



# International Standard

**ISO/IEC 15408-2**

## Information security, cybersecurity and privacy protection — Evaluation criteria for IT security —

### Part 2: Security functional components

*Sécurité de l'information, cybersécurité et protection de la vie  
privée — Critères d'évaluation pour la sécurité des technologies  
de l'information —*

*Partie 2: Composants fonctionnels de sécurité*

**Fifth edition  
2026-05**



Please share your feedback about  
the standard. Scan the QR code  
with your phone or click the link

[Customer Feedback Form](#)

## **ISO/IEC 15408-2:2026(en)**



### **COPYRIGHT PROTECTED DOCUMENT**

© ISO/IEC 2026

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

ISO copyright office  
CP 401 • Ch. de Blandonnet 8  
CH-1214 Vernier, Geneva  
Phone: +41 22 749 01 11  
Email: [copyright@iso.org](mailto:copyright@iso.org)  
Website: [www.iso.org](http://www.iso.org)

Published in Switzerland

# Contents

	Page
<b>Foreword</b> .....	<b>xv</b>
<b>Introduction</b> .....	<b>xvi</b>
<b>1 Scope</b> .....	<b>1</b>
<b>2 Normative references</b> .....	<b>1</b>
<b>3 Terms and definitions</b> .....	<b>1</b>
<b>4 Abbreviated terms</b> .....	<b>3</b>
<b>5 Overview</b> .....	<b>4</b>
5.1 General.....	4
5.2 Organization of this document.....	4
<b>6 Functional requirements paradigm</b> .....	<b>5</b>
<b>7 Security functional components</b> .....	<b>8</b>
7.1 Overview.....	8
7.2 Functional class structure.....	8
7.2.1 General.....	8
7.2.2 Class name.....	9
7.2.3 Class introduction.....	9
7.2.4 Class informative notes.....	9
7.2.5 Functional families.....	9
7.3 Functional family structure.....	9
7.3.1 General.....	9
7.3.2 Family name.....	10
7.3.3 Family Behaviour.....	10
7.3.4 Component levelling and description.....	10
7.3.5 Component management.....	10
7.3.6 Component audit.....	11
7.3.7 Family application notes.....	11
7.3.8 Family evaluator notes.....	11
7.3.9 Functional components.....	11
7.4 Functional component structure.....	12
7.4.1 General.....	12
7.4.2 Component name.....	12
7.4.3 Component relationships.....	12
7.4.4 Component rationale.....	13
7.4.5 Component notes.....	13
7.4.6 Functional elements.....	13
7.5 Functional elements.....	13
7.6 Component catalogue.....	14
7.6.1 General.....	14
7.6.2 Highlighting of component changes.....	15
<b>8 Class FAU Security audit</b> .....	<b>15</b>
8.1 Introduction.....	15
8.2 Notes on class FAU.....	17
8.2.1 General information about audit requirements.....	17
8.2.2 Audit requirements in a distributed environment.....	17
8.3 Security audit automatic response (FAU_ARP).....	18
8.3.1 Family Behaviour.....	18
8.3.2 Component levelling and description.....	18
8.3.3 Component management.....	18
8.3.4 Component audit.....	18
8.3.5 Application notes.....	18
8.3.6 FAU_ARP.1 Security alarms.....	19
8.4 Security audit data generation (FAU_GEN).....	19

**ISO/IEC 15408-2:2026(en)**

8.4.1	Family Behaviour	19
8.4.2	Component levelling and description	19
8.4.3	Component management	19
8.4.4	Component audit	20
8.4.5	Application notes	20
8.4.6	Evaluator notes	21
8.4.7	FAU_GEN.1 Audit data generation	21
8.4.8	FAU_GEN.2 User identity association	22
8.5	Security audit analysis (FAU_SAA)	23
8.5.1	Family Behaviour	23
8.5.2	Component levelling and description	23
8.5.3	Component management	23
8.5.4	Component audit	24
8.5.5	Application notes	24
8.5.6	FAU_SAA.1 Potential violation analysis	24
8.5.7	FAU_SAA.2 Profile based anomaly detection	25
8.5.8	FAU_SAA.3 Simple attack heuristics	26
8.5.9	FAU_SAA.4 Complex attack heuristics	27
8.6	Security audit review (FAU_SAR)	29
8.6.1	Family Behaviour	29
8.6.2	Component levelling and description	29
8.6.3	Component management	29
8.6.4	Component audit	30
8.6.5	Application notes	30
8.6.6	FAU_SAR.1 Audit review	30
8.6.7	FAU_SAR.2 Restricted audit review	31
8.6.8	FAU_SAR.3 Selectable audit review	31
8.7	Security audit event selection (FAU_SEL)	32
8.7.1	Family Behaviour	32
8.7.2	Component levelling and description	32
8.7.3	Component management	32
8.7.4	Component audit	32
8.7.5	Application notes	32
8.7.6	FAU_SEL.1 Selective audit	33
8.8	Security audit data storage (FAU_STG)	33
8.8.1	Family Behaviour	33
8.8.2	Component levelling and description	33
8.8.3	Component management	34
8.8.4	Component audit	35
8.8.5	Application notes	35
8.8.6	FAU_STG.1 Audit data storage location	35
8.8.7	FAU_STG.2 Protected audit data storage	36
8.8.8	FAU_STG.3 Guarantees of audit data availability	36
8.8.9	FAU_STG.4 Action in case of possible audit data loss	37
8.8.10	FAU_STG.5 Prevention of audit data loss	38
<b>9</b>	<b>Class FCO Communication</b>	<b>38</b>
9.1	Introduction	38
9.2	Notes on class FCO	39
9.3	Non-repudiation of origin (FCO_NRO)	39
9.3.1	Family Behaviour	39
9.3.2	Component levelling and description	39
9.3.3	Component management	40
9.3.4	Component audit	40
9.3.5	Application notes	40
9.3.6	FCO_NRO.1 Selective proof of origin	41
9.3.7	FCO_NRO.2 Enforced proof of origin	42
9.4	Non-repudiation of receipt (FCO_NRR)	42
9.4.1	Family Behaviour	42

**ISO/IEC 15408-2:2026(en)**

9.4.2	Component levelling and description .....	43
9.4.3	Component management .....	43
9.4.4	Component audit .....	43
9.4.5	Application notes .....	43
9.4.6	FCS_NRR.1 Selective proof of receipt .....	44
9.4.7	FCS_NRR.2 Enforced proof of receipt .....	45
<b>10</b>	<b>Class FCS Cryptographic support .....</b>	<b>46</b>
10.1	Introduction .....	46
10.2	Notes on class FCS .....	48
10.3	Cryptographic key management (FCS_CKM) .....	50
10.3.1	Family Behaviour .....	50
10.3.2	Component levelling and description .....	50
10.3.3	Component management .....	51
10.3.4	Component audit .....	51
10.3.5	Application notes .....	51
10.3.6	Evaluator notes .....	52
10.3.7	FCS_CKM.1 Cryptographic key generation .....	52
10.3.8	FCS_CKM.2 Cryptographic key distribution .....	53
10.3.9	FCS_CKM.3 Cryptographic key access .....	53
10.3.10	FCS_CKM.5 Cryptographic key derivation .....	54
10.3.11	FCS_CKM.6 Timing and event of cryptographic key destruction .....	55
10.4	Cryptographic operation (FCS_COP) .....	56
10.4.1	Family Behaviour .....	56
10.4.2	Component levelling and description .....	56
10.4.3	Component management .....	56
10.4.4	Component audit .....	56
10.4.5	Application notes .....	56
10.4.6	FCS_COP.1 Cryptographic operation .....	57
10.5	Random bit generation (FCS_RBG) .....	58
10.5.1	Family Behaviour .....	58
10.5.2	Component levelling and description .....	58
10.5.3	Component management .....	59
10.5.4	Component audit .....	59
10.5.5	Application notes .....	59
10.5.6	FCS_RBG.1 Random bit generation (RBG) .....	59
10.5.7	FCS_RBG.2 Random bit generation (external seeding) .....	61
10.5.8	FCS_RBG.3 Random bit generation (internal seeding - single source) .....	61
10.5.9	FCS_RBG.4 Random bit generation (internal seeding - multiple sources) .....	62
10.5.10	FCS_RBG.5 Random bit generation (combining entropy sources) .....	62
10.5.11	FCS_RBG.6 Random bit generation service .....	63
10.6	Generation of random numbers (FCS_RNG) .....	63
10.6.1	Family Behaviour .....	63
10.6.2	Component levelling and description .....	64
10.6.3	Component management .....	64
10.6.4	Component audit .....	64
10.6.5	Application notes .....	64
10.6.6	FCS_RNG.1 Random number generation .....	64
<b>11</b>	<b>Class FDP User data protection .....</b>	<b>66</b>
11.1	Introduction .....	66
11.2	Notes on class FDP .....	69
11.3	Access control policy (FDP_ACC) .....	71
11.3.1	Family Behaviour .....	71
11.3.2	Component levelling and description .....	72
11.3.3	Component management .....	72
11.3.4	Component audit .....	72
11.3.5	Application notes .....	72
11.3.6	FDP_ACC.1 Subset access control .....	73
11.3.7	FDP_ACC.2 Complete access control .....	73

**ISO/IEC 15408-2:2026(en)**

11.4	Access control functions (FDP_ACF).....	74
11.4.1	Family Behaviour.....	74
11.4.2	Component levelling and description.....	74
11.4.3	Component management.....	74
11.4.4	Component audit.....	75
11.4.5	Application notes.....	75
11.4.6	FDP_ACF.1 Security attribute-based access control.....	75
11.5	Data authentication (FDP_DAU).....	77
11.5.1	Family Behaviour.....	77
11.5.2	Component levelling and description.....	77
11.5.3	Component management.....	77
11.5.4	Component audit.....	77
11.5.5	Application notes.....	78
11.5.6	FDP_DAU.1 Basic Data Authentication.....	78
11.5.7	FDP_DAU.2 Data Authentication with Identity of Guarantor.....	78
11.6	Export from the TOE (FDP_ETC).....	79
11.6.1	Family Behaviour.....	79
11.6.2	Component levelling and description.....	79
11.6.3	Component management.....	80
11.6.4	Component audit.....	80
11.6.5	Application notes.....	80
11.6.6	FDP_ETC.1 Export of user data without security attributes.....	80
11.6.7	FDP_ETC.2 Export of user data with security attributes.....	81
11.7	Information flow control policy (FDP_IFC).....	82
11.7.1	Family Behaviour.....	82
11.7.2	Component levelling and description.....	82
11.7.3	Component management.....	82
11.7.4	Component audit.....	82
11.7.5	Application notes.....	82
11.7.6	FDP_IFC.1 Subset information flow control.....	83
11.7.7	FDP_IFC.2 Complete information flow control.....	84
11.8	Information flow control functions (FDP_IFF).....	85
11.8.1	Family Behaviour.....	85
11.8.2	Component levelling and description.....	85
11.8.3	Component management.....	86
11.8.4	Component audit.....	86
11.8.5	Application notes.....	86
11.8.6	FDP_IFF.1 Simple security attributes.....	87
11.8.7	FDP_IFF.2 Hierarchical security attributes.....	88
11.8.8	FDP_IFF.3 Limited illicit information flows.....	90
11.8.9	FDP_IFF.4 Partial elimination of illicit information flows.....	91
11.8.10	FDP_IFF.5 No illicit information flows.....	91
11.8.11	FDP_IFF.6 Illicit information flow monitoring.....	92
11.9	Information retention control (FDP_IRC).....	92
11.9.1	Family Behaviour.....	92
11.9.2	Component levelling and description.....	93
11.9.3	Component management.....	93
11.9.4	Component audit.....	93
11.9.5	Application notes.....	93
11.9.6	FDP_IRC.1 Information retention control.....	94
11.10	Import from outside of the TOE (FDP_ITC).....	94
11.10.1	Family Behaviour.....	94
11.10.2	Component levelling and description.....	94
11.10.3	Component management.....	95
11.10.4	Component audit.....	95
11.10.5	Application notes.....	95
11.10.6	FDP_ITC.1 Import of user data without security attributes.....	96
11.10.7	FDP_ITC.2 Import of user data with security attributes.....	97
11.11	Internal TOE transfer (FDP_ITT).....	98

