8255 **I.3.4.2 Operations**8256 **I.3.4.2.1 Assignment**

8257 In FPR\_PSE.3.1, the PP/ST author **should** specify the set of users and/or subjects against which  
8258 the TSF must provide protection. For example, even if the PP/ST author specifies a single user  
8259 or subject role, the TSF must not only provide protection against each individual user or subject  
8260 but must protect with respect to cooperating users and/or subjects.

## EXAMPLE

A set of users **could** be a group of users which can operate under the same role or can all use the same process(es).



8261 In FPR\_PSE.3.1, the PP/ST author **should** identify the list of subjects and/or operations and/or  
 8262 objects where the real user name of the subject **should** be protected.

EXAMPLE

“the accessing of job offers”.

8263

8264 NOTE “objects” includes any other attributes which might enable another user or subject to derive the actual  
 8265 identity of the user.

8266 In FPR\_PSE.3.2, the PP/ST author **should** identify the (one or more) number of aliases the TSF is  
 8267 able to provide.

8268 In FPR\_PSE.3.2, the PP/ST author **should** identify the list of subjects to whom the TSF is able to  
 8269 provide an alias.

#### 8270 **I.3.4.2.2 Selection**

8271 In FPR\_PSE.3.3, the PP/ST author **should** specify whether the user alias is generated by the TSF,  
 8272 or supplied by the user. Only one of these options **may** be chosen.

#### 8273 **I.3.4.2.3 Assignment**

8274 In FPR\_PSE.3.3, the PP/ST author **should** identify the metric to which the TSF-generated or  
 8275 user-generated alias **should** conform.

8276 In FPR\_PSE.3.4, the PP/ST author **should** identify the list of conditions that indicate when the  
 8277 used reference for the real user name **shall** be identical and when it **shall** be different, for  
 8278 example, “when the user logs on to the same host” it will use a unique alias.

### 8279 **I.4 Distribution of trust (FPR\_TRD)**

#### 8280 **I.4.1 User notes**

#### 8281 **I.4.2 FPR\_TRD.1 Administrative domains**

##### 8282 **I.4.2.1 User application notes**

#### 8283 **I.4.3 FPR\_TRD.2 Allocation of information assets**

##### 8284 **I.4.3.1 User application notes**

##### 8285 **I.4.3.2 Operations**

##### 8286 **I.4.3.2.1 Assignment**

8287 In FPR\_TRD.2.3, the PP/ST author **should**

#### 8288 **I.4.4 FPR\_TRD.3 Allocation of processing activities**

##### 8289 **I.4.4.1 User application notes**

##### 8290 **I.4.4.2 Operations**

##### 8291 **I.4.4.2.1 Assignment**

8292 In FPR\_TRD.3.3, the PP/ST author **should**

### 8293 **I.5 Unlinkability (FPR\_UNL)**

#### 8294 **I.5.1 User notes**

8295 Unlinkability ensures that a user **may** make multiple uses of resources or services without  
 8296 others being able to link these uses together. Unlinkability differs from pseudonymity that,  
 8297 although in pseudonymity the user is also not known, relations between different actions **can** be  
 8298 provided.

8299 The requirements for unlinkability are intended to protect the user identity against the use of  
8300 profiling of the operations.

EXAMPLE

For example, when a telephone smart card is employed with a unique number, the telephone company can determine the behaviour of the user of this telephone card. When a telephone profile of the users is known, the card can be linked to a specific user.

8301 Hiding the relationship between different invocations of a service or access of a resource will  
8302 prevent this kind of information gathering.

8303 As a result, a requirement for unlinkability **could** imply that the subject and user identity of an  
8304 operation must be protected. Otherwise this information might be used to link operations  
8305 together.

8306 Unlinkability requires that different operations cannot be related. This relationship **can** take  
8307 several forms.

EXAMPLE

The user associated with the operation, or the terminal which initiated the action, or the time the action was executed.

8308 The PP/ST author **can** specify what kind of relationships are present that must be countered.

8309 Possible applications include the ability to make multiple use of a pseudonym without creating  
8310 a usage pattern that might disclose the user's identity.

EXAMPLE

Potential hostile subjects and users include providers, system operators, communication partners and users, who smuggle malicious parts, (including malware) into systems, they do not operate but want to get information about. All of these attackers **can** investigate (such as which users used which services) and misuse this information.

8311 Unlinkability protects users from linkages, which **could** be drawn between several actions of a  
8312 customer.

EXAMPLE

a series of phone calls made by an anonymous customer to different partners, where the combination of the partner's identities might disclose the identity of the customer.

## 8313 **I.5.2 FPR\_UNL.1 Unlinkability**

### 8314 **I.5.2.1 User application notes**

8315 This component ensures that users cannot link different operations in the system and thereby  
8316 obtain information.

### 8317 **I.5.2.2 Operations**

#### 8318 **I.5.2.2.1 Assignment**

8319 In FPR\_UNL.1.1, the PP/ST author **should** specify the set of users and/or subjects against which  
8320 the TSF must provide protection.

EXAMPLE

Even if the PP/ST author specifies a single user or subject role, the TSF must not only provide protection against each individual user or subject but must protect with respect to cooperating users and/or subjects.

8321

EXAMPLE

A set of users **could** be a group of users which can operate under the same role or can all use the same process(es).

8322 In FPR\_UNL.1.1, the PP/ST author **should** identify the list of operations which **should** be  
8323 subjected to the unlinkability requirement.

## EXAMPLE

“sending email”.

8324 **I.5.2.2.2 Selection**

8325 In FPR\_UNL.1.1, the PP/ST author **should** select the relationships that **should** be obscured. The  
8326 selection allows either the user identity or an assignment of relations to be specified.

8327 **I.5.2.2.3 Assignment**

8328 In FPR\_UNL.1.1, the PP/ST author **should** identify the list of relations which **should** be protected  
8329 against.

## EXAMPLE

“originate from the same IP address”.

8330 **I.5.3 FPR\_UNL.2 Unlinkability of users**8331 **I.5.3.1 User application notes**8332 **I.5.3.2 Operations**8333 **I.5.3.2.1 Assignment**8334 **I.5.4 FPR\_UNL.3 Unlinkability of subjects**8335 **I.5.4.1 User application notes**8336 **I.5.4.2 Operations**8337 **I.5.4.2.1 Assignment**8338 **I.6 Unobservability (FPR\_UNO)**8339 **I.6.1 User notes**

8340 Unobservability ensures that a user **may** use a resource or service without others, especially  
8341 third parties, being able to observe that the resource or service is being used.

8342 Unobservability approaches the user identity from a different direction than the previous  
8343 families Anonymity, Pseudonymity and Unlinkability. In this case, the intent is to hide the use of  
8344 a resource or service, rather than to hide the user's identity.

8345 A number of techniques **can** be applied to implement unobservability.

## EXAMPLE

Examples of techniques to provide unobservability are:

- a) Allocation of information impacting unobservability: Unobservability relevant information (such as information that describes that an operation occurred) can be allocated in several locations within the TOE. The information might be allocated to a single randomly chosen part of the TOE such that an attacker does not know which part of the TOE should be attacked. An alternative system might distribute the information such that no single part of the TOE has sufficient information that, if circumvented, the privacy of the user would be compromised. This technique is explicitly addressed in FPR\_UNO.2 Allocation of information impacting unobservability.
- b) Broadcast: When information is broadcast (such as Internet and Radio frequencies, including Ethernet, Bluetooth, WiFi and Near-field communication bands), users cannot determine who actually received and used that information. This technique is especially useful when information should reach receivers which have to fear a stigma for being interested in that information (such as sensitive medical information).
- c) Cryptographic protection and message padding: People observing a message stream might obtain information from the fact that a message is transferred and from attributes on that message. By traffic padding, message padding and encrypting the message stream, the transmission of a message and its attributes can be protected.

8346 Sometimes, users **should** not see the use of a resource, but an authorized user must be allowed  
 8347 to see the use of the resource in order to perform his duties. In such cases, the FPR\_UNO.4  
 8348 Authorized user observability **could** be used, which provides the capability for one or more  
 8349 authorized users to see the usage.

8350 This family makes use of the concept “parts of the TOE”. This is considered any part of the TOE  
 8351 that is either physically or logically separated from other parts of the TOE.

8352 Unobservability of communications **may** be an important factor in many areas, such as the  
 8353 enforcement of constitutional rights, organizational policies, or in defense related applications.

## 8354 **I.6.2 FPR\_UNO.1 Unobservability**

### 8355 **I.6.2.1 User application notes**

8356 This component requires that the use of a function or resource cannot be observed by  
 8357 unauthorized users.

### 8358 **I.6.2.2 Operations**

#### 8359 **I.6.2.2.1 Assignment**

8360 In FPR\_UNO.1.1, the PP/ST author **should** specify the list of users and/or subjects against which  
 8361 the TSF must provide protection.

#### EXAMPLE

Even if the PP/ST author specifies a single user or subject role, the TSF must not only provide protection against each individual user or subject but must protect with respect to cooperating users and/or subjects.

8362

#### EXAMPLE

A set of users **could** be a group of users which **can** operate under the same role or **can** all use the same process(es).

8363 In FPR\_UNO.1.1, the PP/ST author **should** identify the list of operations that are subjected to the  
 8364 unobservability requirement. Other users/subjects will then not be able to observe the  
 8365 operations on a covered object in the specified list.

#### EXAMPLE

reading and writing to the object.

8366 In FPR\_UNO.1.1, the PP/ST author **should** identify the list of objects which are covered by the  
 8367 unobservability requirement.

#### EXAMPLE

a specific mail server or ftp site.

8368 In FPR\_UNO.1.1, the PP/ST author **should** specify the set of protected users and/or subjects  
 8369 whose unobservability information will be protected.

#### EXAMPLE

“Users accessing the system through the internet”.

## 8370 **I.6.3 FPR\_UNO.2 Allocation of information impacting unobservability**

### 8371 **I.6.3.1 User application notes**

8372 This component requires that the use of a function or resource cannot be observed by specified  
 8373 users or subjects. Furthermore, this component specifies that information related to the privacy  
 8374 of the user is distributed within the TOE such that attackers might not know which part of the  
 8375 TOE to target, or they need to attack multiple parts of the TOE.

8376 An example of the use of this component is the use of a randomly allocated node to provide a  
 8377 function. In such a case the component might require that the privacy related information **shall**

8378 only be available to one identified part of the TOE and will not be communicated outside this  
8379 part of the TOE.