**ISO/IEC 15408-2:2026(en)**

11.11.1	Family Behaviour.....	98
11.11.2	Component levelling and description.....	98
11.11.3	Component management.....	98
11.11.4	Component audit.....	99
11.11.5	Application notes.....	99
11.11.6	FDP_ITT.1 Basic internal transfer protection.....	99
11.11.7	FDP_ITT.2 Transmission separation by attribute.....	100
11.11.8	FDP_ITT.3 Integrity monitoring.....	101
11.11.9	FDP_ITT.4 Attribute-based integrity monitoring.....	101
11.12	Residual information protection (FDP_RIP).....	102
11.12.1	Family Behaviour.....	102
11.12.2	Component levelling and description.....	103
11.12.3	Component management.....	103
11.12.4	Component audit.....	103
11.12.5	Application notes.....	103
11.12.6	FDP_RIP.1 Subset residual information protection.....	104
11.12.7	FDP_RIP.2 Full residual information protection.....	105
11.13	Rollback (FDP_ROL).....	105
11.13.1	Family Behaviour.....	105
11.13.2	Component levelling and description.....	105
11.13.3	Component management.....	105
11.13.4	Component audit.....	106
11.13.5	Application notes.....	106
11.13.6	FDP_ROL.1 Basic rollback.....	106
11.13.7	FDP_ROL.2 Advanced rollback.....	107
11.14	Stored data confidentiality (FDP_SDC).....	108
11.14.1	Family Behaviour.....	108
11.14.2	Component levelling and description.....	108
11.14.3	Component management.....	108
11.14.4	Component audit.....	108
11.14.5	Application notes.....	108
11.14.6	Evaluator notes.....	108
11.14.7	FDP_SDC.1 Stored data confidentiality.....	109
11.14.8	FDP_SDC.2 Stored data confidentiality with dedicated method.....	109
11.15	Stored data integrity (FDP_SDI).....	110
11.15.1	Family Behaviour.....	110
11.15.2	Component levelling and description.....	110
11.15.3	Component management.....	110
11.15.4	Component audit.....	110
11.15.5	Application notes.....	111
11.15.6	FDP_SDI.1 Stored data integrity monitoring.....	111
11.15.7	FDP_SDI.2 Stored data integrity monitoring and action.....	111
11.16	Inter-TSF user data confidentiality transfer protection (FDP_UCT).....	112
11.16.1	Family Behaviour.....	112
11.16.2	Component levelling and description.....	112
11.16.3	Component management.....	112
11.16.4	Component audit.....	112
11.16.5	Application notes.....	113
11.16.6	FDP_UCT.1 Basic data exchange confidentiality.....	113
11.17	Inter-TSF user data integrity transfer protection (FDP_UIT).....	113
11.17.1	Family Behaviour.....	113
11.17.2	Component levelling and description.....	113
11.17.3	Component management.....	114
11.17.4	Component audit.....	114
11.17.5	Application notes.....	115
11.17.6	FDP_UIT.1 Data exchange integrity.....	115
11.17.7	FDP_UIT.2 Source data exchange recovery.....	116
11.17.8	FDP_UIT.3 Destination data exchange recovery.....	116

**ISO/IEC 15408-2:2026(en)**

<b>12</b>	<b>Class FIA Identification and authentication</b>	<b>117</b>
12.1	Introduction	117
12.2	Notes on class FIA	119
12.3	Authentication failures (FIA_AFL)	120
12.3.1	Family Behaviour	120
12.3.2	Component levelling and description	120
12.3.3	Component management	120
12.3.4	Component audit	120
12.3.5	Application notes	120
12.3.6	FIA_AFL.1 Authentication failure handling	121
12.4	Authentication proof of identity (FIA_API)	122
12.4.1	Family Behaviour	122
12.4.2	Component levelling and description	122
12.4.3	Component management	122
12.4.4	Component audit	122
12.4.5	Application notes	122
12.4.6	FIA_API.1 Authentication proof of identity	123
12.5	User attribute definition (FIA_ATD)	123
12.5.1	Family Behaviour	123
12.5.2	Component levelling and description	123
12.5.3	Component management	124
12.5.4	Component audit	124
12.5.5	Application notes	124
12.5.6	FIA_ATD.1 User attribute definition	124
12.6	Specification of secrets (FIA_SOS)	124
12.6.1	Family Behaviour	124
12.6.2	Component levelling and description	124
12.6.3	Component management	125
12.6.4	Component audit	125
12.6.5	Application notes	125
12.6.6	FIA_SOS.1 Verification of secrets	126
12.6.7	FIA_SOS.2 TSF Generation of secrets	126
12.7	User authentication (FIA_UAU)	127
12.7.1	Family Behaviour	127
12.7.2	Component levelling and description	127
12.7.3	Component management	128
12.7.4	Component audit	128
12.7.5	Application notes	129
12.7.6	FIA_UAU.1 Timing of authentication	129
12.7.7	FIA_UAU.2 User authentication before any action	130
12.7.8	FIA_UAU.3 Unforgeable authentication	130
12.7.9	FIA_UAU.4 Single-use authentication mechanisms	131
12.7.10	FIA_UAU.5 Multiple authentication mechanisms	131
12.7.11	FIA_UAU.6 Re-authenticating	132
12.7.12	FIA_UAU.7 Protected authentication feedback	132
12.8	User identification (FIA_UID)	133
12.8.1	Family Behaviour	133
12.8.2	Component levelling and description	133
12.8.3	Component management	133
12.8.4	Component audit	133
12.8.5	Application notes	133
12.8.6	FIA_UID.1 Timing of identification	134
12.8.7	FIA_UID.2 User identification before any action	134
12.9	User-subject binding (FIA_USB)	134
12.9.1	Family Behaviour	134
12.9.2	Component levelling and description	135
12.9.3	Component management	135
12.9.4	Component audit	135
12.9.5	Application notes	135

**ISO/IEC 15408-2:2026(en)**

	12.9.6	FIA_USB.1 User-subject binding.....	135
<b>13</b>		<b>Class FMT Security management.....</b>	<b>136</b>
	13.1	Introduction.....	136
	13.2	Notes on class FMT.....	138
	13.3	Limited capabilities and availability (FMT_LIM).....	138
	13.3.1	Family Behaviour.....	138
	13.3.2	Component levelling and description.....	139
	13.3.3	Component management.....	139
	13.3.4	Component audit.....	139
	13.3.5	Application notes.....	139
	13.3.6	FMT_LIM.1 Limited capabilities.....	139
	13.3.7	FMT_LIM.2 Limited availability.....	140
	13.4	Management of functions in TSF (FMT_MOF).....	140
	13.4.1	Family Behaviour.....	140
	13.4.2	Component levelling and description.....	140
	13.4.3	Component management.....	141
	13.4.4	Component audit.....	141
	13.4.5	Application notes.....	141
	13.4.6	FMT_MOF.1 Management of security functions behaviour.....	142
	13.5	Management of security attributes (FMT_MSA).....	142
	13.5.1	Family Behaviour.....	142
	13.5.2	Component levelling and description.....	142
	13.5.3	Component management.....	143
	13.5.4	Component audit.....	144
	13.5.5	Application notes.....	144
	13.5.6	FMT_MSA.1 Management of security attributes.....	144
	13.5.7	FMT_MSA.2 Secure security attributes.....	145
	13.5.8	FMT_MSA.3 Static attribute initialization.....	146
	13.5.9	FMT_MSA.4 Security attribute value inheritance.....	146
	13.6	Management of TSF data (FMT_MTD).....	147
	13.6.1	Family Behaviour.....	147
	13.6.2	Component levelling and description.....	147
	13.6.3	Component management.....	147
	13.6.4	Component audit.....	148
	13.6.5	Application notes.....	148
	13.6.6	FMT_MTD.1 Management of TSF data.....	148
	13.6.7	FMT_MTD.2 Management of limits on TSF data.....	149
	13.6.8	FMT_MTD.3 Secure TSF data.....	149
	13.7	Revocation (FMT_REV).....	150
	13.7.1	Family Behaviour.....	150
	13.7.2	Component levelling and description.....	150
	13.7.3	Component management.....	150
	13.7.4	Component audit.....	150
	13.7.5	Application notes.....	150
	13.7.6	FMT_REV.1 Revocation.....	150
	13.8	Security attribute expiration (FMT_SAE).....	151
	13.8.1	Family Behaviour.....	151
	13.8.2	Component levelling and description.....	151
	13.8.3	Component management.....	152
	13.8.4	Component audit.....	152
	13.8.5	Application notes.....	152
	13.8.6	FMT_SAE.1 Time-limited authorization.....	152
	13.9	Specification of Management Functions (FMT_SMF).....	153
	13.9.1	Family Behaviour.....	153
	13.9.2	Component levelling and description.....	153
	13.9.3	Component management.....	153
	13.9.4	Component audit.....	153
	13.9.5	Application notes.....	153

**ISO/IEC 15408-2:2026(en)**

	13.9.6	FMT_SMF.1 Specification of Management Functions.....	153
13.10		Security management roles (FMT_SMR).....	154
	13.10.1	Family Behaviour.....	154
	13.10.2	Component levelling and description.....	154
	13.10.3	Component management.....	154
	13.10.4	Component audit.....	155
	13.10.5	Application notes.....	155
	13.10.6	FMT_SMR.1 Security roles.....	155
	13.10.7	FMT_SMR.2 Restrictions on security roles.....	156
	13.10.8	FMT_SMR.3 Assuming roles.....	157
<b>14</b>		<b>Class FPR Privacy.....</b>	<b>157</b>
	14.1	Introduction.....	157
	14.2	Notes on class FPR.....	158
	14.3	Anonymity (FPR_ANO).....	159
	14.3.1	Family Behaviour.....	159
	14.3.2	Component levelling and description.....	159
	14.3.3	Component management.....	159
	14.3.4	Component audit.....	159
	14.3.5	Application notes.....	159
	14.3.6	FPR_ANO.1 Anonymity.....	160
	14.3.7	FPR_ANO.2 Anonymity without soliciting information.....	161
	14.4	Pseudonymity (FPR_PSE).....	161
	14.4.1	Family Behaviour.....	161
	14.4.2	Component levelling and description.....	161
	14.4.3	Component management.....	162
	14.4.4	Component audit.....	162
	14.4.5	Application notes.....	162
	14.4.6	FPR_PSE.1 Pseudonymity.....	163
	14.4.7	FPR_PSE.2 Reversible pseudonymity.....	164
	14.4.8	FPR_PSE.3 Alias pseudonymity.....	165
	14.5	Unlinkability (FPR_UNL).....	167
	14.5.1	Family Behaviour.....	167
	14.5.2	Component levelling and description.....	167
	14.5.3	Component management.....	167
	14.5.4	Component audit.....	167
	14.5.5	Application notes.....	167
	14.5.6	FPR_UNL.1 Unlinkability of operations.....	168
	14.6	Unobservability (FPR_UNO).....	169
	14.6.1	Family Behaviour.....	169
	14.6.2	Component levelling and description.....	169
	14.6.3	Component management.....	169
	14.6.4	Component audit.....	169
	14.6.5	Application notes.....	170
	14.6.6	FPR_UNO.1 Unobservability.....	170
	14.6.7	FPR_UNO.2 Allocation of information impacting unobservability.....	171
	14.6.8	FPR_UNO.3 Unobservability without soliciting information.....	172
	14.6.9	FPR_UNO.4 Authorized user observability.....	173
<b>15</b>		<b>Class FPT Protection of the TSF.....</b>	<b>173</b>
	15.1	Introduction.....	173
	15.2	Notes on class FPT.....	176
	15.3	TOE emanation (FPT_EMS).....	177
	15.3.1	Family Behaviour.....	177
	15.3.2	Component levelling and description.....	177
	15.3.3	Component management.....	177
	15.3.4	Component audit.....	178
	15.3.5	Application notes.....	178
	15.3.6	FPT_EMS.1 Emanation of TSF and User data.....	178
	15.4	Fail secure (FPT_FLS).....	179

**ISO/IEC 15408-2:2026(en)**

15.4.1	Family Behaviour.....	179
15.4.2	Component levelling and description.....	179
15.4.3	Component management.....	179
15.4.4	Component audit.....	179
15.4.5	Application notes.....	179
15.4.6	FPT_FLS.1 Failure with preservation of secure state.....	179
15.5	TSF initialization (FPT_INI).....	180
15.5.1	Family Behaviour.....	180
15.5.2	Component levelling and description.....	180
15.5.3	Component management.....	180
15.5.4	Component audit.....	180
15.5.5	Application notes.....	180
15.5.6	FPT_INI.1 TSF initialization.....	181
15.6	Availability of exported TSF data (FPT_ITA).....	182
15.6.1	Family Behaviour.....	182
15.6.2	Component levelling and description.....	182
15.6.3	Component management.....	182
15.6.4	Component audit.....	182
15.6.5	Application notes.....	182
15.6.6	FPT_ITA.1 Inter-TSF availability within a defined availability metric.....	182
15.7	Confidentiality of exported TSF data (FPT_ITC).....	183
15.7.1	Family Behaviour.....	183
15.7.2	Component levelling and description.....	183
15.7.3	Component management.....	183
15.7.4	Component audit.....	183
15.7.5	Application notes.....	183
15.7.6	Evaluator notes.....	184
15.7.7	FPT_ITC.1 Inter-TSF confidentiality during transmission.....	184
15.8	Integrity of exported TSF data (FPT_ITI).....	184
15.8.1	Family Behaviour.....	184
15.8.2	Component levelling and description.....	184
15.8.3	Component management.....	184
15.8.4	Component audit.....	185
15.8.5	Application notes.....	185
15.8.6	Evaluator notes.....	185
15.8.7	FPT_ITI.1 Inter-TSF detection of modification.....	185
15.8.8	FPT_ITI.2 Inter-TSF detection and correction of modification.....	186
15.9	Internal TOE TSF data transfer (FPT_ITT).....	187
15.9.1	Family Behaviour.....	187
15.9.2	Component levelling and description.....	187
15.9.3	Component management.....	187
15.9.4	Component audit.....	188
15.9.5	Application notes.....	188
15.9.6	Evaluator notes.....	188
15.9.7	FPT_ITT.1 Basic internal TSF data transfer protection.....	188
15.9.8	FPT_ITT.2 TSF data transfer separation.....	189
15.9.9	FPT_ITT.3 TSF data integrity monitoring.....	189
15.10	TSF physical protection (FPT_PHP).....	190
15.10.1	Family Behaviour.....	190
15.10.2	Component levelling and description.....	190
15.10.3	Component management.....	190
15.10.4	Component audit.....	191
15.10.5	Application notes.....	191
15.10.6	FPT_PHP.1 Passive detection of physical attack.....	191
15.10.7	FPT_PHP.2 Notification of physical attack.....	192
15.10.8	FPT_PHP.3 Resistance to physical attack.....	193
15.11	Trusted recovery (FPT_RCV).....	193
15.11.1	Family Behaviour.....	193
15.11.2	Component levelling and description.....	193

**ISO/IEC 15408-2:2026(en)**

15.11.3	Component management .....	194
15.11.4	Component audit .....	194
15.11.5	Application notes .....	195
15.11.6	Evaluator notes .....	196
15.11.7	FPT_RCV.1 Manual recovery .....	196
15.11.8	FPT_RCV.2 Automated recovery .....	197
15.11.9	FPT_RCV.3 Automated recovery without undue loss .....	197
15.11.10	.....	
	FPT_RCV.4 Function recovery .....	198
15.12	Replay detection (FPT_RPL) .....	199
15.12.1	Family Behaviour .....	199
15.12.2	Component levelling and description .....	199
15.12.3	Component management .....	199
15.12.4	Component audit .....	199
15.12.5	Application notes .....	199
15.12.6	FPT_RPL.1 Replay detection .....	199
15.13	State synchrony protocol (FPT_SSP) .....	200
15.13.1	Family Behaviour .....	200
15.13.2	Component levelling and description .....	200
15.13.3	Component management .....	200
15.13.4	Component audit .....	200
15.13.5	Application notes .....	201
15.13.6	FPT_SSP.1 Simple trusted acknowledgement .....	201
15.13.7	FPT_SSP.2 Mutual trusted acknowledgement .....	201
15.14	Time stamps (FPT_STM) .....	202
15.14.1	Family Behaviour .....	202
15.14.2	Component levelling and description .....	202
15.14.3	Component management .....	202
15.14.4	Component audit .....	203
15.14.5	Application notes .....	203
15.14.6	FPT_STM.1 Reliable time stamps .....	203
15.14.7	FPT_STM.2 Time source .....	203
15.15	Inter-TSF TSF data consistency (FPT_TDC) .....	204
15.15.1	Family Behaviour .....	204
15.15.2	Component levelling and description .....	204
15.15.3	Component management .....	204
15.15.4	Component audit .....	204
15.15.5	Application notes .....	204
15.15.6	FPT_TDC.1 Inter-TSF basic TSF data consistency .....	205
15.16	Testing of external entities (FPT_TEE) .....	205
15.16.1	Family Behaviour .....	205
15.16.2	Component levelling and description .....	206
15.16.3	Component management .....	206
15.16.4	Component audit .....	206
15.16.5	Application notes .....	206
15.16.6	Evaluator notes .....	206
15.16.7	FPT_TEE.1 Testing of external entities .....	207
15.17	Internal TOE TSF data replication consistency (FPT_TRC) .....	208
15.17.1	Family Behaviour .....	208
15.17.2	Component levelling and description .....	208
15.17.3	Component management .....	208
15.17.4	Component audit .....	208
15.17.5	Application notes .....	208
15.17.6	FPT_TRC.1 Internal TSF consistency .....	209
15.18	TSF self-test (FPT_TST) .....	209
15.18.1	Family Behaviour .....	209
15.18.2	Component levelling and description .....	209
15.18.3	Component management .....	210
15.18.4	Component audit .....	210

**ISO/IEC 15408-2:2026(en)**

	15.18.5 Application notes .....	210
	15.18.6 Evaluator notes .....	210
	15.18.7 FPT_TST.1 TSF self-testing .....	210
<b>16</b>	<b>Class FRU Resource utilization .....</b>	<b>211</b>
	16.1 Introduction .....	211
	16.2 Notes on class FRU .....	212
	16.3 Fault tolerance (FRU_FLT) .....	212
	16.3.1 Family Behaviour .....	212
	16.3.2 Component levelling and description .....	213
	16.3.3 Component management .....	213
	16.3.4 Component audit .....	213
	16.3.5 Application notes .....	213
	16.3.6 FRU_FLT.1 Degraded fault tolerance .....	214
	16.3.7 FRU_FLT.2 Limited fault tolerance .....	214
	16.4 Priority of service (FRU_PRS) .....	215
	16.4.1 Family Behaviour .....	215
	16.4.2 Component levelling and description .....	215
	16.4.3 Component management .....	215
	16.4.4 Component audit .....	215
	16.4.5 Application notes .....	215
	16.4.6 FRU_PRS.1 Limited priority of service .....	216
	16.4.7 FRU_PRS.2 Full priority of service .....	216
	16.5 Resource allocation (FRU_RSA) .....	217
	16.5.1 Family Behaviour .....	217
	16.5.2 Component levelling and description .....	217
	16.5.3 Component management .....	217
	16.5.4 Component audit .....	217
	16.5.5 Application notes .....	217
	16.5.6 FRU_RSA.1 Maximum quotas .....	218
	16.5.7 FRU_RSA.2 Minimum and maximum quotas .....	219
<b>17</b>	<b>Class FTA TOE access .....</b>	<b>219</b>
	17.1 Introduction .....	219
	17.2 Notes on class FTA .....	221
	17.3 Limitation on scope of selectable attributes (FTA_LSA) .....	221
	17.3.1 Family Behaviour .....	221
	17.3.2 Component levelling and description .....	221
	17.3.3 Component management .....	221
	17.3.4 Component audit .....	221
	17.3.5 Application notes .....	221
	17.3.6 FTA_LSA.1 Limitation on scope of selectable attributes .....	222
	17.4 Limitation on multiple concurrent sessions (FTA_MCS) .....	222
	17.4.1 Family Behaviour .....	222
	17.4.2 Component levelling and description .....	222
	17.4.3 Component management .....	223
	17.4.4 Component audit .....	223
	17.4.5 Application notes .....	223
	17.4.6 FTA_MCS.1 Basic limitation on multiple concurrent sessions .....	223
	17.4.7 FTA_MCS.2 Per user attribute limitation on multiple concurrent sessions .....	224
	17.5 Session locking and termination (FTA_SSL) .....	224
	17.5.1 Family Behaviour .....	224
	17.5.2 Component levelling and description .....	224
	17.5.3 Component management .....	225
	17.5.4 Component audit .....	226
	17.5.5 Application notes .....	226
	17.5.6 FTA_SSL.1 TSF-initiated session locking .....	226
	17.5.7 FTA_SSL.2 User-initiated locking .....	227
	17.5.8 FTA_SSL.3 TSF-initiated termination .....	228
	17.5.9 FTA_SSL.4 User-initiated termination .....	228

**ISO/IEC 15408-2:2026(en)**

17.6	TOE access banners (FTA_TAB) .....	229
17.6.1	Family Behaviour .....	229
17.6.2	Component levelling and description .....	229
17.6.3	Component management .....	229
17.6.4	Component audit .....	229
17.6.5	Application notes .....	229
17.6.6	FTA_TAB.1 Default TOE access banners .....	229
17.7	TOE access history (FTA_TAH) .....	230
17.7.1	Family Behaviour .....	230
17.7.2	Component levelling and description .....	230
17.7.3	Component management .....	230
17.7.4	Component audit .....	230
17.7.5	Application notes .....	230
17.7.6	FTA_TAH.1 TOE access history .....	230
17.8	TOE session establishment (FTA_TSE) .....	231
17.8.1	Family Behaviour .....	231
17.8.2	Component levelling and description .....	231
17.8.3	Component management .....	231
17.8.4	Component audit .....	231
17.8.5	Application notes .....	232
17.8.6	FTA_TSE.1 TOE session establishment .....	232
<b>18</b>	<b>Class FTP Trusted path/channels .....</b>	<b>233</b>
18.1	Introduction .....	233
18.2	Notes on class FTP .....	234
18.3	Inter-TSF trusted channel (FTP_ITC) .....	234
18.3.1	Family Behaviour .....	234
18.3.2	Component levelling and description .....	234
18.3.3	Component management .....	235
18.3.4	Component audit .....	235
18.3.5	Application notes .....	235
18.3.6	FTP_ITC.1 Inter-TSF trusted channel .....	235
18.4	Trusted channel protocol (FTP_PRO) .....	236
18.4.1	Family Behaviour .....	236
18.4.2	Component levelling and description .....	236
18.4.3	Component management .....	236
18.4.4	Component audit .....	237
18.4.5	Application notes .....	237
18.4.6	FTP_PRO.1 Trusted channel protocol .....	238
18.4.7	FTP_PRO.2 Trusted channel establishment .....	239
18.4.8	FTP_PRO.3 Trusted channel data protection .....	240
18.5	Trusted path (FTP_TRP) .....	240
18.5.1	Family Behaviour .....	240
18.5.2	Component levelling and description .....	240
18.5.3	Component management .....	241
18.5.4	Component audit .....	241
18.5.5	Application notes .....	241
18.5.6	FTP_TRP.1 Trusted path .....	241
	<b>Bibliography .....</b>	<b>243</b>

**ISO/IEC 15408-2:2026(en)**

## Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

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

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at [www.iso.org/patents](http://www.iso.org/patents) and <https://patents.iec.ch>. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

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

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

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 27, *Information security, cybersecurity and privacy protection*, in collaboration with the European Committee for Standardization (CEN) Technical Committee CEN/CLC/JTC 13, *Cybersecurity and data protection*, in accordance with the Agreement on technical cooperation between ISO and CEN (Vienna Agreement).

This fifth edition cancels and replaces the fourth edition (ISO/IEC 15408-2:2022), which has been technically revised.

The main changes are as follows:

- the document has been restructured;
- technical changes have been introduced:
  - the terminology has been reviewed and updated;
  - the dependency and hierarchy for each component have been reviewed and updated;
  - the new notes for operations have been introduced.

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

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

## Introduction

Security functional components, as defined in this document, are the basis for the security functional requirements (SFRs) or components expressed in a Protection Profile (PP), PP-Module, functional package 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, PP-Module, functional package or an ST. These requirements describe security properties that users can detect by direct interaction (i.e. inputs, outputs) with the Information Technology (IT) or by the IT response to stimulus.

Security functional components allow for the expression of SFRs intended to counter threats in the assumed operating environment of the TOE and cover any identified organizational security policies.

The audience for this document includes consumers, developers and evaluators of secure IT products. ISO/IEC 15408-1 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 use this document as follows:

- consumers, who use this document when selecting components to express functional requirements which satisfy the security objectives expressed in a PP, PP-Module, functional package or ST. ISO/IEC 15408-1 provides more detailed information on the relationship between security objectives and security requirements;
- developers, who respond to actual or perceived consumer security requirements in constructing a TOE and will find a standardized method to understand those requirements in this document. They also use the contents of this document as a basis for further defining the TOE security functionality and mechanisms that conform with those requirements;
- evaluators, who use the SFRs defined in this document in verifying that the TOE functional requirements expressed in the PP, PP-Module, functional package or ST satisfy the IT security objectives and that all dependencies are accounted for and shown to be satisfied. Evaluators use this document to assist in determining whether a given TOE satisfies stated requirements.