EXAMPLE

A more complex example can be found in some “voting algorithms”. Several parts of the TOE will be involved in the service, but no individual part of the TOE will be able to violate the policy. So, a person may cast a vote (or not) without the TOE being able to determine whether a vote has been cast and what the vote happened to be (unless the vote was unanimous).

### 8380 I.6.3.2 Operations

#### 8381 I.6.3.2.1 Assignment

8382 In FPR\_UNO.2.1, the PP/ST author **should** specify the list of users and/or subjects against which  
8383 the TSF must provide protection. For example, even if the PP/ST author specifies a single user  
8384 or subject role, the TSF must not only provide protection against each individual user or subject  
8385 but must protect with respect to cooperating users and/or subjects.

EXAMPLE

A set of users **could** be a group of users which **can** operate under the same role or **can** all use the same process(es).

8386 In FPR\_UNO.2.1, the PP/ST author **should** identify the list of operations that are subjected to the  
8387 unobservability requirement. Other users/subjects will then not be able to observe the  
8388 operations on a covered object in the specified list

EXAMPLE

Reading and writing to the object.

8389 In FPR\_UNO.2.1, the PP/ST author **should** identify the list of objects which are covered by the  
8390 unobservability requirement. An example **could** be a specific mail server or ftp site.

8391 In FPR\_UNO.2.1, the PP/ST author **should** specify the set of protected users and/or subjects  
8392 whose unobservability information will be protected.

EXAMPLE

“users accessing the system through the internet”.

8393 In FPR\_UNO.2.2, the PP/ST author **should** identify which privacy related information **should** be  
8394 distributed in a controlled manner.

EXAMPLE

This information **could** include: IP address of subject, IP address of object, time, used encryption keys.

8395 In FPR\_UNO.2.2, the PP/ST author **should** specify the conditions to which the dissemination of  
8396 the information **should** adhere. These conditions **should** be maintained throughout the lifetime  
8397 of the privacy related information of each instance.

EXAMPLE

Examples of these conditions **could** be:

- “the information shall only be present at a single separated part of the TOE and shall not be communicated outside this part of the TOE.”,
- “the information shall only reside in a single separated part of the TOE, but shall be moved to another part of the TOE periodically”;
- “the information shall be distributed between the different parts of the TOE such that compromise of any 5 separated parts of the TOE will not compromise the security policy”.

### 8398 I.6.4 FPR\_UNO.3 Unobservability without soliciting information

#### 8399 I.6.4.1 User application notes

8400 This component is used to require that the TSF does not try to obtain information that might  
8401 compromise unobservability when provided specific services. Therefore, the TSF will not solicit

8402 (i.e. try to obtain from other entities) any information that might be used to compromise  
8403 unobservability.

#### 8404 **I.6.4.2 Operations**

##### 8405 **I.6.4.2.1 Assignment**

8406 In FPR\_UNO.3.1, the PP/ST author **should** identify the list of services which are subject to the  
8407 unobservability requirement.

EXAMPLE

"the accessing of job descriptions".

8408 In FPR\_UNO.3.1, the PP/ST author **should** identify the list of subjects from which privacy related  
8409 information **should** be protected when the specified services are provided.

8410 In FPR\_UNO.3.1, the PP/ST author **should** specify the privacy related information that will be  
8411 protected from the specified subjects.

EXAMPLE

Examples include the identity of the subject that used a service and the quantity of a service that has been used such as memory resource utilization.

#### 8412 **I.6.5 FPR\_UNO.4 Authorized user observability**

##### 8413 **I.6.5.1 User application notes**

8414 This component is used to require that there will be one or more authorized users with the  
8415 rights to view the resource utilization. Without this component, this review is allowed, but not  
8416 mandated.

#### 8417 **I.6.5.2 Operations**

##### 8418 **I.6.5.2.1 Assignment**

8419 In FPR\_UNO.4.1, the PP/ST author **should** specify the set of authorized users for which the TSF  
8420 must provide the capability to observe the resource utilization. A set of authorized users, for  
8421 example, **could** be a group of authorized users which **can** operate under the same role or **can** all  
8422 use the same process(es).

8423 In FPR\_UNO.4.1, the PP/ST author **should** specify the set of resources and/or services that the  
8424 authorized user must be able to observe.

## Annex J (normative)

### Class FPT: Protection of the TSF- application notes

#### J.1 General information

This class contains families of functional requirements that relate to the integrity and management of the mechanisms that constitute the TSF and to the integrity of TSF data. In some sense, families in this class **may** appear to duplicate components in the FDP: User data protection class; they **may** even be implemented using the same mechanisms. However, FDP: User data protection focuses on user data protection, while FPT: Protection of the TSF focuses on TSF data protection. In fact, components from the FPT: Protection of the TSF class are necessary to provide requirements that the SFPs in the TOE cannot be tampered with or bypassed.

From the point of view of this class, regarding to the TSF there are three significant elements:

- a) The TSF's implementation, which executes and implements the mechanisms that enforce the SFRs.
- b) The TSF's data, which are the administrative databases that guide the enforcement of the SFRs.
- c) The external entities that the TSF **may** interact with in order to enforce the SFRs.

All of the families in the FPT: Protection of the TSF class **can** be related to these areas, and fall into the following groupings:

- a) TSF physical protection (FPT\_PHP), which provides an authorized user with the ability to detect external attacks on the parts of the TOE that comprise the TSF.
- b) Testing of external entities (FPT\_TEE) and TSF self-test (FPT\_TST), which provide an authorized user with the ability to verify the correct operation of the external entities interacting with the TSF to enforce the SFRs, and the integrity of the TSF data and TSF itself.
- c) Trusted recovery (FPT\_RCV), Fail secure (FPT\_FLS), and Internal TOE TSF data replication consistency (FPT\_TRC), which address the behaviour of the TSF when failure occurs and immediately after.
- d) Availability of exported TSF data (FPT\_ITA), Confidentiality of exported TSF data (FPT\_ITC), Integrity of exported TSF data (FPT\_ITI), which address the protection and availability of TSF data between the TSF and another trusted IT product.
- e) Internal TOE TSF data transfer (FPT\_ITT), which addresses protection of TSF data when it is transmitted between physically-separated parts of the TOE.
- f) Replay detection (FPT\_RPL), which addresses the replay of various types of information and/or operations.
- g) State synchrony protocol (FPT\_SSP), which addresses the synchronization of states, based upon TSF data, between different parts of a distributed TSF.
- h) Time stamps (FPT\_STM), which addresses reliable timing.
- i) Inter-TSF TSF data consistency (FPT\_TDC), which addresses the consistency of TSF data shared between the TSF and another trusted IT product.

#### J.2 User notes



8468 **J.3 FPT\_EMS TOE emanation**8469 **J.3.1 User notes**

8470 This family defines the requirements for the TSF to be able to prevent attacks against secret  
 8471 data stored in and used by the TOE where the attack is based on external observable physical  
 8472 phenomena of the TOE.

**EXAMPLE**

Examples of such attacks are evaluation of TOE's electromagnetic radiation, simple power analysis (SPA), differential power analysis (DPA), timing attacks, etc..

8473 FPT\_EMS.1.1 Limit of Emissions requires to not emit intelligible emissions enabling access to  
 8474 TSF data or user data.

8475 FPT\_EMS.1.2 Interface Emanation requires to not emit interface emanation enabling access to  
 8476 TSF data or user data.

8477 **J.3.2 FPT\_EMS.1 TOE emanation**8478 **J.3.3 User application notes**8479 **J.3.3.1 Operations**8480 **J.3.3.1.1 Assignment**8481 **J.4 Fail secure (FPT\_FLS)**8482 **J.4.1 User notes**

8483 The requirements of this family ensure that the TOE will always enforce its SFRs in the event of  
 8484 certain types of failures in the TSF.

8485 **J.4.2 FPT\_FLS.1 Failure with preservation of secure state**8486 **J.4.2.1 User application notes**

8487 The term “secure state” refers to a state in which the TSF data are consistent and the TSF  
 8488 continues correct enforcement of the SFRs.

8489 Although it is desirable to audit situations in which failure with preservation of secure state  
 8490 occurs, it is not possible in all situations. The PP/ST author **should** specify those situations in  
 8491 which audit is desired and feasible.

8492 Failures in the TSF **may** include “hard” failures, which indicate an equipment malfunction and  
 8493 which **may** require maintenance, service, or repair of the TSF. Failures in the TSF **may** also  
 8494 include recoverable “soft” failures, which **may** only require initialization or resetting of the TSF.

8495 **J.4.2.2 Operations**8496 **J.4.2.2.1 Assignment**

8497 In FPT\_FLS.1.1, the PP/ST author **should** list the types of failures in the TSF for which the TSF  
 8498 **should** “fail secure,” that is, **should** preserve a secure state and continue to correctly enforce the  
 8499 SFRs.

8500 **J.5 Fail secure (FPT\_INI)**8501 **J.5.1 User notes**8502 **J.5.2 FPT\_INI.1 XXX**8503 **J.5.3 User application notes**8504 **J.5.3.1 Operations**8505 **J.5.3.1.1 Assignment**



8506

8507 **J.6 Availability of exported TSF data (FPT\_ITA)**8508 **J.6.1 User notes**

8509 This family defines the rules for the prevention of loss of availability of TSF data moving  
 8510 between the TSF and another trusted IT product. This data **could** be TSF critical data such as  
 8511 passwords, keys, audit data, or TSF executable code.

8512 This family is used in a distributed context where the TSF is providing TSF data to another  
 8513 trusted IT product. The TSF **can** only take the measures at its site and cannot be held  
 8514 responsible for the TSF at the other trusted IT product.

8515 If there are different availability metrics for different types of TSF data, then this component  
 8516 **should** be iterated for each unique pairing of metrics and types of TSF data.

8517 **J.6.2 FPT\_ITA.1 Inter-TSF availability within a defined availability metric**8518 **J.6.2.1 Operations**8519 **J.6.2.1.1 Assignment**

8520 In FPT\_ITA.1.1, the PP/ST author **should** specify the types of TSF data that are subject to the  
 8521 availability metric.

8522 In FPT\_ITA.1.1, the PP/ST **should** specify the availability metric for the applicable TSF data.

8523 In FPT\_ITA.1.1, the PP/ST author **should** specify the conditions under which availability must be  
 8524 ensured.

**EXAMPLE**

There must be a connection between the TOE and another trusted IT product.

8525 **J.7 Confidentiality of exported TSF data (FPT\_ITC)**8526 **J.7.1 User notes**

8527 This family defines the rules for the protection from unauthorized disclosure of TSF data  
 8528 moving between the TSF and another trusted IT product.