This document uses bold type to highlight hierarchical relationships between requirements. This convention calls for the use of bold type for all new requirements.

For security functional requirements, the use of italics denotes assignment and selection items.

Several governmental organizations have contributed to the development of this version of the Common Criteria for Information Technology Security Evaluations. As the joint holders of the copyright in the Common Criteria for Information Technology Security Evaluations (called CC), they hereby grant non-exclusive license to ISO/IEC to use CC in the continued development/maintenance of the ISO/IEC 15408 series of standards. However, these governmental organizations retain the right to use, copy, distribute, translate, or modify CC as they see fit. More information on these agencies can be found at <https://commoncriteriaportal.org/cc/copyright/index.cfm>.

# Information security, cybersecurity and privacy protection — Evaluation criteria for IT security —

## Part 2: Security functional components

### 1 Scope

This document specifies requirements for the required structure and content of security functional components for use during a security evaluation. It includes a catalogue of functional components that 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 15408-1:2026, *Information security, cybersecurity and privacy protection — Evaluation criteria for IT security — Part 1: Introduction and general model*

ISO/IEC 15408-3:2026, *Information security, cybersecurity and privacy protection — Evaluation criteria for IT security — Part 3: Security assurance components*

ISO/IEC 18045:2026, *Information security, cybersecurity and privacy protection — Evaluation criteria for IT security — Requirements and methodology for IT security evaluation*

### 3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 15408-1, ISO/IEC 15408-3, ISO/IEC 18045 and the following apply.

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

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

#### 3.1 identity

representation uniquely identifying an entity within the context of the target of evaluation (TOE)

Note 1 to entry: Entities can be diverse such as a user, process, or disk. For a human user, the representation can be the full or abbreviated name or a unique pseudonym. An entity can have more than one identity.

EXAMPLE An example of such a representation is a string.

#### 3.2 inter-target of evaluation security functionality transfer inter-TSF transfer

communication between the target of evaluation (TOE) and the security functionality of other trusted IT products

**ISO/IEC 15408-2:2026(en)****3.3****internal communication channel**

communication channel between separated parts of the target of evaluation (TOE)

**3.4****internal TOE transfer**

communicating data between separated parts of the target of evaluation (TOE)

**3.5****operation**

(on an ISO/IEC 15408-2 component) modification or repetition of a component by assignment, iteration, refinement, or selection

**3.6****secret**

information that is known only to authorized users and/or the TOE security functionality (TSF) in order to enforce a specific *security function policy* ([3.8](#))

**3.7****secure state**

state in which the TOE security functionality (TSF) data are consistent and the TSF continues correct enforcement of the security functional requirements (SFRs)

**3.8****security function policy****SFP**

set of rules describing specific security behaviour enforced by the TOE security functionality (TSF) and expressible as a set of security functional requirements (SFRs)

**3.9****TOE resource**

anything usable or consumable in the target of evaluation (TOE)

**3.10****transfer outside of the TOE**

target of evaluation (TOE) security functionality (TSF)-mediated communication of data to entities not under the control of the TSF

**3.11****trusted channel**

means by which a target of evaluation (TOE) security functionality (TSF) and another trusted IT product can communicate with necessary confidence

**3.12****trusted path**

means by which a user and a target of evaluation (TOE) security functionality (TSF) can communicate with the necessary confidence

Note 1 to entry: Communication typically implies the establishment of identification and authentication of both parties, as well as the concept of a user specific session which is integrity-protected.

Note 2 to entry: When the external entity is a trusted IT product, the notion of *trusted channel* ([3.11](#)) is used instead of trusted path.

Note 3 to entry: Both physical and logical aspects of secure communication can be considered as mechanisms for gaining confidence.

**3.13****TSF data**

data for the operation of the target of evaluation (TOE) upon which the enforcement of the security functional requirement (SFR) relies

**ISO/IEC 15408-2:2026(en)****3.14****user data**

data received or produced by the target of evaluation (TOE), which is meaningful to some external entity, but which do not affect the operation of the TOE security functionality (TSF)

Note 1 to entry: Depending on the concept, this definition assumes that the same data created by users that has an actual impact on the operation of the TSF can be regarded as the *TSF data* ([3.13](#)).

**4 Abbreviated terms**

API	application programming interface
CD	compact disk
DAC	discretionary access control
DEMA	differential electromagnetic analysis
DPA	differential power analysis
DRBG	deterministic random bit generator
EMS	electromagnetic spectrum
GB	gigabyte
GHz	gigahertz
GUI	graphical user interface
HSM	hardware security module
HTTPS	hypertext transfer protocol secure
IOCTL	input output control
IP	internet protocol
IPsec	IP security (protocol)
IT	information technology
LDAP	lightweight directory access protocol
MAC	mandatory access control
MB	megabyte
MBps	megabytes per second
OS	operating system
OTP	one-time programmable
PC	personal computer
PCI	peripheral component interconnect
PKI	public key infrastructure
PP	protection profile

**ISO/IEC 15408-2:2026(en)**

RAM	random access memory
RBG	random bit generator
RNG	random number generator
RPC	remote procedure call
SEMA	simple electromagnetic analysis
SPA	simple power analysis
SFP	security function policy
SFR	security functional requirement
SSH	secure shell
ST	security target
TCP	transmission control protocol
TLS	transport layer security
TOE	target of evaluation
TSF	TOE security functionality
TSFI	TSF interface
USB	universal serial bus
VPN	virtual private network

## 5 Overview

### 5.1 General

The ISO/IEC 15408 series and the associated security functional requirements (SFRs) 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.

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

Since the understanding and needs of consumers can change, it is expected that the functional components in this document will need to be maintained. It is envisioned that some authors of PPs, PP-Modules, functional packages and STs can have security needs not covered by the security functional components in this document. In those cases, the author of a PP, PP-Module, functional package or ST may choose to consider using functional components and requirements that are not given in this document. The concepts of extensibility are explained in ISO/IEC 15408-1:2026, 8.4.

### 5.2 Organization of this document

[Clause 6](#) describes the paradigm used in the SFRs of this document.

[Clause 7](#) introduces the catalogue of functional components, followed by their definition, from class FAU (Security audit) ([Clause 8](#)) through class FTP (Trusted path/channels) ([Clause 18](#)).

## ISO/IEC 15408-2:2026(en)

Those who author PPs, PP-Modules, functional packages, or STs shall refer to ISO/IEC 15408-1:2026 for relevant structures, rules, and guidance, in particular:

- ISO/IEC 15408-1:2026, Clause 7, which describes how SFRs can be specified using the security functional components;
- ISO/IEC 15408-1:2026, Clause 8, which describes how security functional components are organized, and the operations that may be applied to them;
- ISO/IEC 15408-1:2026, Clause 9, which provides further information on the structure for security functional packages;
- ISO/IEC 15408-1:2026, Annex B, which provides further information on the structure for PPs;
- ISO/IEC 15408-1:2026, Annex C, which provides further information on the structure of PP-Modules and PP-Configurations;
- ISO/IEC 15408-1:2026, Annex D, which provides further information on the structure for STs.

## 6 Functional requirements paradigm

This clause describes the paradigm used in the security functional components and the derivation of SFRs.

This document is a catalogue of security functional components that may be used for the specification of SFRs describing a TOE.

TOE evaluation is concerned primarily with ensuring that a defined set of 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 enforces. Each SFP specifies its scope of control, by defining the subjects, objects, resources or information, and operations to which it applies. All SFPs are implemented by the TOE Security Functionality (TSF), whose mechanisms enforce the rules defined in the SFRs and provide necessary capabilities.

Those portions of a TOE that are relied upon for the correct enforcement of the SFRs are collectively referred to as the 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 can 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 can be localized to the particular TOE, or they can allow interaction with other IT products over external communication channels. These external interactions with other IT products may take two forms:

- 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;
- the other IT product is not 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

**ISO/IEC 15408-2:2026(en)**

information in this situation are called transfers outside of the TOE, as there is either no TSF, or its policy characteristics are unknown, on the other IT product.

The set of interfaces, whether interactive (human-machine interface) or programmatic [application programming interface (API)], 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 either the rights or privileges, or both that are 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 (Security management roles) (13.10)] 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 can 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 can 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 can 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. This document addresses the enforcement of the SFRs over types of subjects.

Several types of subjects can exist within a TOE:

- those acting on behalf of an authorized user: Unix-like processes.
- those acting as a specific functional process that can in turn act on behalf of multiple users: functions as can be found in client/server architectures.
- those acting as part of the TOE itself: processes not acting on behalf of a user.

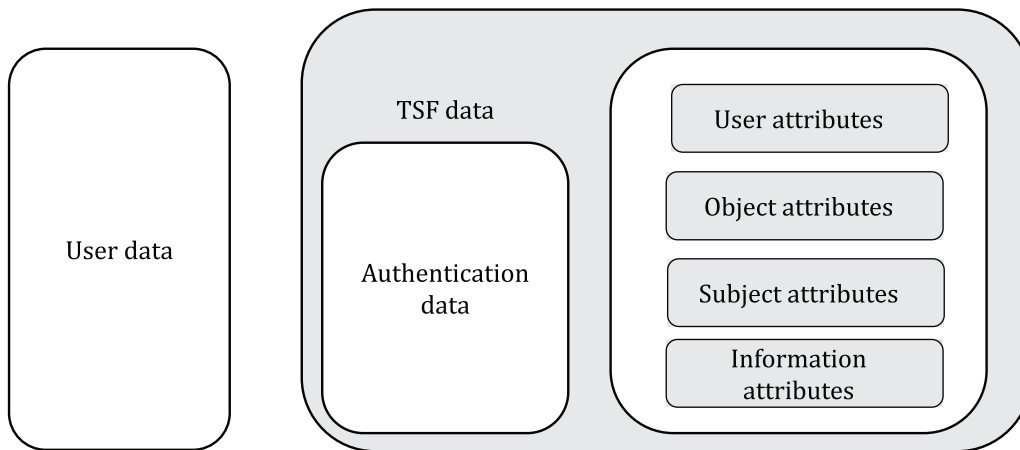
Passive entities in the TOE that contain or receive information and upon which subjects perform operations are called objects. In the case where a subject (an active entity) is the target of an operation, a subject may also be acted on as an object.

**EXAMPLE 5** An example of an operation is an inter-process communication.

## ISO/IEC 15408-2:2026(en)

Objects can contain information. This concept is required to specify information flow control policies as addressed in class FDP (User data protection) ([Clause 11](#)).

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, can exist specifically for the enforcement of the SFRs. These latter attributes are generally referred to as "security attributes". The word attribute is 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 can be necessary to have controls on attributes as dictated by the SFRs.



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

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 6 User data:

- the content of an electronic mail message can be user data.

EXAMPLE 7 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.

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 are required to be kept secret.

## ISO/IEC 15408-2:2026(en)

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 8 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 also intended to be applicable to other types of data that are required to be kept secret in order to enforce a specific SFP.

Therefore, some, but not all, authentication data are required 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 sections are indicated.

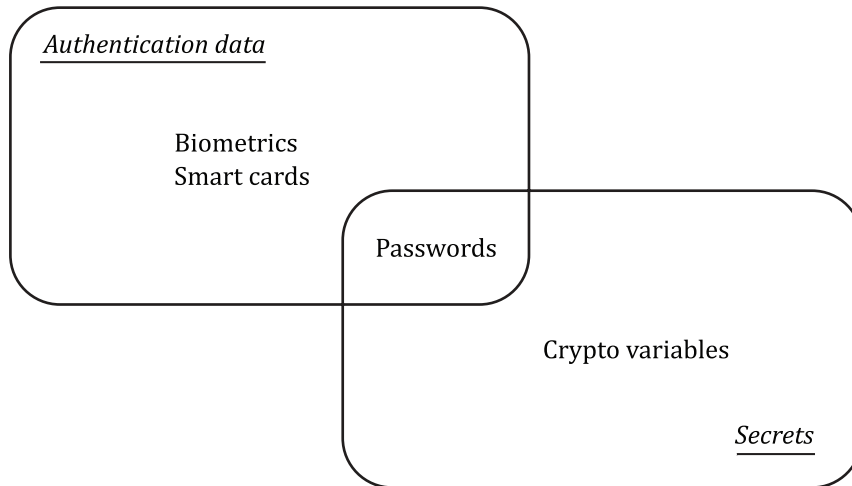


Figure 2 — Relationship between authentication data and secrets

## 7 Security functional components

### 7.1 Overview

This clause describes the constructs used in representing the functional classes, families, and components that define the security functional requirements (SFRs) defined in this document.

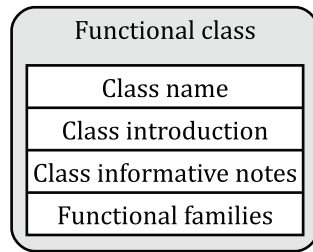
Most abstract collection of SFRs is referred to as a class. Each class contains functional families, which then contain functional components, which in turn contain functional elements. Classes and families are used to provide a taxonomy for classifying SFRs, while components are used to specify SFRs in a PP, PP-Module, functional package or ST.

### 7.2 Functional class structure

#### 7.2.1 General

[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.

## ISO/IEC 15408-2:2026(en)



**Figure 3 — Functional class structure**

### 7.2.2 Class name

The class name 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.

### 7.2.3 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 [7.6](#).

The class introduction includes a class decomposition diagram, showing the families and components of the class. It also provides a dependency table that shows the hierarchical, direct, indirect, and optional dependencies among functional components, as explained in [7.4.3](#).

### 7.2.4 Class informative notes

The informative notes contain additional information that supports the interpretation and understanding of the concepts, structure and usage of the functional class.

### 7.2.5 Functional families

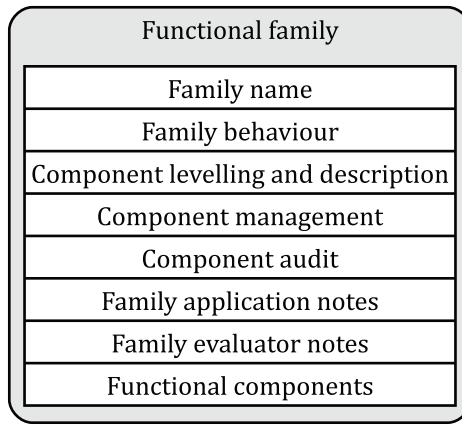
Each functional class contains at least one functional family. The structure of the functional families is described in the following subclause.

## 7.3 Functional family structure

### 7.3.1 General

[Figure 4](#) illustrates the functional family structure in diagrammatic form.

## ISO/IEC 15408-2:2026(en)



**Figure 4 — Functional family structure**

### 7.3.2 Family name

The family name 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.

### 7.3.3 Family Behaviour

The family behaviour subclause provides 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:

- 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;
- 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.

### 7.3.4 Component levelling and description

Functional families contain one or more components, any one of which may be selected for inclusion in STs, PPs, PP-Modules or security functional packages. The goal of the component levelling and description 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.

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 be hierarchical. A component is hierarchical to another if it offers more security.

As explained in [7.6](#), the descriptions of the families provide a graphical overview of the hierarchy of the components in a family.

### 7.3.5 Component management

The management subclauses (for example, [8.3.3](#)) contain information for ST, PP, PP-Module, or security functional package authors to consider as management activities for a given component. The clauses

## ISO/IEC 15408-2:2026(en)

reference components of class FMT (Security management) ([Clause 13](#)) 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.

### 7.3.6 Component audit

The audit requirements subclauses (for example, [8.3.4](#)) contain auditable events for the authors to select, if requirements from class FAU (Security audit) ([Clause 8](#)) 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 FAU\_GEN (Security audit data generation) ([8.4](#)) family.

EXAMPLE 1 An audit note can include actions that are:

- 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.

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

EXAMPLE 2 When Basic Audit Generation is needed, all auditable events identified as being both Minimal and Basic are included in the PP, PP-Module, functional package or 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 needed, all identified auditable events (Minimal, Basic and Detailed) are included in the PP, PP-Module, functional package or ST.

In class FAU (Security audit) ([Clause 8](#)) the rules governing the audit are explained in more detail.

### 7.3.7 Family application notes

The application notes contain additional information that is of interest to potential users of the family, that is PP, PP-Module, ST and functional package authors, and developers of TOEs incorporating the functional components. The presentation is informative and can cover warnings about limitations of use and areas where specific attention can be required when using the components.

NOTE The term PP, PP-Module, functional package or ST author includes authors of documents used to formulate a PP or ST. This also includes authors of PP-Modules and functional packages.

### 7.3.8 Family evaluator notes

The evaluator notes contain any information that is of interest to developers and evaluators of TOEs that claim conformance with a component of the family. The presentation provides information and can cover a variety of areas where specific attention can 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.

The family application and evaluator notes are not required and appear only if appropriate.

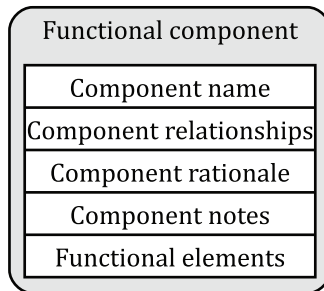
### 7.3.9 Functional components

Each functional family has at least one functional component. The structure of the functional components is provided in the following subclause.

## 7.4 Functional component structure

### 7.4.1 General

[Figure 5](#) illustrates the functional component structure.



**Figure 5 — Functional component structure**

### 7.4.2 Component name

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

- a unique name. The name reflects the purpose of the component;
- a unique short name. A unique short form of the functional component name. This short name serves as the principal reference name for the categorization, registration, and cross-referencing of the component. This short name reflects the class and family to which the component belongs and the component number within the family.

### 7.4.3 Component relationships

These subclauses (for example, [8.3.6.1](#)) define the relationships of each component with the rest of components defined in this document:

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

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

Each functional component provides a complete list of dependencies to other functional and assurance components. Some components may list "No dependencies". The components depended upon may in turn have dependencies on other components. The list provided in the components will be the direct dependencies, i.e. only references to the other functional components that are required for this component to perform its job properly. The indirect dependencies, i.e. the dependencies that result from the depended upon components, can be found in the tables included in the functional class introduction. It is noted that in some cases the dependency is optional in that a number of functional components are provided, where each one of them would be sufficient to satisfy the dependency.

Some dependencies are optional and are indicated as such.

**EXAMPLE** An example of a component with optional dependencies is FDP\_ETC.1 (Export of user data without security attributes) ([11.6.6](#)), which requires either FDP\_ACC.1 (Subset access control) ([11.3.6](#)) or FDP\_IFC.1 (Subset information flow control) ([11.7.6](#)) to be present. So, if FDP\_ACC.1 (Subset access control) ([11.3.6](#)) is present, FDP\_IFC.1 (Subset information flow control) ([11.7.6](#)) is not necessary and vice versa.

## ISO/IEC 15408-2:2026(en)

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

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

Where a dependency is given for security assurance requirements, ISO/IEC 15408-3 shall be referred to.

The dependencies of functional components are shown in a table at the class introduction clause, where each of the components that is a dependency of some other functional component is allocated a column. Each functional component is allocated a row. The value in the table cell indicates whether the column label component is a hierarchical requirement (indicated by an "H"), 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. Sets of optional requirements are indicated by using an entry of type O<sup>x</sup> (e.g. O<sup>1</sup>) whereby an SFR with such entry requires at minimum only one of the SFRs contained in that O<sup>x</sup>-set.

### 7.4.4 Component rationale

Any specific information related to the component is found in the component rationale subclause.

The component rationale contains information that refines the general statements on rationale for the specific component level and is used if level-specific amplification is needed.

The component rationale subclause appears only if appropriate.

### 7.4.5 Component notes

The component notes contain additional refinement in the form of narrative qualifications for a specific component.

This refinement may pertain to application notes and/or evaluator notes.

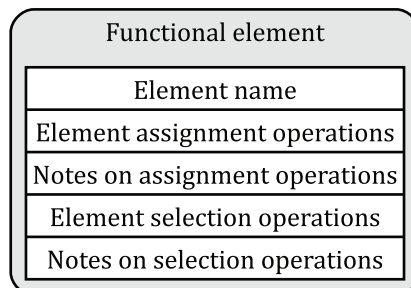
The component application and evaluator notes subclauses appear only if appropriate.

### 7.4.6 Functional elements

A set of functional elements is provided for each functional component. A functional element is a security requirement which, if further divided, would not yield a meaningful expression of a functional requirement. It is the smallest security requirement recognized in the ISO/IEC 15408 series.

## 7.5 Functional elements

[Figure 6](#) illustrates the functional element structure.



**Figure 6 — Functional element structure**

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

**ISO/IEC 15408-2:2026(en)**

The content of the functional element may include assignment and/or selection operations. Where considered useful, these have been complemented by notes on the text.

To facilitate precise identification, assignments are denoted using parentheses superscript notation (e.g., (a<sup>1</sup>), (a<sup>2</sup>), (a<sup>3</sup>)), and selections are denoted in the same manner (e.g., (s<sup>1</sup>), (s<sup>2</sup>), (s<sup>3</sup>)). This notation is consistently applied within elements to reference specific assignments and selections in the notes.

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

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

EXAMPLE The element 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.

## 7.6 Component catalogue

### 7.6.1 General

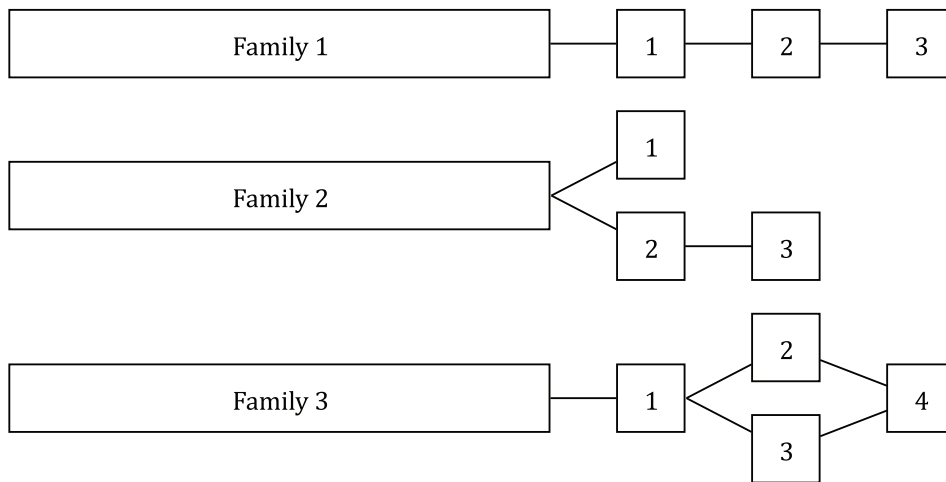
The grouping of the components in this document does not reflect any formal taxonomy.

This document contains classes of families and components, which are rough groupings on the basis of related function or purpose, presented in alphabetical order. At the start of each class is an informative figure that indicates the taxonomy of each class, indicating the families in each class and the components in each family. [Figure 7](#) is a useful indicator of the hierarchical relationship that can 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 similarly to [Figure 7](#) is provided. In [Figure 7](#), 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.

## ISO/IEC 15408-2:2026(en)



**Key**  
1,2,3,4 components

**Figure 7 — 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 provided in each class 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.

### 7.6.2 Highlighting of component changes

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.