**EXAMPLE**

Examples of this data are TSF critical data such as passwords, keys, audit data, or TSF executable code.

8529 This family is used in a distributed context where the TSF is providing TSF data to another  
 8530 trusted IT product. The TSF **can** only take the measures at its site and cannot be held  
 8531 responsible for the behaviour of the other trusted IT product.

8532 **J.7.2 FPT\_ITC.1 Inter-TSF confidentiality during transmission**8533 **J.7.2.1 Evaluator notes**

8534 Confidentiality of TSF Data during transmission is necessary to protect such information from  
 8535 disclosure.

**EXAMPLE**

Some possible implementations that **could** provide confidentiality include the use of cryptographic algorithms as well as spread spectrum techniques.

8536 **J.8 Integrity of exported TSF data (FPT\_ITI)**8537 **J.8.1 User notes**

8538 This family defines the rules for the protection, from unauthorized modification, of TSF data  
 8539 during transmission between the TSF and another trusted IT product.

EXAMPLE

Examples of this data are TSF critical data such as passwords, keys, audit data, or TSF executable code.

8540 This family is used in a distributed context where the TSF is exchanging TSF data with another  
 8541 trusted IT product. Note that a requirement that addresses modification, detection, or recovery  
 8542 at another trusted IT product cannot be specified, as the mechanisms that another trusted IT  
 8543 product will use to protect its data cannot be determined in advance. For this reason, these  
 8544 requirements are expressed in terms of the “TSF providing a capability” which another trusted  
 8545 IT product **can** use.

## 8546 **J.8.2 FPT\_ITI.1 Inter-TSF detection of modification**

### 8547 **J.8.2.1 User application notes**

8548 This component **should** be used in situations where it is sufficient to detect when data have  
 8549 been modified. An example of such a situation is one in which another trusted IT product **can**  
 8550 request the TOE's TSF to retransmit data when modification has been detected or respond to  
 8551 such types of request.

8552 The desired strength of modification detection is based upon a specified modification metric  
 8553 that is a function of the algorithm used, which **may** range from a weak checksum and parity  
 8554 mechanisms that **may** fail to detect multiple bit changes, to more complicated cryptographic  
 8555 checksum approaches.

### 8556 **J.8.2.2 Operations**

#### 8557 **J.8.2.2.1 Assignment**

8558 In FPT\_ITI.1.1, the PP/ST **should** specify the modification metric that the detection mechanism  
 8559 must satisfy. This modification metric **shall** specify the desired strength of the modification  
 8560 detection.

8561 In FPT\_ITI.1.2, the PP/ST **should** specify the actions to be taken if a modification of TSF data has  
 8562 been detected. An example of an action is: “ignore the TSF data and request the originating  
 8563 trusted product to send the TSF data again”.

## 8564 **J.8.3 FPT\_ITI.2 Inter-TSF detection and correction of modification**

### 8565 **J.8.3.1 User application notes**

8566 This component **should** be used in situations where it is necessary to detect or correct  
 8567 modifications of TSF critical data.

8568 The desired strength of modification detection is based upon a specified modification metric  
 8569 that is a function of the algorithm used, which **may** range from a checksum and parity  
 8570 mechanisms that **may** fail to detect multiple bit changes, to more complicated cryptographic  
 8571 checksum approaches. The metric that needs to be defined **can** either refer to the attacks it will  
 8572 resist or to mechanisms that are well known in the public literature.

EXAMPLE

Attack reference: “only 1 in a 1000 random messages will be accepted”.

Well known mechanism: “the strength must be conformant to the strength offered by Secure Hash Algorithm”.

8573  
 8574 The approach taken to correct modification might be done through some form of error  
 8575 correcting checksum.

### 8576 **J.8.3.2 Evaluator notes**

8577 Some possible means of satisfying this requirement involves the use of cryptographic functions  
 8578 or some form of checksum.

**J.8.3.3 Operations****J.8.3.3.1 Assignment**

In FPT\_ITI.2.1, the PP/ST **should** specify the modification metric that the detection mechanism must satisfy. This modification metric **shall** specify the desired strength of the modification detection.

In FPT\_ITI.2.2, the PP/ST **should** specify the actions to be taken if a modification of TSF data has been detected.

**EXAMPLE**

An example of an action is: "ignore the TSF data and request the originating trusted product to send the TSF data again".

In FPT\_ITI.2.3, the PP/ST author **should** define the types of modification from which the TSF **should** be capable of recovering.

**J.9 Internal TOE TSF data transfer (FPT\_ITT)****J.9.1 User notes**

This family provides requirements that address protection of TSF data when it is transferred between separate parts of a TOE across an internal channel.

The determination of the degree of separation (i.e., physical, or logical) that would make application of this family useful depends on the intended environment of use. In a hostile environment, there **may** be risks arising from transfers between parts of the TOE separated by only a system bus or an inter-process communications channel. In more benign environments, the transfers **may** be across more traditional network media.

**J.9.2 Evaluator notes**

One practical mechanism available to a TSF to provide this protection is cryptographically-based.

**J.9.3 FPT\_ITT.1 Basic internal TSF data transfer protection****J.9.3.1 Operations****J.9.3.1.1 Selection**

In FPT\_ITT.1.1, the PP/ST author **should** specify the desired type of protection to be provided from the choices: disclosure, modification.

**J.9.4 FPT\_ITT.2 TSF data transfer separation****J.9.4.1 User application notes**

One of the ways to achieve separation of TSF data based on SFP-relevant attributes is through the use of separate logical or physical channels.

**J.9.4.2 Operations****J.9.4.2.1 Selection**

In FPT\_ITT.2.1, the PP/ST author **should** specify the desired type of protection to be provided from the choices: disclosure, modification.

**J.9.5 FPT\_ITT.3 TSF data integrity monitoring****J.9.5.1 Operations****J.9.5.1.1 Selection**

8617 In FPT\_ITT.3.1, the PP/ST author **should** specify the desired type of modification that the TSF  
 8618 **shall** be able to detect. The PP/ST author **should** select from: modification of data, substitution  
 8619 of data, re-ordering of data, deletion of data, or any other integrity errors.

#### 8620 **J.9.5.1.2 Assignment**

8621 In FPT\_ITT.3.1, if the PP/ST author chooses the latter selection noted in the preceding  
 8622 paragraph, then the author **should** also specify what those other integrity errors are that the  
 8623 TSF **should** be capable of detecting.

8624 In FPT\_ITT.3.2, the PP/ST author **should** specify the action to be taken when an integrity error  
 8625 is identified.

### 8626 **J.10 TSF physical protection (FPT\_PHP)**

#### 8627 **J.10.1 User notes**

8628 TSF physical protection components refer to restrictions on unauthorized physical access to the  
 8629 TSF, and to the deterrence of, and resistance to, unauthorized physical modification, or  
 8630 substitution of the TSF.

8631 The requirements in this family ensure that the TSF is protected from physical tampering and  
 8632 interference. Satisfying the requirements of these components results in the TSF being  
 8633 packaged and used in such a manner that physical tampering is detectable, or resistance to  
 8634 physical tampering is measurable based on defined work factors. Without these components,  
 8635 the protection functions of a TSF lose their effectiveness in environments where physical  
 8636 damage cannot be prevented. This component also provides requirements regarding how the  
 8637 TSF must respond to physical tampering attempts.

##### EXAMPLE

Examples of physical tampering scenarios include mechanical attack, radiation, changing the temperature.

8638 It is acceptable for the functions that are available to an authorized user for detecting physical  
 8639 tampering to be available only in an off-line or maintenance mode. Controls **should** be in place  
 8640 to limit access during such modes to authorized users. As the TSF **may** not be “operational”  
 8641 during those modes, it **may** not be able to provide normal enforcement for authorized user  
 8642 access. The physical implementation of a TOE might consist of several structures. This set of  
 8643 “elements” as a whole must protect (protect, notify and resist) the TSF from physical tampering.  
 8644 This does not mean that all devices must provide these features, but the complete physical  
 8645 construct as a whole **should**.

##### EXAMPLE

Examples of structures include an outer shielding, cards, and chips.

8646 Although there is only minimal auditing associating with these components, this is solely  
 8647 because there is the potential that the detection and alarm mechanisms **may** be implemented  
 8648 completely in hardware, below the level of interaction with an audit subsystem. Nevertheless, a  
 8649 PP/ST author **may** determine that for a particular anticipated threat environment, there is a  
 8650 need to audit physical tampering. If this is the case, the PP/ST author **should** include  
 8651 appropriate requirements in the list of audit events.

8652 NOTE inclusion of these requirements **may** have implications on the hardware design and its interface to the  
 8653 software.

##### EXAMPLE

Examples of a hardware-based detection system is one based on breaking a circuit and lighting a light emitting diode (LED) if the circuit is broken when a button is pressed by the authorized user.

8654

#### 8655 **J.10.2 FPT\_PHP.1 Passive detection of physical attack**

##### 8656 **J.10.2.1 User application notes**

FPT\_PHP.1 Passive detection of physical attack **should** be used when threats from unauthorized physical tampering with parts of the TOE are not countered by procedural methods. It addresses the threat of undetected physical tampering with the TSF. Typically, an authorized user would be given the function to verify whether tampering took place. As written, this component simply provides a TSF capability to detect tampering. Specification of management functions in FMT\_LIM.1 **should** be considered to specify who **can** make use of that capability, and how they **can** make use of that capability. If this is done by non-IT mechanisms such as physical inspection, management functions are not required.

### **J.10.3 FPT\_PHP.2 Notification of physical attack**

#### **J.10.3.1 User application notes**

FPT\_PHP.2 Notification of physical attack **should** be used when threats from unauthorized physical tampering with parts of the TOE are not countered by procedural methods, and it is required that designated individuals be notified of physical tampering. It addresses the threat that physical tampering with TSF elements, although detected, **may** not be noticed. Specification of management functions in FMT\_MOF.1 Management of security functions behaviour **should** be considered to specify who **can** make use of that capability, and how they **can** make use of that capability.

#### **J.10.3.2 Operations**

##### **J.10.3.2.1 Assignment**

In FPT\_PHP.2.3, the PP/ST author **should** provide a list of TSF devices/elements for which active detection of physical tampering is required.

In FPT\_PHP.2.3, the PP/ST author **should** designate a user or role that is to be notified when tampering is detected. The type of user or role **may** vary depending on the particular security administration component (from the FMT\_LIM.1 family) included in the PP/ST.

### **J.10.4 FPT\_PHP.3 Resistance to physical attack**

#### **J.10.4.1 User application notes**

For some forms of tampering, it is necessary that the TSF not only detects the tampering, but actually resists it or delays the attacker.

This component **should** be used when TSF devices and TSF elements are expected to operate in an environment where a physical tampering of the internals of a TSF device or TSF element itself is a threat.

##### **EXAMPLE**

Physical tampering includes observation, analysis, or modification.

#### **J.10.4.2 Operations**

##### **J.10.4.2.1 Assignment**

In FPT\_PHP.3.1, the PP/ST author **should** specify tampering scenarios to a list of TSF devices/elements for which the TSF **should** resist physical tampering. This list **may** be applied to a defined subset of the TSF physical devices and elements based on considerations such as technology limitations and relative physical exposure of the device. Such sub setting **should** be clearly defined and justified. Furthermore, the TSF **should** automatically respond to physical tampering. The automatic response **should** be such that the policy of the device is preserved.

##### **EXAMPLE**

An example of policy protection:

with a confidentiality policy, it would be acceptable to physically disable the device so that the protected information **may** not be retrieved.

8697 In FPT\_PHP.3.1, the PP/ST author **should** specify the list of TSF devices/elements for which the  
 8698 TSF **should** resist physical tampering in the scenarios that have been identified.

## 8699 **J.11 Trusted recovery (FPT\_RCV)**

### 8700 **J.11.1 User notes**

8701 The requirements of this family ensure that the TSF **can** determine that the TOE is started-up  
 8702 without protection compromise and **can** recover without protection compromise after  
 8703 discontinuity of operations. This family is important because the start-up state of the TSF  
 8704 determines the protection of subsequent states.

8705 Recovery components reconstruct the TSF secure states, or prevent transitions to insecure  
 8706 states, as a direct response to occurrences of expected failures, discontinuity of operation or  
 8707 start-up.

#### EXAMPLE

Failures that must be generally anticipated include the following:

- a) Unmaskable action failures that always result in a system crash (such as persistent inconsistency of critical system tables, uncontrolled transfers within the TSF code caused by transient failures of hardware or firmware, power failures, processor failures, communication failures).
- b) Media failures causing part or all of the media representing the TSF objects to become inaccessible or corrupt (such as parity errors, disk head crash, persistent read/write failure caused by misaligned disk heads, worn-out magnetic coating, dust on the disk surface, loss of Internet connection).
- c) Discontinuity of operation caused by erroneous administrative action or lack of timely administrative action (such as unexpected shutdowns by turning off power, ignoring the exhaustion of critical resources, inadequate installed configuration).

8708

8709 **NOTE** Recovery may be from either a complete or partial failure scenario. Although a complete failure might  
 8710 occur in a monolithic operating system, it is less likely to occur in a distributed environment. In such environments,  
 8711 subsystems may fail, but other portions remain operational. Further, critical components may be redundant (disk  
 8712 mirroring, alternative routes), and checkpoints may be available. Thus, recovery is expressed in terms of recovery to  
 8713 a secure state.

8714 There are different interactions between Trusted recovery (FPT\_RCV) and TSF self-test  
 8715 (FPT\_TST) components to be considered when selecting Trusted recovery (FPT\_RCV):

- 8716 a) The need for trusted recovery **may** be indicated through the results of TSF self-  
 8717 testing, where the results of the self-tests indicate that the TSF is in an insecure  
 8718 state and return to a secure state or entrance in maintenance mode is required.
- 8719 b) A failure, as discussed above, **may** be identified by an administrator. Either the  
 8720 administrator **may** perform the actions to return the TOE to a secure state and then  
 8721 invoke TSF self-tests to confirm that the secure state has been achieved. Or, the TSF  
 8722 self-tests **may** be invoked to complete the recovery process.
- 8723 c) A combination of a. and b. above, where the need for trusted recovery is indicated  
 8724 through the results of TSF self-testing, the administrator performs the actions to  
 8725 return the TOE to a secure state and then invokes TSF self-tests to confirm that the  
 8726 secure state has been achieved.
- 8727 d) Self-tests detect a failure/service discontinuity, then either automated recovery or  
 8728 entrance to a maintenance mode.

8729 This family identifies a maintenance mode. In this maintenance mode, normal operation might  
 8730 be impossible or severely restricted, as otherwise insecure situations might occur. Typically,  
 8731 only authorized users **should** be allowed access to this mode but the real details of who **can**  
 8732 access this mode is a function of FMT: Security management. If FMT: Security management does  
 8733 not put any controls on who **can** access this mode, then it **may** be acceptable to allow any user  
 8734 to restore the system if the TOE enters such a state. However, in practice, this is probably not



8735 desirable as the user restoring the system has an opportunity to configure the TOE in such a  
8736 way as to violate the SFRs.

8737 Mechanisms designed to detect exceptional conditions during operation fall under TSF self-test  
8738 (FPT\_TST), Fail secure (FPT\_FLS), and other areas that address the concept of “Software Safety.”  
8739 It is likely that the use of one of these families will be required to support the adoption of  
8740 Trusted recovery (FPT\_RCV). This is to ensure that the TOE will be able to detect when recovery  
8741 is required.

8742 Throughout this family, the phrase “secure state” is used. This refers to some state in which the  
8743 TOE has consistent TSF data and a TSF that **can** correctly enforce the policy. This state **may** be  
8744 the initial “boot” of a clean system, or it might be some checkpointed state.

8745 Following recovery, it **may** be necessary to confirm that the secure state has been achieved  
8746 through self-testing of the TSF. However, if the recovery is performed in a manner such that  
8747 only a secure state **can** be achieved, else recovery fails, then the dependency to the FPT\_TST.1  
8748 TSF self-testing component **may** be argued away.

### 8749 **J.11.2 FPT\_RCV.1 Manual recovery**

#### 8750 **J.11.2.1 User application notes**

8751 In the hierarchy of the trusted recovery family, recovery that requires only manual intervention  
8752 is the least desirable, for it precludes the use of the system in an unattended fashion.

8753 This component is intended for use in TOEs that do not require unattended recovery to a secure  
8754 state. The requirements of this component reduce the threat of protection compromise  
8755 resulting from an attended TOE returning to an insecure state after recovery from a failure or  
8756 other discontinuity.

#### 8757 **J.11.2.2 Evaluator notes**

8758 It is acceptable for the functions that are available to an authorized user for trusted recovery to  
8759 be available only in a maintenance mode. Controls **should** be in place to limit access during  
8760 maintenance to authorized users.

#### 8761 **J.11.2.3 Operations**

##### 8762 **J.11.2.3.1 Assignment**

8763 In FPT\_RCV.1.1, the PP/ST author **should** specify the list of failures or service discontinuities  
8764 following which the TOE will enter a maintenance mode.

##### EXAMPLE

power failure, audit storage exhaustion, any failure or discontinuity.

8765

### 8766 **J.11.3 FPT\_RCV.2 Automated recovery**

#### 8767 **J.11.3.1 User application notes**

8768 Automated recovery is considered to be more useful than manual recovery, as it allows the  
8769 machine to operate in an unattended fashion.

8770 The component FPT\_RCV.2 Automated recovery extends the feature coverage of FPT\_RCV.1  
8771 Manual recovery by requiring that there be at least one automated method of recovery from  
8772 failure or service discontinuity. It addresses the threat of protection compromise resulting from  
8773 an unattended TOE returning to an insecure state after recovery from a failure or other  
8774 discontinuity.

#### 8775 **J.11.3.2 Evaluator notes**

8776 It is acceptable for the functions that are available to an authorized user for trusted recovery to  
 8777 be available only in a maintenance mode. Controls **should** be in place to limit access during  
 8778 maintenance to authorized users.

8779 For FPT\_RCV.2.1, it is the responsibility of the developer of the TSF to determine the set of  
 8780 recoverable failures and service discontinuities.

8781 It is assumed that the robustness of the automated recovery mechanisms will be verified.

### 8782 **J.11.3.3 Operations**

#### 8783 **J.11.3.3.1 Assignment**

8784 In FPT\_RCV.2.1, the PP/ST author **should** specify the list of failures or service discontinuities  
 8785 following which the TOE will need to enter a maintenance mode.

#### EXAMPLE

power failure, audit storage exhaustion.

8786 In FPT\_RCV.2.2, the PP/ST author **should** specify the list of failures or other discontinuities for  
 8787 which automated recovery must be possible.

### 8788 **J.11.4 FPT\_RCV.3 Automated recovery without undue loss**

#### 8789 **J.11.4.1 User application notes**

8790 Automated recovery is considered to be more useful than manual recovery, but it runs the risk  
 8791 of losing a substantial number of objects. Preventing undue loss of objects provides additional  
 8792 utility to the recovery effort.

8793 The component FPT\_RCV.3 Automated recovery without undue loss extends the feature  
 8794 coverage of FPT\_RCV.2 Automated recovery by requiring that there not be undue loss of TSF  
 8795 data or objects under the control of the TSF. At FPT\_RCV.2 Automated recovery, the automated  
 8796 recovery mechanisms **could** conceivably recover by deleting all objects and returning the TSF to  
 8797 a known secure state. This type of drastic automated recovery is precluded in FPT\_RCV.3  
 8798 Automated recovery without undue loss.

8799 This component addresses the threat of protection compromise resulting from an unattended  
 8800 TOE returning to an insecure state after recovery from a failure or other discontinuity with a  
 8801 large loss of TSF data or objects under the control of the TSF.

#### 8802 **J.11.4.2 Evaluator notes**

8803 It is acceptable for the functions that are available to an authorized user for trusted recovery to  
 8804 be available only in a maintenance mode. Controls **should** be in place to limit access during  
 8805 maintenance to authorized users.

8806 It is assumed that the evaluators will verify the robustness of the automated recovery  
 8807 mechanisms.

### 8808 **J.11.4.3 Operations**

#### 8809 **J.11.4.3.1 Assignment**

8810 In FPT\_RCV.3.1, the PP/ST author **should** specify the list of failures or service discontinuities  
 8811 following which the TOE will need to enter a maintenance mode.

#### EXAMPLE

power failure, audit storage exhaustion.

8812 In FPT\_RCV.3.2, the PP/ST author **should** specify the list of failures or other discontinuities for  
 8813 which automated recovery must be possible.

8814 In FPT\_RCV.3.3, the PP/ST author **should** provide a quantification for the amount of loss of TSF  
 8815 data or objects that is acceptable.



## 8816 **J.11.5 FPT\_RCV.4 Function recovery**

### 8817 **J.11.5.1 User application notes**

8818 Function recovery requires that if there **should** be some failure in the TSF, that certain functions  
8819 in the TSF **should** either complete successfully or recover to a secure state.