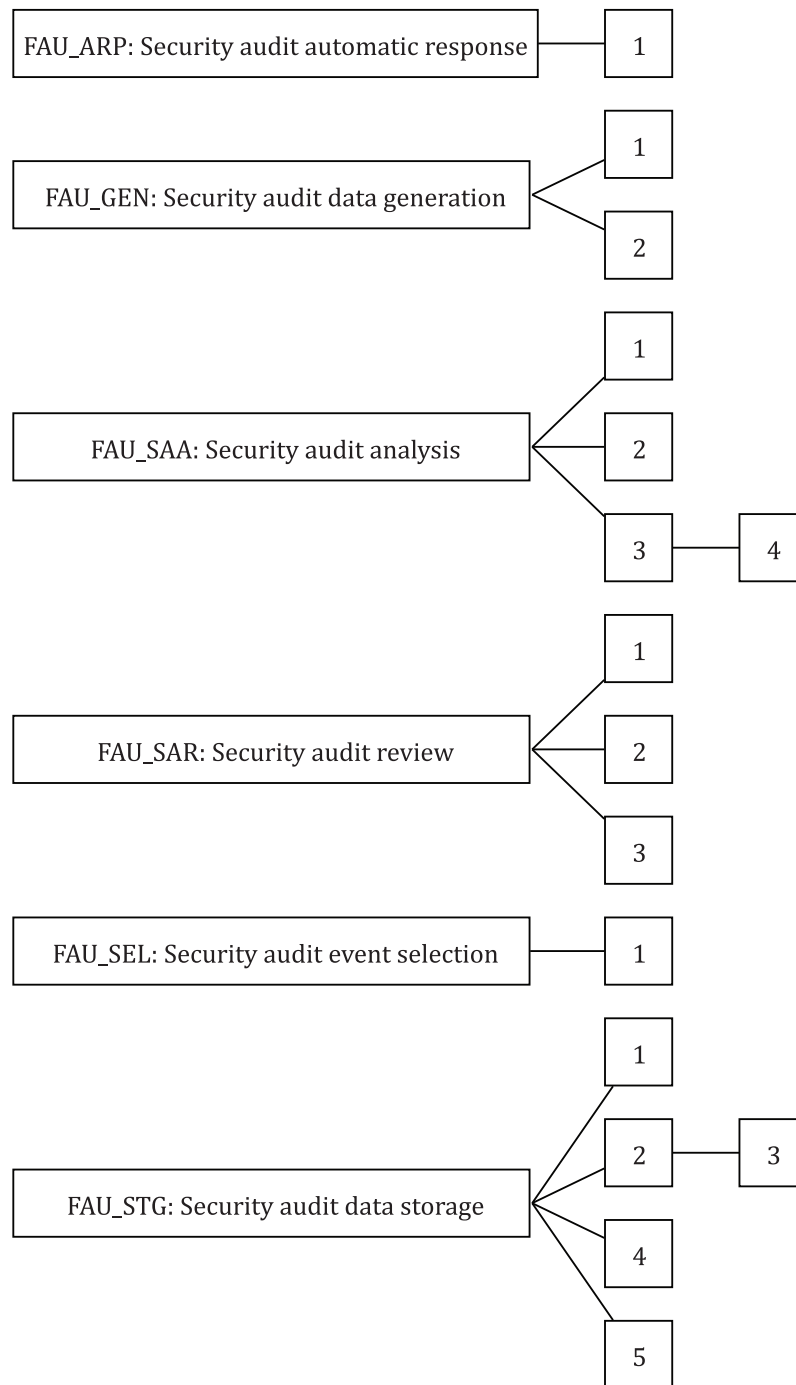
## 8 Class FAU Security audit

### 8.1 Introduction

Security auditing involves recognizing, recording, storing, and analysing 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 8](#) describes the family hierarchy of class FAU.

**ISO/IEC 15408-2:2026(en)**



**Key**  
1,2,3,4,5 components

**Figure 8 — FAU: Security audit, class decomposition**

Direct, indirect and optional dependencies for the components of class FAU are shown in [Table 1](#).

**ISO/IEC 15408-2:2026(en)**

**Table 1 — Dependency table for class FAU: Security audit**

	FAU_GEN.1 (8.4.7)	FAU_SAA.1 (8.5.6)	FAU_SAR.1 (8.6.6)	FAU_STG.2 (8.8.7)	FIA_UID.1 (12.8.6)	FMT_MTD.1 (13.6.6)	FPT_STM.1 (15.14.6)	FTP_ITC.1 (18.3.6)
FAU_ARP.1 (8.3.6)	-	X					-	
FAU_GEN.1 (8.4.7)							X	
FAU_GEN.2 (8.4.8)	X				X		-	
FAU_SAA.1 (8.5.6)	X						-	
FAU_SAA.2 (8.5.7)					X			
FAU_SAA.3 (8.5.8)								
FAU_SAA.4 (8.5.9)								
FAU_SAR.1 (8.6.6)	X						-	
FAU_SAR.2 (8.6.7)	-		X				-	
FAU_SAR.3 (8.6.8)	-		X				-	
FAU_SEL.1 (8.7.6)	X					X	-	
FAU_STG.1 (8.8.6)	X						-	X
FAU_STG.2 (8.8.7)	X						-	
FAU_STG.3 (8.8.8)	X						-	
FAU_STG.4 (8.8.9)	-			X			-	
FAU_STG.5 (8.8.10)	X			X			-	

**8.2 Notes on class FAU**

**8.2.1 General information about audit requirements**

The audit families allow PP, PP-Module, functional package or 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 may 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.

EXAMPLE 2 An example of indirect presentation is by using audit reduction tools.

While developing the security audit requirements, the author of a PP, PP-Module, functional package or ST should take note of the inter-relationships among the audit families and components. The potential exists to specify a set of audit requirements that conform with the family/component dependencies lists, while at the same time resulting in a deficient audit function.

EXAMPLE 3 An example of a deficient audit function is one 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.

**8.2.2 Audit requirements in a distributed environment**

The implementation of audit requirements for networks and other large systems 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 can be managed, due to the lowered feasibility of interpreting (or even storing) what gets collected. The traditional notion of a time-ordered list, set of

## ISO/IEC 15408-2:2026(en)

records or "trail" of audited events 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 can have differing naming policies and values. Further, the use of symbolic names for audit review 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, are usually required if audit repositories are to serve a useful function in distributed systems.

Finally, misuse of authority by authorized users can be addressed by systematically avoiding local storage of audit data pertaining to administrator actions.

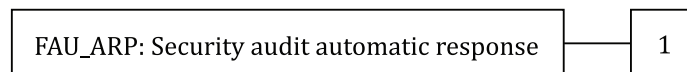
### 8.3 Security audit automatic response (FAU\_ARP)

#### 8.3.1 Family Behaviour

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

#### 8.3.2 Component levelling and description

[Figure 9](#) illustrates the component levelling of family FAU\_ARP.



#### Key

1 component

**Figure 9 — FAU\_ARP: Component levelling**

At FAU\_ARP.1 (Security alarms) ([8.3.6](#)), the TSF shall take actions in case a potential security violation is detected.

#### 8.3.3 Component management

The following action related to FAU\_ARP.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

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

#### 8.3.4 Component audit

The following action related to FAU\_ARP.1 should be auditable if FAU\_GEN (Security audit data generation) ([8.4](#)) is included in the PP, PP-Module, functional package or ST:

- minimal: actions taken due to potential security violations.

#### 8.3.5 Application notes

The security audit automatic response family describes requirements for the handling of audit events. The requirement can include requirements for alarms or TSF action (automatic response).

**EXAMPLE** The TSF can include the generation of real time alarms, termination of the offending process, disabling of a service, or disconnection or invalidation of a user account.

An audit event is defined to be a "potential security violation" when indicated by the FAU\_SAA (Security audit analysis) ([8.5](#)) components.

## ISO/IEC 15408-2:2026(en)

### 8.3.6 FAU\_ARP.1 Security alarms

#### 8.3.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FAU\_SAA.1 (Potential violation analysis) ([8.5.6](#))

#### 8.3.6.2 Component rationale

One or more actions should be taken for follow up action in the event of an alarm.

These actions can include informing the authorized user of the alarm, presenting the authorized user with a set of possible containment actions, or options for the authorized user to take corrective actions.

The timing of the actions should be carefully considered by the PP, PP-Module, functional package or ST author.

#### 8.3.6.3 Element FAU\_ARP.1.1

**The TSF shall take [assignment (a1): *list of actions*] upon detection of a potential security violation.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the actions to be taken in case of a potential security violation.

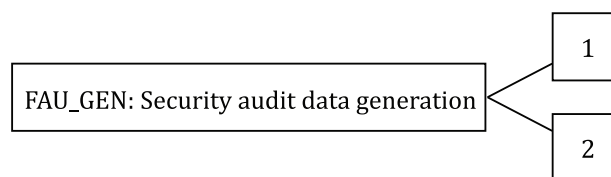
## 8.4 Security audit data generation (FAU\_GEN)

### 8.4.1 Family Behaviour

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

### 8.4.2 Component levelling and description

[Figure 10](#) illustrates the component levelling of family FAU\_GEN.



#### Key

1,2 components

**Figure 10 — FAU\_GEN: Component levelling**

FAU\_GEN.1 (Audit data generation) ([8.4.7](#)) defines the level of auditable events and specifies the list of data that shall be recorded in each record.

In FAU\_GEN.2 (User identity association) ([8.4.8](#)), the TSF shall associate auditable events to individual user identities.

### 8.4.3 Component management

There are no management activities foreseen related to FAU\_GEN.1 or FAU\_GEN.2.

#### 8.4.4 Component audit

There are no auditable events foreseen related to FAU\_GEN.1 or FAU\_GEN.2.

#### 8.4.5 Application notes

The security audit data generation family includes requirements to specify the audit events that shall be generated by the TSF for security-relevant events.

This family is presented in a manner that avoids a dependency on all components requiring audit support. Each component has an audit subclause developed in which the events to be audited for that functional area are listed. When the PP, PP-Module, functional package or ST is written, the items in the audit area are used to complete the variable in these components. Thus, the specification of what can be audited for a functional area is localized in that functional area.

The list of auditable events is entirely dependent on the other functional families within the PP, PP-Module, functional package or ST. Each family definition should therefore include a list of its family-specific auditable events. Each auditable event in the list of auditable events specified in the functional family should correspond to one of the levels of audit event generation specified in this family (i.e. minimal, basic, detailed). This provides the author of a PP, PP-Module, functional package or ST with the information necessary to ensure that all appropriate auditable events are specified in the PP, PP-Module, functional package or ST. The following example shows how auditable events are expected to be specified in appropriate functional families:

**EXAMPLE 1** The following actions are expected to be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- a) minimal: successful use of the user security attribute administration functions;
- b) basic: all attempted uses of the user security attribute administration functions;
- c) basic: identification of which user security attributes have been modified;
- d) detailed: with the exception of specific sensitive attribute data items, the new values of the attributes should be captured.

**NOTE** Sensitive attribute data items include passwords and cryptographic keys.

For each functional component that is chosen, the auditable events that are indicated in that component, at and below the level indicated in FAU\_GEN (Security audit data generation) should be auditable. So, in the previous example "Basic" would be selected in FAU\_GEN (Security audit data generation), the auditable events mentioned in a), b) and c) should be auditable.

Observe that the categorization of auditable events (minimal, basic, detailed) is hierarchical in that order.

This means that:

- when minimal audit generation is desired, all auditable events identified as being minimal should be included in the PP, PP-Module, functional package or ST through the use of the appropriate assignment operation;
- when basic audit generation is desired, all auditable events identified as being either minimal or basic, should also be included in the PP, PP-Module, functional package or 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, PP-Module, functional package or ST.

**ISO/IEC 15408-2:2026(en)**

A PP, PP-Module, functional package or ST author may decide to include other auditable events beyond those required for a given audit level.

**EXAMPLE 2** The PP, PP-Module, functional package or ST can claim only minimal audit capabilities while including most of the basic capabilities because the few excluded capabilities conflict with other PP, PP-Module, functional package or ST constraints (perhaps because they require the collection of unavailable data).

The functionality that creates the auditable event should be specified in the PP, PP-Module, functional package or ST as a functional requirement.

**EXAMPLE 3** The following are examples of the types of the events that can be defined as auditable within each PP, PP-Module, functional package or ST functional component:

- introduction of objects within the control of the TSF into a subject's address space;
- deletion of objects;
- distribution or revocation of access rights or capabilities;
- changes to subject or object security attributes;
- policy checks performed by the TSF as a result of a request by a subject;
- the use of access rights to bypass a policy check;
- use of Identification and Authentication functions;
- actions taken by an operator, and/or authorized user (such as suppression of a TSF protection mechanism as human-readable labels);
- import/export of data from/to removable media (such as printed output, tapes, USB sticks).

#### **8.4.6 Evaluator notes**

FAU\_GEN.1 (Audit data generation) (8.4.7) has a dependency on FPT\_STM.1 (Reliable time stamps) (15.14.6). If correctness of time is not an issue for this TOE, elimination of this dependency can be justified by the author of a PP, PP-Module, functional package or ST.

#### **8.4.7 FAU\_GEN.1 Audit data generation**

##### **8.4.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FPT\_STM.1 (Reliable time stamps) (15.14.6)

##### **8.4.7.2 Component rationale**

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 can be used when the SFRs do not require that individual user identities be associated with audit events. This can be appropriate when the PP, PP-Module, functional package or ST also contains privacy requirements. If the user identity is incorporated FAU\_GEN.2 (User identity association) (8.4.8) can be used in addition to FAU\_GEN.1 (Audit data generation).