### 8820 **J.11.5.2 Operations**

#### 8821 **J.11.5.2.1 Assignment**

8822 In FPT\_RCV.4.1, the PP/ST author **should** specify a list the functions and failure scenarios. In the  
8823 event that any of the identified failure scenarios happen, the functions that have been specified  
8824 must either complete successfully or recover to a consistent and secure state.

## 8825 **J.12 Replay detection (FPT\_RPL)**

### 8826 **J.12.1 User notes**

8827 This family addresses detection of replay for various types of entities and subsequent actions to  
8828 correct.

### 8829 **J.12.2 FPT\_RPL.1 Replay detection**

#### 8830 **J.12.2.1 User application notes**

8831 The entities included here are those that can be involved in replay detection.

#### EXAMPLE

Messages, service requests, service responses, or sessions.

#### 8832 **J.12.2.2 Operations**

##### 8833 **J.12.2.2.1 Assignment**

8834 In FPT\_RPL.1.1, the PP/ST author **should** provide a list of identified entities for which detection  
8835 of replay **should** be possible.

#### EXAMPLE

Messages, service requests, service responses, and user sessions.

8836 In FPT\_RPL.1.2, the PP/ST author **should** specify the list of actions to be taken by the TSF when  
8837 replay is detected. The potential set of actions that **can** be taken includes: ignoring the replayed  
8838 entity, requesting confirmation of the entity from the identified source, and terminating the  
8839 subject from which the re-played entity originated.

## 8840 **J.13 State synchrony protocol (FPT\_SSP)**

### 8841 **J.13.1 User notes**

8842 Distributed TOEs **may** give rise to greater complexity than monolithic TOEs through the  
8843 potential for differences in state between parts of the TOE, and through delays in  
8844 communication. In most cases, synchronization of state between distributed functions involves  
8845 an exchange protocol, not a simple action. When malice exists in the distributed environment of  
8846 these protocols, more complex defensive protocols are required.

8847 State synchrony protocol (FPT\_SSP) establishes the requirement for certain critical functions of  
8848 the TSF to use a trusted protocol. State synchrony protocol (FPT\_SSP) ensures that two  
8849 distributed parts of the TOE, such as hosts, have synchronized their states after a security-  
8850 relevant action.

8851 Some states **may** never be synchronized, or the transaction cost **may** be too high for practical  
8852 use.

#### EXAMPLE

encryption key revocation is an example, where knowing the state after the revocation action is initiated **can** never be known. Either the action was taken and acknowledgment cannot be sent, or the message was ignored by hostile communication partners and the revocation never occurred.

8853

8854 Indeterminacy is unique to distributed TOEs. Indeterminacy and state synchrony are related,  
8855 and the same solution **may** apply. It is futile to design for indeterminate states; the PP/ST  
8856 author **should** express other requirements in such cases.

**EXAMPLE**

raise an alarm, audit the event.

8857

**8858 J.13.2 FPT\_SSP.1 Simple trusted acknowledgement****8859 J.13.2.1 User application notes**

8860 In this component, the TSF must supply an acknowledgement to another part of the TSF when  
8861 requested. This acknowledgement **should** indicate that one part of a distributed TOE  
8862 successfully received an unmodified transmission from a different part of the distributed TOE.

**8863 J.13.3 FPT\_SSP.2 Mutual trusted acknowledgement****8864 J.13.3.1 User application notes**

8865 In this component, in addition to the TSF being able to provide an acknowledgement for the  
8866 receipt of a data transmission, the TSF must comply with a request from another part of the TSF  
8867 for an acknowledgement to the acknowledgement.

**EXAMPLE**

The local TSF transmits some data to a remote part of the TSF. The remote part of the TSF acknowledges the successful receipt of the data and requests that the sending TSF confirm that it receives the acknowledgement. This mechanism provides additional confidence that both parts of the TSF involved in the data transmission know that the transmission completed successfully.

**8868 J.14 Time stamps (FPT\_STM)****8869 J.14.1 User notes**

8870 This family addresses requirements for a reliable time stamp function within a TOE.

8871 It is the responsibility of the PP/ST author to clarify the meaning of the phrase “reliable time  
8872 stamp”, and to indicate where the responsibility lies in determining the acceptance of trust.

**8873 J.14.2 FPT\_STM.1 Reliable time stamps****8874 J.14.2.1 User application notes**

8875 Some possible uses of this component include providing reliable time stamps for the purposes  
8876 of audit as well as for security attribute expiration.

**8877 J.15 Inter-TSF TSF data consistency (FPT\_TDC)****8878 J.15.1 User notes**

8879 In a distributed or composite environment, a TOE **may** need to exchange TSF data with another  
8880 trusted IT Product.

**EXAMPLE**

the SFP-attributes associated with data, audit information, identification information.

8881 This family defines the requirements for sharing and consistent interpretation of these  
8882 attributes between the TSF of the TOE and that of a different trusted IT Product.

The components in this family are intended to provide requirements for automated support for TSF data consistency when such data is transmitted between the TSF of the TOE and another trusted IT Product. It is also possible that wholly procedural means **could** be used to produce security attribute consistency, but they are not provided for here.

This family is different from FDP\_ETC and FDP\_ITC, as those two families are concerned only with resolving the security attributes between the TSF and its import/export medium.

If the integrity of the TSF data is of concern, requirements **should** be chosen from the Integrity of exported TSF data (FPT\_ITI) family. These components specify requirements for the TSF to be able to detect or detect and correct modifications to TSF data in transit.

## **J.15.2 FPT\_TDC.1 Inter-TSF basic TSF data consistency**

### **J.15.2.1 User application notes**

The TSF is responsible for maintaining the consistency of TSF data used by or associated with the specified function and that are common between two or more trusted systems.

#### **EXAMPLE**

The TSF data of two different systems **may** have different conventions internally. For the TSF data to be used properly (such as to afford the user data the same protection as within the TOE) by the receiving trusted IT product, the TOE and the other trusted IT product must use a pre-established protocol to exchange TSF data.

### **J.15.2.2 Operations**

#### **J.15.2.2.1 Assignment**

In FPT\_TDC.1.1, the PP/ST author **should** define the list of TSF data types, for which the TSF **shall** provide the capability to consistently interpret, when shared between the TSF and another trusted IT product.

In FPT\_TDC.1.2, the PP/ST **should** assign the list of interpretation rules to be applied by the TSF.

## **J.16 Testing of external entities (FPT\_TEE)**

### **J.16.1 User notes**

This family defines requirements for the testing of one or more external entities by the TSF. These external entities are not human users, and they **can** include combinations of software and/or hardware interacting with the TOE.

#### **EXAMPLE**

Examples of the types of tests that **may** be run are:

- a) tests for the presence of a firewall, and possibly whether it is correctly configured;
- b) tests of some of the properties of the operating system that an application TOE runs on;
- c) tests of some of the properties of the IC that a smart card OS TOE runs on (such as the random number generator).

Note The external entity **may** "lie" about the test results, either on purpose or because it is not working correctly.

These tests **can** be carried out either in some maintenance state, at start-up, on-line, or continuously. The actions to be taken by the TOE as the result of testing are defined also in this family.

### **J.16.2 Evaluator notes**

The tests of external entities **should** be sufficient to test all of the characteristics of them upon which the TSF relies.

### **J.16.3 FPT\_TEE.1 Testing of external entities**

8918 **J.16.3.1 User application notes**

8919 This component is not intended to be applied to human users.

8920 This component provides support for the periodic testing of properties related to external  
 8921 entities upon which the TSF's operation depends, by requiring the ability to periodically invoke  
 8922 testing functions.

8923 The PP/ST author **may** refine the requirement to state whether the function **should** be available  
 8924 in off-line, on-line or maintenance mode.

8925 **J.16.3.2 Evaluator notes**

8926 It is acceptable for the functions for periodic testing to be available only in an off-line or  
 8927 maintenance mode. Controls **should** be in place to limit access, during maintenance, to  
 8928 authorized users.

8929 **J.16.3.3 Operations**8930 **J.16.3.3.1 Selection**

8931 In FPT\_TEE.1.1, the PP/ST author **should** specify when the TSF will run the testing of external  
 8932 entities, during initial start-up, periodically during normal operation, at the request of an  
 8933 authorized user, or under other conditions. If the tests are run often, then the end users **should**  
 8934 have more confidence that the TOE is operating correctly than if the tests are run less  
 8935 frequently. However, this need for confidence that the TOE is operating correctly must be  
 8936 balanced with the potential impact on the availability of the TOE, as often times, the testing of  
 8937 external entities **may** delay the normal operation of a TOE.

8938 **J.16.3.3.2 Assignment**

8939 In FPT\_TEE.1.1, the PP/ST author **should** specify the properties of the external entities to be  
 8940 checked by the tests.

**EXAMPLE**

Examples of these properties **may** include configuration or availability properties of a directory server supporting some access control part of the TSF.

8941

8942 In FPT\_TEE.1.1, the PP/ST author **should**, if other conditions are selected, specify the frequency  
 8943 with which the testing of external entities will be run.

**EXAMPLE**

An example of this other frequency or condition **may** be to run the tests each time a user requests to initiate a session with the TOE. For instance, this **could** be the case of testing a directory server before its interaction with the TSF during the user authentication process.

8944

8945 In FPT\_TEE.1.2, the PP/ST author **should** specify what are the action(s) that the TSF **shall**  
 8946 perform when the testing fails.

**EXAMPLE**

Examples of these action(s), illustrated by a directory server instance, **may** include to connect to an alternative available server or otherwise to look for a backup server.

8947 **J.17 Internal TOE TSF data replication consistency (FPT\_TRC)**8948 **J.17.1 User notes**

8949 The requirements of this family are needed to ensure the consistency of TSF data when such  
 8950 data is replicated internal to the TOE. Such data **may** become inconsistent if an internal channel  
 8951 between parts of the TOE becomes inoperative. If the TOE is internally structured as a network

8952 of parts of the TOE, this **can** occur when parts become disabled, network connections are  
8953 broken, and so on.

8954 The method of ensuring consistency is not specified in this component. It **could** be attained  
8955 through a form of transaction logging (where appropriate transactions are “rolled back” to a  
8956 site upon reconnection); it **could** be updating the replicated data through a synchronization  
8957 protocol. If a particular protocol is necessary for a PP/ST, it **can** be specified through  
8958 refinement.

8959 It **may** be impossible to synchronize some states, or the cost of such synchronization **may** be too  
8960 high.

**EXAMPLE**

Examples of this situation are communication channel and encryption key revocations. Indeterminate states **may** also occur; if a specific behaviour is desired, it **should** be specified via refinement.

8961

## 8962 **J.17.2 FPT\_TRC.1 Internal TSF consistency**

### 8963 **J.17.2.1 Operations**

#### 8964 **J.17.2.1.1 Assignment**

8965 In FPT\_TRC.1.2, the PP/ST author **should** specify the list of functions dependent on TSF data  
8966 replication consistency.

## 8967 **J.18 TSF self-test (FPT\_TST)**

### 8968 **J.18.1 User notes**

8969 The family defines the requirements for the self-testing of the TSF with respect to some  
8970 expected correct operation.

**EXAMPLE**

Examples are interfaces to enforcement functions, and sample arithmetical operations on critical parts of the TOE.

8971

8972 These tests **can** be carried out at start-up, periodically, at the request of an authorized user, or  
8973 when other conditions are met. The actions to be taken by the TOE as the result of self-testing  
8974 are defined in other families.

8975 The requirements of this family are also needed to detect the corruption of TSF data and TSF  
8976 itself (i.e. TSF executable code or TSF hardware component) by various failures that do not  
8977 necessarily stop the TOE's operation (which would be handled by other families). These checks  
8978 must be performed because these failures **may** not necessarily be prevented. Such failures **can**  
8979 occur either because of unforeseen failure modes or associated oversights in the design of  
8980 hardware, firmware, or software, or because of malicious corruption of the TSF due to  
8981 inadequate logical and/or physical protection.

8982 In addition, use of this component may, with appropriate conditions, help to prevent  
8983 inappropriate or damaging TSF changes being applied to an operational TOE as the result of  
8984 maintenance activities.

8985 The term “correct operation of the TSF” refers primarily to the operation of the TSF and the  
8986 integrity of the TSF data.

### 8987 **J.18.2 FPT\_TST.1 TSF testing**

#### 8988 **J.18.2.1 User application notes**

8989 This component provides support for the testing of the critical functions of the TSF's operation  
8990 by requiring the ability to invoke testing functions and check the integrity of TSF data and  
8991 executable code.

8992 **J.18.2.2 Evaluator notes**

8993 It is acceptable for the functions that are available to the authorized user for periodic testing to  
 8994 be available only in an off-line or maintenance mode. Controls **should** be in place to limit access  
 8995 during these modes to authorized users.

8996 **J.18.2.3 Operations**8997 **J.18.2.3.1 Selection**

8998 In FPT\_TST.1.1, the PP/ST author **should** specify when the TSF will execute the TSF test; during  
 8999 initial start-up, periodically during normal operation, at the request of an authorized user, at  
 9000 other conditions. In the case of the latter option, the PP/ST author **should** also assign what  
 9001 those conditions are via the following assignment.

9002 In FPT\_TST.1.1, the PP/ST author **should** specify whether the self-tests are to be carried out to  
 9003 demonstrate the correct operation of the entire TSF, or of only specified parts of TSF.

9004 **J.18.2.3.2 Assignment**

9005 In FPT\_TST.1.1, the PP/ST author **should**, if selected, specify the conditions under which the  
 9006 self-test **should** take place.

9007 In FPT\_TST.1.1, the PP/ST author **should**, if selected, specify the list of parts of the TSF that will  
 9008 be subject to TSF self-testing.

9009 **J.18.2.3.3 Selection**

9010 In FPT\_TST.1., the PP/ST author **should** specify whether data integrity is to be verified for all  
 9011 TSF data, or only for selected data.

9012 **J.18.2.3.4 Assignment**

9013 In FPT\_TST.1., the PP/ST author **should**, if selected, specify the list of TSF data that will be  
 9014 verified for integrity.

9015 **J.18.2.3.5 Selection**

9016 In FPT\_TST.1., the PP/ST author **should** specify whether TSF integrity is to be verified for all  
 9017 TSF, or only for selected TSF.

9018 **J.18.2.3.6 Assignment**

9019 In FPT\_TST.1., the PP/ST author **should**, if selected, specify the list of TSF that will be verified  
 9020 for integrity.

## Annex K (normative)

### Class FRU: Resource utilization- application notes

#### K.1 General information

This class provides three families that support the availability of required resources such as processing capability and/or storage capacity. The family Fault Tolerance provides protection against unavailability of capabilities caused by failure of the TOE. The family Priority of Service ensures that the resources will be allocated to the more important or time-critical tasks and cannot be monopolized by lower priority tasks. The family Resource Allocation provides limits on the use of available resources, therefore preventing users from monopolizing the resources.

#### K.2 Fault tolerance (FRU\_FLT)

##### K.2.1 User notes

This family provides requirements for the availability of capabilities even in the case of failures.

###### EXAMPLE

Examples of such failures are power failure, hardware failure, or software error.

In case of these errors, if so specified, the TOE will maintain the specified capabilities.

###### EXAMPLE

The PP/ST author **could** specify that a TOE used in a nuclear plant will continue the operation of the shut-down procedure in the case of power-failure or communication-failure

Because the TOE **can** only continue its correct operation if the SFRs are enforced, there is a requirement that the system must remain in a secure state after a failure. This capability is provided by FPT\_FLS.1 Failure with preservation of secure state.

The mechanisms to provide fault tolerance **could** be active or passive. In case of an active mechanism, specific functions are in place that are activated in case the error occurs. For example, a fire alarm is an active mechanism: the TSF will detect the fire and **can** take action such as switching operation to a backup. In a passive scheme, the architecture of the TOE is capable of handling the error. For example, the use of a majority voting scheme with multiple processors is a passive solution; failure of one processor will not disrupt the operation of the TOE (although it needs to be detected to allow correction).

For this family, it does not matter whether the failure has been initiated accidentally (such as flooding or unplugging the wrong device) or intentionally (such as monopolizing).

##### K.2.2 FRU\_FLT.1 Degraded fault tolerance

###### K.2.2.1 User application notes

This component is intended to specify which capabilities the TOE will still provide after a failure of the system. Since it would be difficult to describe all specific failures, categories of failures **may** be specified.

###### EXAMPLE

Examples of general failures are flooding of the computer room, short term power interruption, breakdown of a CPU or host, software failure, or buffer overflow.

###### K.2.2.2 Operations



9056 **K.2.2.2.1 Assignment**

9057 In FRU\_FLT.1.1, the PP/ST author **should** specify the list of TOE capabilities the TOE will  
9058 maintain during and after a specified failure.

9059 In FRU\_FLT.1.1, the PP/ST author **should** specify the list of types of failures against which the  
9060 TOE has to be explicitly protected. If a failure in this list occurs, the TOE will be able to continue  
9061 its operation.

9062 **K.2.3 FRU\_FLT.2 Limited fault tolerance**

9063 **K.2.3.1 User application notes**

9064 This component is intended to specify against what type of failures the TOE must be resistant.  
9065 Since it would be difficult to describe all specific failures, categories of failures **may** be specified.

EXAMPLE

Examples of general failures are flooding of the computer room, short term power interruption, breakdown of a CPU or host, software failure, or overflow of buffer.

9066

9067 **K.2.3.2 Operations**

9068 **K.2.3.2.1 Assignment**

9069 In FRU\_FLT.2.1, the PP/ST author **should** specify the list of types of failures against which the  
9070 TOE has to be explicitly protected. If a failure in this list occurs, the TOE will be able to continue  
9071 its operation.

9072 **K.3 Priority of service (FRU\_PRS)**

9073 **K.3.1 User notes**

9074 The requirements of this family allow the TSF to control the use of resources under the control  
9075 of the TSF by users and subjects such that high priority activities under the control of the TSF  
9076 will always be accomplished without interference or delay due to low priority activities. In  
9077 other words, time critical tasks will not be delayed by tasks that are less time critical.

9078 This family **could** be applicable to several types of resources.

EXAMPLE

processing capacity, and communication channel capacity.

9079 The Priority of Service mechanism might be passive or active. In a passive Priority of Service  
9080 system, the system will select the task with the highest priority when given a choice between  
9081 two waiting applications. While using passive Priority of Service mechanisms, when a low  
9082 priority task is running, it cannot be interrupted by a high priority task. While using an active  
9083 Priority of Service mechanisms, lower priority tasks might be interrupted by new high priority  
9084 tasks.

9085 The audit requirement states that all reasons for rejection **should** be audited. It is left to the  
9086 developer to argue that an operation is not rejected but delayed.

9087 **K.3.2 FRU\_PRS.1 Limited priority of service**

9088 **K.3.2.1 User application notes**

9089 This component defines priorities for a subject, and the resources for which this priority will be  
9090 used. If some subject attempts to take action on a resource controlled by the Priority of Service  
9091 requirements, the access and/or time of access will be dependent on the subject's priority, the  
9092 priority of the currently acting subject, and the priority of the subjects still in the queue.

9093 **K.3.2.2 Operations**

9094 **K.3.2.2.1 Assignment**



9095 In FRU\_PRS.1.2, the PP/ST author **should** specify the list of controlled resources for which the  
 9096 TSF enforces priority of service

EXAMPLE

resources such as processes, disk space, memory, bandwidth.

### 9097 **K.3.3 FRU\_PRS.2 Full priority of service**

#### 9098 **K.3.3.1 User application notes**

9099 This component defines priorities for a subject. All shareable resources under the control of the  
 9100 TSF will be subjected to the Priority of Service mechanism. If some subject attempts to take  
 9101 action on a shareable TSF resource, the access and/or time of access will be dependent on the  
 9102 subject's priority, the priority of the currently acting subject, and the priority of the subjects still  
 9103 in the queue.

### 9104 **K.4 Resource allocation (FRU\_RSA)**

#### 9105 **K.4.1 User notes**

9106 The requirements of this family allow the TSF to control the use of resources under the control  
 9107 of the TSF by users and subjects such that unauthorized denial of service will not take place by  
 9108 means of monopolization of resources by other users or subjects.

9109 Resource allocation rules allow the creation of quotas or other means of defining limits on the  
 9110 amount of resource space or time that **may** be allocated on behalf of a specific user or subjects.

EXAMPLE

These rules may, for example:

- Provide for object quotas that constrain the number and/or size of objects a specific user may allocate;
- Control the allocation/deallocation of preassigned resource units where these units are under the control of the TSF.

9111 In general, these functions will be implemented through the use of attributes assigned to users  
 9112 and resources.

9113 The objective of these components is to ensure a certain amount of fairness among the users  
 9114 and subjects.

EXAMPLE

A single user **should** not allocate all the available space

9115 Since resource allocation often goes beyond the lifespan of a subject (i.e. files often exist longer  
 9116 than the applications that generated them), and multiple instantiations of subjects by the same  
 9117 user **should** not negatively affect other users too much, the components allow that the  
 9118 allocation limits are related to the users. In some situations, the resources are allocated by a  
 9119 subject.