If the subject is a user, the user identity can be recorded as the subject identity. If FIA\_UAU (User authentication) (12.7) has not been applied however, the identity of the user will not yet have been verified. Therefore, in the instance of an invalid login the claimed user identity should be recorded. It should also be considered whether to indicate when a recorded identity has not been authenticated.

## ISO/IEC 15408-2:2026(en)

### 8.4.7.3 Element FAU\_GEN.1.1

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

- start-up and shutdown of the audit functions;
- all auditable events for the [selection, choose one of <sup>(s1)</sup>: *minimum, basic, detailed, not specified*] level of audit;
- [assignment <sup>(a1)</sup>: *other specifically defined auditable events*].

In selection (s1), the author of a PP, PP-Module, functional package or ST should select the level of auditable events called out in the audit subclause of other functional components included in the PP, PP-Module, functional package or ST. This level is one of the following: "minimum", "basic", "detailed" or "not specified".

### 8.4.7.4 Element FAU\_GEN.1.2

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

- date and time of the auditable event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event;
- for each auditable event type, based on the auditable event definitions of the functional components included in the PP, PP-Module, functional package or ST, [assignment <sup>(a1)</sup>: *other audit relevant information*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should assign, for each of the auditable events included in the PP, PP-Module, functional package or ST, either a list of other audit relevant information to be included in audit events records or none.

## 8.4.8 FAU\_GEN.2 User identity association

### 8.4.8.1 Component relationships

Hierarchical to: No other components.

Dependencies: FAU\_GEN.1 (Audit data generation) ([8.4.7](#))  
FIA\_UID.1 (Timing of identification) ([12.8.6](#))

### 8.4.8.2 Component rationale

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) ([8.4.7](#)).

There is a potential conflict between the audit and privacy requirements. For audit purposes, it can be desirable to know who performed an action. It is possible that a user wants to keep their actions to themselves and not be identified by other persons such as a site with job offers. Alternatively, it can 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 can contradict each other. Therefore, if this requirement is selected and privacy is important, inclusion of the component user pseudonymity should 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.

### 8.4.8.3 Element FAU\_GEN.2.1

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

## 8.5 Security audit analysis (FAU\_SAA)

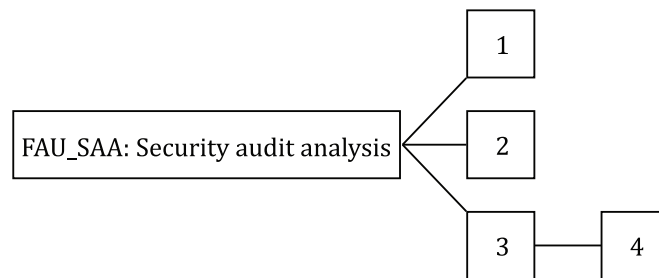
### 8.5.1 Family Behaviour

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

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

### 8.5.2 Component levelling and description

[Figure 11](#) describes the component levelling of family FAU\_SAA.



#### Key

1,2,3,4 components

**Figure 11 — FAU\_SAA: Component levelling**

In FAU\_SAA.1 (Potential violation analysis) ([8.5.6](#)), basic threshold detection on the basis of a fixed rule set is required.

In FAU\_SAA.2 (Profile based anomaly detection) ([8.5.7](#)), the TSF maintains individual profiles of system usage, where a profile represents the historical patterns of usage performed by members of the profile target group. A profile target group refers to a group of one or more individuals who interact with the TSF. Each member of a profile target group is assigned an individual suspicion rating that represents how well that member's current activity corresponds to the established patterns of usage represented in the profile. This analysis can be performed at runtime or during a post-collection batch-mode analysis.

In FAU\_SAA.3 (Simple attack heuristics) ([8.5.8](#)), the TSF shall be able to detect the occurrence of signature events that represent a significant threat to enforcement of the SFRs. This search for signature events may occur in real-time or during a post-collection batch-mode analysis.

In FAU\_SAA.4 (Complex attack heuristics) ([8.5.9](#)), the TSF shall be able to represent and detect multi-step intrusion scenarios. The TSF is able to compare system events (possibly performed by multiple individuals) against event sequences known to represent entire intrusion scenarios. The TSF shall be able to indicate when a signature event or event sequence is found that indicates a potential violation of the enforcement of the SFRs.

### 8.5.3 Component management

The following action related to FAU\_SAA.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- maintenance of the rules by (adding, modifying, deleting) rules from the set of rules.

The following action related to FAU\_SAA.2 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- maintenance (deletion, modification, addition) of the group of users in the profile target group.

**ISO/IEC 15408-2:2026(en)**

The following action related to FAU\_SAA.3 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- maintenance (deletion, modification, addition) of the subset of system events.

The following actions related to FAU\_SAA.4 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- maintenance (deletion, modification, addition) of the subset of system events;
- maintenance (deletion, modification, addition) of the set of sequences of system events.

**8.5.4 Component audit**

The following actions related to FAU\_SAA.1, FAU\_SAA.2, FAU\_SAA.3 and FAU\_SAA.4 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: enabling and disabling of any of the analysis mechanisms;
- minimal: automated responses performed by the tool.

**8.5.5 Application notes**

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

The action to be performed by the TSF on detection of a potential violation is defined in FAU\_ARP (Security audit automatic response) components.

For real-time analysis, audit data can be transformed into a useful format for automated treatment, but into a different useful format for delivery to authorized users for review.

**8.5.6 FAU\_SAA.1 Potential violation analysis****8.5.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FAU\_GEN.1 (Audit data generation) ([8.4.7](#))

**8.5.6.2 Component rationale**

This component is used to specify the set of auditable events whose occurrence or accumulated occurrence is held to indicate a potential violation of the enforcement of the SFRs, and any rules to be used to perform the violation analysis.

**8.5.6.3 Element FAU\_SAA.1.1**

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

**8.5.6.4 Element FAU\_SAA.1.2**

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

- **accumulation or combination of [assignment <sup>(a1)</sup>: *subset of defined auditable events*] known to indicate a potential security violation;**
- **[assignment <sup>(a2)</sup>: *any other rules*].**

## ISO/IEC 15408-2:2026(en)

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify the subset of defined auditable events whose occurrence or accumulated occurrence is detected as an indication of a potential violation of the enforcement of the SFRs.

### 8.5.7 FAU\_SAA.2 Profile based anomaly detection

#### 8.5.7.1 Component relationships

Hierarchical to: No other components.

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

#### 8.5.7.2 Component rationale

A profile is a structure that characterizes the behaviour of users and/or subjects; it represents how the users/subjects interact with the TSF in a variety of ways. Patterns of usage are established with respect to the various types of activity the users/subjects engage in. The ways in which the various types of activity are recorded in the profile are referred to as profile metrics.

EXAMPLE 1 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.

Each profile represents the expected patterns of usage performed by members of the profile target group. This pattern may be based on past use (historical patterns) or on normal use for users of similar target groups (expected behaviour). A profile target group refers to one or more users who interact with the TSF. The activity of each member of the profile group is used by the analysis tool in establishing the usage patterns represented in the profile. The following are some examples of profile target groups:

- single user account: one profile per user;
- group ID or group account: one profile for all users who possess the same group ID or operate using the same group account;
- operating role: one profile for all users sharing a given operating role;
- system: one profile for all users of a system.

Each member of a profile target group is assigned an individual suspicion rating that represents how closely that member's new activity corresponds to the established patterns of usage represented in the group profile.

The sophistication of the anomaly detection tool will largely be determined by the number of target profile groups required by the PP, PP-Module, functional package or ST and the complexity of the required profile metrics.

The author of a PP, PP-Module, functional package or ST should enumerate specifically what activity should be monitored and/or analysed by the TSF. The author of a PP, PP-Module, functional package or ST should also identify specifically what information pertaining to the activity is necessary to construct the usage profiles.

FAU\_SAA.2 (Profile based anomaly detection) requires that the TSF maintain profiles of system usage. The word maintain implies that the anomaly detector is actively updating the usage profile based on new activity performed by the profile target members. It is important here that the metrics for representing user activity are defined by the author of a PP, PP-Module, functional package or ST.

EXAMPLE 2 There can be a thousand different actions an individual is capable of performing, but the anomaly detector can choose to monitor a subset of that activity.

**ISO/IEC 15408-2:2026(en)**

Anomalous activity gets integrated into the profile just like non-anomalous activity (assuming the tool is monitoring those actions). Things that may have appeared anomalous four months ago, can over time become the norm (and vice-versa) as the user's work duties change. The TSF would not be able to capture this notion if it filtered out anomalous activity from the profile updating algorithms.

Administrative notification should be provided such that the authorized user understands the significance of the suspicion rating.

The author of a PP, PP-Module, functional package or ST should define how to interpret suspicion ratings and the conditions under which anomalous activity is indicated to the FAU\_ARP (Security audit automatic response) mechanism.

**8.5.7.3 Element FAU\_SAA.2.1**

**The TSF shall be able to maintain profiles of system usage, where an individual profile represents the historical patterns of usage performed by the member(s) of [assignment <sup>(a1)</sup>: *the profile target group*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the profile target group. A single PP, PP-Module, functional package or ST may include multiple profile target groups.

**8.5.7.4 Element FAU\_SAA.2.2**

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

**8.5.7.5 Element FAU\_SAA.2.3**

**The TSF shall be able to indicate a possible violation of the enforcement of the SFRs when a user's suspicion rating exceeds the following threshold conditions [assignment <sup>(a1)</sup>: *conditions under which anomalous activity is reported by the TSF*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify conditions under which anomalous activity is reported by the TSF. Conditions may include the suspicion rating reaching a certain value or be based on the type of anomalous activity observed.

**8.5.8 FAU\_SAA.3 Simple attack heuristics****8.5.8.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**8.5.8.2 Component rationale**

In practice, it is at best rare when an analysis tool can detect with certainty when a security violation is imminent. However, there do exist some system events that are so significant that they are always worthy of independent review.

**EXAMPLE 1** 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.

These events are referred to as signature events in that their occurrence in isolation from the rest of the system activity are indicative of intrusive activity.

The complexity of a given tool will depend greatly on the assignments defined by the author of a PP, PP-Module, functional package or ST in identifying the base set of signature events.

**ISO/IEC 15408-2:2026(en)**

The author of a PP, PP-Module, functional package or ST should enumerate specifically what events should be monitored by the TSF in order to perform the analysis. The author of a PP, PP-Module, functional package or ST should identify specifically what information pertaining to the event is necessary to determine if the event maps to a signature event.

Administrative notification should be provided such that the authorized user understands the 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 2** Examples of other input data include network datagrams, resource/accounting data, or combinations of various system data.

The elements of FAU\_SAA.3 (Simple attack heuristics) do not require that the TSF implementing the immediate attack heuristics be the same TSF whose activity is being monitored. Thus, it is possible to develop an intrusion detection component that operates independently of the system whose system activity is being analysed.

### 8.5.8.3 Element FAU\_SAA.3.1

**The TSF shall be able to maintain an internal representation of the following signature events [assignment <sup>(a1)</sup>: *a subset of system events*] that may indicate a violation of the enforcement of the SFRs.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify a base subset of system events whose occurrence, in isolation from all other system activity, can indicate a violation of the enforcement of the SFRs. These include events that by themselves indicate a clear violation to the enforcement of the SFRs, or whose occurrence is so significant that they warrant actions.

### 8.5.8.4 Element FAU\_SAA.3.2

**The TSF shall be able to compare the signature events against the record of system activity discernible from an examination of [assignment <sup>(a1)</sup>: *the information to be used to determine system activity*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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 author of a PP, PP-Module, functional package or ST should define precisely what system events and event attributes are being monitored within the input data.

### 8.5.8.5 Element FAU\_SAA.3.3

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

## 8.5.9 FAU\_SAA.4 Complex attack heuristics

### 8.5.9.1 Component relationships

Hierarchical to: FAU\_SAA.3 (Simple attack heuristics) ([8.5.8](#))

Dependencies: No dependencies.

**ISO/IEC 15408-2:2026(en)****8.5.9.2 Component rationale**

In practice, it is at best rare when an analysis tool can detect with certainty when a security violation is imminent. However, there do exist some system events that are so significant they are always worthy of independent review.

**EXAMPLE 1** Examples 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.

These events are referred to as signature events in that their occurrence in isolation from the rest of the system activity are indicative of intrusive activity. Event sequences are an ordered set of signature events that can indicate intrusive activity.

The complexity of a given tool will depend greatly on the assignments defined by the author of a PP, PP-Module, functional package or ST in identifying the base set of signature events and event sequences.