EXAMPLE

Main memory or CPU cycles.

9120 In those instances, the components allow that the resource allocation be on the level of subjects.

9121 This family imposes requirements on resource allocation, not on the use of the resource itself.  
 9122 The audit requirements therefore, as stated, also apply to the allocation of the resource, not to  
 9123 the use of the resource.

#### 9124 **K.4.2 FRU\_RSA.1 Maximum quotas**

##### 9125 **K.4.2.1 User application notes**

9126 This component provides requirements for quota mechanisms that apply to only a specified set  
 9127 of the shareable resources in the TOE. The requirements allow the quotas to be associated with  
 9128 a user, possibly assigned to groups of users or subjects as applicable to the TOE.

#### 9129 **K.4.2.2 Operations**

##### 9130 **K.4.2.2.1 Assignment**

9131 In FRU\_RSA.1.1, the PP/ST author **should** specify the list of controlled resources for which  
 9132 maximum resource allocation limits are required.

###### EXAMPLE

processes, disk space, memory, bandwidth.

9133 If all resources under the control of the TSF need to be included, the words “all TSF resources”  
 9134 **may** be specified.

##### 9135 **K.4.2.2.2 Selection**

9136 In FRU\_RSA.1.1, the PP/ST author **should** select whether the maximum quotas apply to  
 9137 individual users, to a defined group of users, or subjects or any combination of these.

9138 In FRU\_RSA.1.1, the PP/ST author **should** select whether the maximum quotas are applicable to  
 9139 any given time (simultaneously), or over a specific time interval.

#### 9140 **K.4.3 FRU\_RSA.2 Minimum and maximum quotas**

##### 9141 **K.4.3.1 User application notes**

9142 This component provides requirements for quota mechanisms that apply to a specified set of  
 9143 the shareable resources in the TOE. The requirements allow the quotas to be associated with a  
 9144 user, or possibly assigned to groups of users as applicable to the TOE.

#### 9145 **K.4.3.2 Operations**

##### 9146 **K.4.3.2.1 Assignment**

9147 In FRU\_RSA.2.1, the PP/ST author **should** specify the controlled resources for which maximum  
 9148 and minimum resource allocation limits are required.

###### EXAMPLE

Processes, disk space, memory, bandwidth.

9149 If all resources under the control of the TSF need to be included, the words “all TSF resources”  
 9150 **can** be specified.

##### 9151 **K.4.3.2.2 Selection**

9152 In FRU\_RSA.2.1, the PP/ST author **should** select whether the maximum quotas apply to  
 9153 individual users, to a defined group of users, or subjects or any combination of these.

9154 In FRU\_RSA.2.1, the PP/ST author **should** select whether the maximum quotas are applicable to  
 9155 any given time (simultaneously), or over a specific time interval.

##### 9156 **K.4.3.2.3 Assignment**

9157 In FRU\_RSA.2.2, the PP/ST author **should** specify the controlled resources for which a minimum  
 9158 allocation limit needs to be set.

###### EXAMPLE

Processes, disk space, memory, bandwidth.

9159 If all resources under the control of the TSF need to be included the words “all TSF resources”  
 9160 **can** be specified.

##### 9161 **K.4.3.2.4 Selection**

- 9162 In FRU\_RSA.2.2, the PP/ST author **should** select whether the minimum quotas apply to  
9163 individual users, to a defined group of users, or subjects or any combination of these.
- 9164 In FRU\_RSA.2.2, the PP/ST author **should** select whether the minimum quotas are applicable to  
9165 any given time (simultaneously), or over a specific time interval.

## Annex L (normative)

### Class FTA: TOE access- application notes

#### L.1 General information

The establishment of a user's session typically consists of the creation of one or more subjects that perform operations in the TOE on behalf of the user. At the end of the session establishment procedure, provided the TOE access requirements are satisfied, the created subjects bear the attributes determined by the identification and authentication functions. This family specifies functional requirements for controlling the establishment of a user's session.

A user session is defined as the period starting at the time of the identification/authentication, or if more appropriate, the start of an interaction between the user and the system, up to the moment that all subjects (resources and attributes) related to that session have been deallocated.

#### L.2 Limitation on scope of selectable attributes (FTA\_LSA)

##### L.2.1 User notes

This family defines requirements that will limit the session security attributes a user **may** select, and the subjects to which a user **may** be bound, based on: the method of access; the location or port of access; and/or the time.

###### EXAMPLE

time-of-day, day-of-week.

This family provides the capability for a PP/ST author to specify requirements for the TSF to place limits on the domain of an authorized user's security attributes based on an environmental condition.

###### EXAMPLE

a user **may** be allowed to establish a "secret session" during normal business hours but outside those hours the same user **may** be constrained to only establishing "unclassified sessions".

The identification of relevant constraints on the domain of selectable attributes **may** be achieved through the use of the selection operation. These constraints **may** be applied on an attribute-by-attribute basis. When there exists a need to specify constraints on multiple attributes this component will have to be replicated for each attribute.

###### EXAMPLE

Examples of attributes that **could** be used to limit the session security attributes are:

The method of access can be used to specify in which type of environment the user will be operating (such as file transfer protocol, terminal, vtam).

The location of access can be used to constrain the domain of a user's selectable attributes based on a user's location or port of access. This capability is of particular use in environments where dial-up facilities or network facilities are available.

The time of access can be used to constrain the domain of a user's selectable attributes. For example, ranges may be based upon time-of-day, day-of-week, or calendar dates. This constraint provides some operational protection against user actions that **could** occur at a time where proper monitoring or where proper procedural measures may not be in place.

##### L.2.2 FTA\_LSA.1 Limitation on scope of selectable attributes

###### L.2.2.1 Operations

9195 **L.2.2.1.1 Assignment**

9196 In FTA\_LSA.1.1, the PP/ST author **should** specify the set of session security attributes that are to  
9197 be constrained.

EXAMPLE

Examples of these session security attributes are user clearance level, integrity level and roles.

9198 In FTA\_LSA.1.1, the PP/ST author **should** specify the set of attributes that **can** be used to  
9199 determine the scope of the session security attributes.

EXAMPLE

Examples of such attributes are user identity, originating location, time of access, and method of access.

9200 **L.3 Limitation on multiple concurrent sessions (FTA\_MCS)**

9201 **L.3.1 User notes**

9202 This family defines how many sessions a user **may** have at the same time (concurrent sessions).  
9203 This number of concurrent sessions **may** either be set for a group of users or for each individual  
9204 user.

9205 **L.3.2 FTA\_MCS.1 Basic limitation on multiple concurrent sessions**

9206 **L.3.2.1 User application notes**

9207 This component allows the system to limit the number of sessions in order to effectively use the  
9208 resources of the TOE.

9209 **L.3.2.2 Operations**

9210 **L.3.2.2.1 Assignment**

9211 In FTA\_MCS.1.2, the PP/ST author **should** specify the default number of maximum concurrent  
9212 sessions to be used.

9213 **L.3.3 FTA\_MCS.2 Per user attribute limitation on multiple concurrent sessions**

9214 **L.3.3.1 User application notes**

9215 This component provides additional capabilities over those of FTA\_MCS.1 Basic limitation on  
9216 multiple concurrent sessions, by allowing further constraints to be placed on the number of  
9217 concurrent sessions that users are able to invoke. These constraints are in terms of a user's  
9218 security attributes, such as a user's identity, or membership of a role.

9219 **L.3.3.2 Operations**

9220 **L.3.3.2.1 Assignment**

9221 In FTA\_MCS.2.1, the PP/ST author **should** specify the rules that determine the maximum  
9222 number of concurrent sessions.

EXAMPLE

An example of a rule is "maximum number of concurrent sessions is one if the user has a classification level of  
"secret" and five otherwise".

9223 In FTA\_MCS.2.2, the PP/ST author **should** specify the default number of maximum concurrent  
9224 sessions to be used.

9225 **L.4 Session locking and termination (FTA\_SSL)**

9226 **L.4.1 User notes**

9227 This family defines requirements for the TSF to provide the capability for TSF-initiated and  
9228 user-initiated locking, unlocking, and termination of interactive sessions.

When a user is directly interacting with subjects in the TOE (interactive session), the user's terminal is vulnerable if left unattended. This family provides requirements for the TSF to disable (lock) the terminal or terminate the session after a specified period of inactivity, and for the user to initiate the disabling (locking) of the terminal or terminate the session. To reactivate the terminal, an event specified by the PP/ST author, such as the user re-authentication must occur.

A user is considered inactive, if he/she has not provided any stimulus to the TOE for a specified period of time.

A PP/ST author **should** consider whether FTP\_TRP.1 Trusted path **should** be included. In that case, the function “session locking” **should** be included in the operation in FTP\_TRP.1 Trusted path.

## **L.4.2 FTA\_SSL.1 TSF-initiated session locking**

### **L.4.2.1 User application notes**

FTA\_SSL.1 TSF-initiated session locking, provides the capability for the TSF to lock an active user session after a specified period of time. Locking a terminal would prevent any further interaction with an existing active session through the use of the locked terminal.

If display devices are overwritten, the replacement contents need not be static (i.e. “screen savers” are permitted).

This component allows the PP/ST author to specify what events will unlock the session. These events **may** be related to the terminal, the user, or time.

#### **EXAMPLE**

Terminal related: a fixed set of keystrokes to unlock the session.

User related: reauthentication.

Time related: after 15 minutes.

## **L.4.2.2 Operations**

### **L.4.2.2.1 Assignment**

In FTA\_SSL.1.1, the PP/ST author **should** specify the interval of user inactivity that will trigger the locking of an interactive session. If so desired the PP/ST author **could**, through the assignment, specify that the time interval is left to the authorized administrator or the user. The management functions in the FMT class **can** specify the capability to modify this time interval, making it the default value.

In FTA\_SSL.1.2, the PP/ST author **should** specify the event(s) that **should** occur before the session is unlocked.

#### **EXAMPLE**

Examples of such an event are: “user re-authentication” or “user enters unlock key-sequence”.

## **L.4.3 FTA\_SSL.2 User-initiated locking**

### **L.4.3.1 User application notes**

FTA\_SSL.2 User-initiated locking, provides the capability for an authorized user to lock and unlock his/her own interactive session. This would provide authorized users with the ability to effectively block further use of their active sessions without having to terminate the active session.

If devices are overwritten, the replacement contents need not be static (i.e. “screen savers” are permitted).

9268 **L.4.3.2 Operations**9269 **L.4.3.2.1 Assignment**

9270 In FTA\_SSL.2.2, the PP/ST author **should** specify the event(s) that **should** occur before the  
 9271 session is unlocked.

**EXAMPLE**

Examples of such an event are: “user re-authentication”, or “user enters unlock key-sequence”.

9272 **L.4.4 FTA\_SSL.3 TSF-initiated termination**9273 **L.4.4.1 User application notes**

9274 FTA\_SSL.3 TSF-initiated termination, requires that the TSF terminate an interactive user  
 9275 session after a period of inactivity.

9276 The PP/ST author **should** be aware that a session **may** continue after the user terminated  
 9277 his/her activity, for example, background processing. This requirement would terminate this  
 9278 background subject after a period of inactivity of the user without regard to the status of the  
 9279 subject.

9280 **L.4.4.2 Operations**9281 **L.4.4.2.1 Assignment**

9282 In FTA\_SSL.3.1, the PP/ST author **should** specify the interval of user inactivity that will trigger  
 9283 the termination of an interactive session. If so desired, the PP/ST author **could**, through the  
 9284 assignment, specify that the interval is left to the authorized administrator or the user. The  
 9285 management functions in the FMT class **can** specify the capability to modify this time interval,  
 9286 making it the default value.

9287 **L.4.5 FTA\_SSL.4 User-initiated termination**9288 **L.4.5.1 User application notes**

9289

9290 FTA\_SSL.4 User-initiated termination, provides the capability for an authorized user to  
 9291 terminate his/her interactive session.

9292 The PP/ST author **should** be aware that a session **may** continue after the user terminated  
 9293 his/her activity.

**EXAMPLE**

background processing

9294 This requirement would allow the user to terminate this background subject without regard to  
 9295 the status of the subject.

9296 **L.5TOE access banners (FTA\_TAB)**9297 **L.5.1 User notes**

9298 Prior to identification and authentication, TOE access requirements provide the ability for the  
 9299 TOE to display an advisory warning message to potential users pertaining to appropriate use of  
 9300 the TOE.

9301 **L.5.2 FTA\_TAB.1 Default TOE access banners**9302 **L.5.2.1 User application notes**

9303 This component requires that there is an advisory warning regarding the unauthorized use of  
 9304 the TOE. A PP/ST author **could** refine the requirement to include a default banner.

9305 **L.6TOE access history (FTA\_TAH)**

9306 **L.6.1 User notes**

9307 This family defines requirements for the TSF to display to users, upon successful session  
 9308 establishment to the TOE, a history of unsuccessful attempts to access the account. This history  
 9309 **may** include the date, time, means of access, and port of the last successful access to the TOE, as  
 9310 well as the number of unsuccessful attempts to access the TOE since the last successful access  
 9311 by the identified user.

9312 **L.6.2 FTA\_TAH.1 TOE access history**9313 **L.6.2.1 User application notes**

9314 This family **can** provide authorized users with information that **may** indicate the possible  
 9315 misuse of their user account.

9316 This component requests that the user is presented with the information. The user **should** be  
 9317 able to review the information but is not forced to do so.

**EXAMPLE**

If a user so desires he might, create scripts that ignore this information and start other processes.

9318 **L.6.2.2 Operations**9319 **L.6.2.2.1 Selection**

9320 In FTA\_TAH.1.1, the PP/ST author **should** select the security attributes of the last successful  
 9321 session establishment that will be shown at the user interface. The items are: date, time,  
 9322 method of access, and/or location.

9323 In FTA\_TAH.1.2, the PP/ST author **should** select the security attributes of the last unsuccessful  
 9324 session establishment that will be shown at the user interface. The items are: date, time,  
 9325 method of access, and/or location.

**EXAMPLE**

Method of access: ftp.

Location: terminal 50.

9326

9327 **L.7 TOE session establishment (FTA\_TSE)**9328 **L.7.1 User notes**

9329 This family defines requirements to deny a user permission to establish a session with the TOE  
 9330 based on attributes such as the location or port of access, the user's security attribute, ranges of  
 9331 time or combinations of parameters.

**EXAMPLE 1**

security attribute: identity, clearance level, integrity level, membership in a role.

ranges of time: time-of-day, day-of-week, calendar dates.

9332 This family provides the capability for the PP/ST author to specify requirements for the TOE to  
 9333 place constraints on the ability of an authorized user to establish a session with the TOE. The  
 9334 identification of relevant constraints **can** be achieved through the use of the selection operation.

**EXAMPLE 2**

Examples of attributes that **could** be used to specify the session establishment constraints are:

- a) The location of access can be used to constrain the ability of a user to establish an active session with the TOE, based on the user's location or port of access. This capability is of particular use in environments where dial-up facilities or network facilities are available.



- b) The user's security attributes can be used to place constraints on the ability of a user to establish an active session with the TOE. For example, these attributes would provide the capability to deny session establishment based on any of the following:

- a user's identity;
- a user's clearance level;
- a user's integrity level; and
- a user's membership in a role.

This capability is particularly relevant in situations where authorization or login may take place at a different location from where TOE access checks are performed.

- c) The time of access can be used to constrain the ability of a user to establish an active session with the TOE based on ranges of time. For example, ranges may be based upon time-of-day, day-of-week, or calendar dates. This constraint provides some operational protection against actions that **could** occur at a time where proper monitoring or where proper procedural measures may not be in place.

## 9335 L.7.2 FTA\_TSE.1 TOE session establishment

### 9336 L.7.2.1 Operations

#### 9337 L.7.2.1.1 Assignment

9338 In FTA\_TSE.1.1, the PP/ST author **should** specify the attributes that **can** be used to restrict the  
9339 session establishment.

#### EXAMPLE

Examples of possible attributes are user identity, originating location (such as no remote terminals), time of access (such as outside hours), or method of access (such as telnet).

9340

## Annex M (normative)

### Class FTP: Trusted path/channels- application notes

#### M.1 General information

Users often need to perform functions through direct interaction with the TSF. A trusted path provides confidence that a user is communicating directly with the TSF whenever it is invoked. A user's response via the trusted path guarantees that untrusted applications cannot intercept or modify the user's response. Similarly, trusted channels are one approach for secure communication between the TSF and another trusted IT product.

Absence of a trusted path **may** allow breaches of accountability or access control in environments where untrusted applications are used. These applications **can** intercept user-private information, such as passwords, and use it to impersonate other users. As a consequence, responsibility for any system actions cannot be reliably assigned to an accountable entity. Also, these applications **could** output erroneous information on an unsuspecting user's display, resulting in subsequent user actions that **may** be erroneous and **may** lead to a security breach.

#### M.2 Inter-TSF trusted channel (FTP\_ITC)

##### M.2.1 User notes

This family defines the rules for the creation of a trusted channel connection that goes between the TSF and another trusted IT product for the performance of security critical operations between the products.

###### EXAMPLE

An example of such a security critical operation is the updating of the TSF authentication database by the transfer of data from a trusted product whose function is the collection of audit data.

##### M.2.2 FTP\_ITC.1 Inter-TSF trusted channel

###### M.2.2.1 User application notes

This component **should** be used when a trusted communication channel between the TSF and another trusted IT product is required.

###### M.2.2.2 Operations

###### M.2.2.2.1 Selection

In FTP\_ITC.1.2, the PP/ST author must specify whether the local TSF, another trusted IT product, or both **shall** have the capability to initiate the trusted channel.

###### M.2.2.2.2 Assignment

In FTP\_ITC.1.3, the PP/ST author **should** specify the functions for which a trusted channel is required.

###### EXAMPLE

Examples of these functions **may** include transfer of user, subject, and/or object security attributes and ensuring consistency of TSF data.

#### M.3 Secure channel (FTP\_PRO)

##### M.3.1 User notes

This family defines the rules for the creation of a secure channel connection that goes between the TSF and another trusted IT product for the protection of data transfers.

Separate iterations of the relevant FTP\_PRO SFRs may be used for different roles where the completion of the SFR needs to be different for each role.

### **M.3.2 FTP\_PRO.1**

#### **M.3.2.1 User application notes**

#### **M.3.2.2 Operations**

##### **M.3.2.2.1 Assignment**

In FTP\_PRO.1.1, if selected, the PP/ST author **should** specify a trusted channel protocol and the defined protocol roles.

#### **EXAMPLE**

Examples of “defined protocol roles” would be ‘client’ or ‘server’ (TLS), ‘initiator’ or ‘responder’ (IKEv2/IPsec), ‘Trust Center’ (ZigBee) or ‘Key Distribution Centre’ (Kerberos).

### **M.3.3 FTP\_PRO.2**

#### **M.3.3.1 User application notes**

#### **M.3.3.2 Operations**

##### **M.3.3.2.1 Assignment**

### **M.3.4 FTP\_PRO.3**

#### **M.3.4.1 User application notes**

#### **M.3.4.2 Operations**

##### **M.3.4.2.1 Assignment**

## **M.4 Trusted path (FTP\_TRP)**

### **M.4.1 User notes**

This family defines the requirements to establish and maintain trusted communication to or from users and the TSF. A trusted path **may** be required for any security-relevant interaction. Trusted path exchanges **may** be initiated by a user during an interaction with the TSF, or the TSF **may** establish communication with the user via a trusted path.

### **M.4.2 FTP\_TRP.1 Trusted path**

#### **M.4.2.1 User application notes**

This component **should** be used when trusted communication between a user and the TSF is required, either for initial authentication purposes only or for additional specified user operations.

#### **M.4.2.2 Operations**

##### **M.4.2.2.1 Selection**

In FTP\_TRP.1.1, the PP/ST author **should** specify whether the trusted path must be extended to remote and/or local users.

In FTP\_TRP.1.1, the PP/ST author **should** specify whether the trusted path **shall** protect the data from modification, disclosure, and/or other types of integrity or confidentiality violation.

##### **M.4.2.2.2 Assignment**

In FTP\_TRP.1.1, if selected, the PP/ST author **should** identify any additional types of integrity or confidentiality violation against which the trusted path **shall** protect the data.

9416 **M.4.2.2.3 Selection**

9417 In FTP\_TRP.1.2, the PP/ST author **should** specify whether the TSF, local users, and/or remote  
9418 users **should** be able to initiate the trusted path.

9419 In FTP\_TRP.1.3, the PP/ST author **should** specify whether the trusted path is to be used for  
9420 initial user authentication and/or for other specified services.

9421 **M.4.2.2.4 Assignment**

9422 In FTP\_TRP.1.3, if selected, the PP/ST author **should** identify other services for which trusted  
9423 path is required, if any.