The author of a PP, PP-Module, functional package or ST should enumerate specifically what events should be monitored by the TSF in order to perform the analysis. The author of a PP, PP-Module, functional package or ST should identify specifically what information pertaining to the event is necessary to determine if the event maps to a signature event.

Administrative notification should be provided such that the authorized user understands the 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 2** Examples of other input data include network datagrams, resource/accounting data, or combinations of various system data.

Levelling, therefore, requires the author of a PP, PP-Module, functional package or ST 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, it is possible to develop an intrusion detection component that operates independently of the system whose system activity is being analysed.

**8.5.9.3 Element FAU\_SAA.4.1**

The TSF shall be able to maintain an internal representation of the following **event sequences of known intrusion scenarios [assignment (a1): *list of sequences of system events whose occurrence are representative of known penetration scenarios*] and the following** signature events [assignment (a2): *a subset of system events*] that may indicate a **potential** violation of the enforcement of the SFRs.

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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 assignment (a2), the author of a PP, PP-Module, functional package or ST 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.

## ISO/IEC 15408-2:2026(en)

### 8.5.9.4 Element FAU\_SAA.4.2

The TSF shall be able to compare the signature events **and event sequences** against the record of system activity discernible from an examination of [assignment <sup>(a1)</sup>: *the information to be used to determine system activity*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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 author of a PP, PP-Module, functional package or ST should define precisely what system events and event attributes are being monitored within the input data.

### 8.5.9.5 Element FAU\_SAA.4.3

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

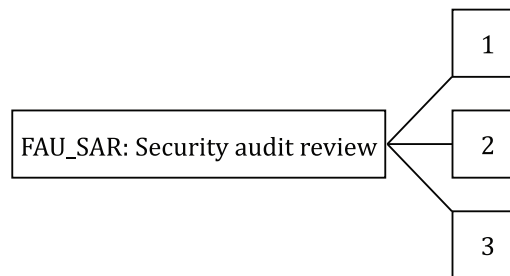
## 8.6 Security audit review (FAU\_SAR)

### 8.6.1 Family Behaviour

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

### 8.6.2 Component levelling and description

[Figure 12](#) describes the component levelling of family FAU\_SAR.



#### Key

1,2,3 components

**Figure 12 — FAU\_SAR: Component levelling**

FAU\_SAR.1 (Audit review) ([8.6.6](#)) provides the capability to read information from the audit data.

FAU\_SAR.2 (Restricted audit review) ([8.6.7](#)) requires that there are no other users except those that have been identified in FAU\_SAR.1 (Audit review) ([8.6.6](#)) that can read the information.

FAU\_SAR.3 (Selectable audit review) ([8.6.8](#)) requires audit review tools to select the audit data to be reviewed based on criteria.

### 8.6.3 Component management

The following action related to FAU\_SAR.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- maintenance (deletion, modification, addition) of the group of users with read access right to the audit records.

## ISO/IEC 15408-2:2026(en)

There are no management activities foreseen related to FAU\_SAR.2 or FAU\_SAR.3.

### 8.6.4 Component audit

The following action related to FAU\_SAR.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

— basic: reading of information from the audit records.

The following action related to FAU\_SAR.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

— basic: unsuccessful attempts to read information from the audit records.

The following action related to FAU\_SAR.3 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

— detailed: the parameters used for the viewing.

### 8.6.5 Application 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 "or") relations and order the audit data before it is reviewed.

### 8.6.6 FAU\_SAR.1 Audit review

#### 8.6.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FAU\_GEN.1 (Audit data generation) ([8.4.7](#))

#### 8.6.6.2 Component rationale

This component provides authorized users the capability to obtain and interpret the information. In the case of human users, this information is presented in a human understandable form. In the case of external IT entities, the information is unambiguously represented in an electronic fashion.

This component is also 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 can have different needs.

The content of the audit records that can be viewed can be specified.

**ISO/IEC 15408-2:2026(en)****8.6.6.3 Element FAU\_SAR.1.1**

**The TSF shall provide [assignment (a1): *authorized users*] with the capability to read [assignment (a2): *list of audit information*] from the audit data.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the authorized users that can use this capability. If appropriate the author of a PP, PP-Module, functional package or ST may include security roles (see FMT\_SMR.1 (Security roles) ([13.10.6](#))).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the type of information the specified user is permitted to obtain from the audit records.

**8.6.6.4 Element FAU\_SAR.1.2**

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

**8.6.7 FAU\_SAR.2 Restricted audit review****8.6.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FAU\_SAR.1 (Audit review) ([8.6.6](#))

**8.6.7.2 Component rationale**

This component specifies that any users not identified in FAU\_SAR.1 (Audit review) ([8.6.6](#)) will not be able to read the audit records.

**8.6.7.3 Element FAU\_SAR.2.1**

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

**8.6.8 FAU\_SAR.3 Selectable audit review****8.6.8.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FAU\_SAR.1 (Audit review) ([8.6.6](#))

**8.6.8.2 Component rationale**

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.

**8.6.8.3 Element FAU\_SAR.3.1**

**The TSF shall provide the ability to apply [assignment (a1): *methods of selection and/or ordering*] of audit data based on [assignment (a2): *criteria with logical relations*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify whether capabilities to select and/or order audit data is required from the TSF.

## ISO/IEC 15408-2:2026(en)

In assignment (a2), the author of a PP, PP-Module, functional package or ST should assign the criteria, possibly with logical relations, to be used to select the audit data for review. The logical relations are intended to specify whether the operation can be on an individual attribute or a collection of attributes.

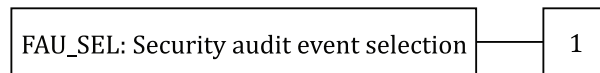
### 8.7 Security audit event selection (FAU\_SEL)

#### 8.7.1 Family Behaviour

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

#### 8.7.2 Component levelling and description

[Figure 13](#) describes the component levelling of family FAU\_SEL.



#### Key

1 component

**Figure 13 — FAU\_SEL: Component levelling**

FAU\_SEL.1 (Selective audit) ([8.7.6](#)) requires the ability to select the set of events to be audited from the set of all auditable events, identified in FAU\_GEN.1 (Audit data generation) ([8.4.7](#)), based upon attributes to be specified by the author of a PP, PP-Module, functional package or ST.

#### 8.7.3 Component management

The following action related to FAU\_SEL.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- maintenance of the rights to view/modify the audit data.

#### 8.7.4 Component audit

The following action related to FAU\_SEL.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: all modifications to the audit configuration that occur while the audit collection functions are operating.

#### 8.7.5 Application notes

The security audit event selection family provides requirements related to the capabilities of identifying which of the possible auditable events are expected to be audited. The auditable events are defined in the FAU\_GEN (Security audit data generation) family, but those events should be defined as being selectable in this component to be audited.

This family ensures that it is possible to keep the audit trail from becoming so large that it becomes useless, by defining the appropriate granularity of the selected security audit events.

**ISO/IEC 15408-2:2026(en)****8.7.6 FAU\_SEL.1 Selective audit****8.7.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FAU\_GEN.1 (Audit data generation) ([8.4.7](#))  
FMT\_MTD.1 (Management of TSF data) ([13.6.6](#))

**8.7.6.2 Component rationale**

This component defines the selection criteria used, and the resulting audited subsets of the set of all auditable events, based on user attributes, subject attributes, object attributes, or event types.

The existence of individual user identities is not assumed for this component. This allows for TOEs such as routers that do not support the notion of users.

For a distributed environment, the host identity can be used as a selection criterion for events to be audited.

The management function FMT\_MTD.1 (Management of TSF data) ([13.6.6](#)) will handle the rights of authorized users to query or modify the selections.

**8.7.6.3 Element FAU\_SEL.1.1**

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

- [selection <sup>(s1)</sup>: *object identity, user identity, subject identity, host identity, event type*]
- [assignment <sup>(a1)</sup>: *list of additional attributes that audit selectivity is based upon*]

In selection (s1), the author of a PP, PP-Module, functional package or ST should select whether the security attributes upon which audit selectivity is based, is related to object identity, user identity, subject identity, host identity, or event type.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify any additional attributes upon which audit selectivity is based. If there are no additional rules upon which audit selectivity is based, this assignment can be completed with "none".

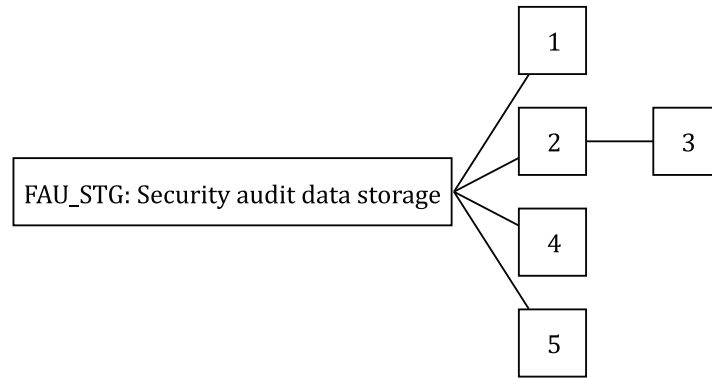
**8.8 Security audit data storage (FAU\_STG)****8.8.1 Family Behaviour**

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.

**8.8.2 Component levelling and description**

[Figure 14](#) describes the component levelling of family FAU\_STG.

## ISO/IEC 15408-2:2026(en)



### Key

1,2,3,4,5 components

**Figure 14 — FAU\_STG: Component levelling**

FAU\_STG.1 (Audit data storage location) (8.8.6) requires that the storage location(s) for audit data be specified.

FAU\_STG.2 (Protected audit data storage) (8.8.7) 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) (8.8.8) specifies the guarantees that the TSF maintains over the audit data given the occurrence of an undesired condition.

FAU\_STG.4 (Action in case of possible audit data loss) (8.8.9) specifies actions to be taken if a threshold on the stored audit data is exceeded.

FAU\_STG.5 (Prevention of audit data loss) (8.8.10) specifies actions to be taken in the case that audit data storage is full.

### 8.8.3 Component management

The following action related to FAU\_STG.1 can be considered for the management functions in class FMT (Security management) (Clause 13):

- maintenance of remote audit storage locations.

The following action related to FAU\_STG.3 can be considered for the management functions in class FMT (Security management) (Clause 13):

- maintenance of the parameters that control the audit data storage capability.

The following action related to FAU\_STG.4 can be considered for the management functions in class FMT (Security management) (Clause 13):

- maintenance (deletion, modification, addition) of actions to be taken in case of imminent audit data storage failure.

The following action related to FAU\_STG.5 can be considered for the management functions in class FMT (Security management) (Clause 13):

- maintenance (deletion, modification, addition) of actions to be taken in case of audit data storage failure.

There are no management activities foreseen related to FAU\_STG.2.

**ISO/IEC 15408-2:2026(en)****8.8.4 Component audit**

The following action related to FAU\_STG.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- basic: changes in the location of remote audit data storage.

The following action related to FAU\_STG.4 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- basic: actions taken due to exceeding of a threshold.

The following action related to FAU\_STG.5 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- basic: actions taken due to the audit data storage failure.

There are no auditable events foreseen related to FAU\_STG.2 or FAU\_STG.3.

**8.8.5 Application notes**

The security audit data storage family describes requirements for storing audit data for later use, including requirements controlling the loss of audit information due to TOE failure, attack and/or exhaustion of storage space.

**8.8.6 FAU\_STG.1 Audit data storage location****8.8.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FAU\_GEN.1 (Audit data generation) ([8.4.7](#))  
FTP\_ITC.1 (Inter-TSF trusted channel) ([18.3.6](#))

**8.8.6.2 Component rationale**

In a distributed environment, as the location of the audit trail is in the TSF, but not necessarily co-located with the function generating the audit data, the author of a PP, PP-Module, functional package or ST can request authentication of the originator of the audit record, or non-repudiation of the origin of the record prior to storing this record in the audit trail.

The TSF will protect the stored audit records in the audit trail from unauthorised deletion and modification. It is noted that in some TOEs the auditor (role) cannot be authorized to delete the audit records for a certain period of time.

FAU\_STG.1 (Audit data storage location) is dependent upon FTP\_ITC.1 (Inter-TSF trusted channel) ([18.3.6](#)), if "transmit the generated audit data to an external IT entity using a trusted channel according to FTP\_ITC" is not selected then the author of a PP, PP-Module, functional package or ST can satisfy the dependency by providing the rationale explaining why it was not selected.

**8.8.6.3 Element FAU\_STG.1.1**

**The TSF shall be able to store generated audit data on the [selection (s<sup>1</sup>): *TOE itself, transmit the generated audit data to an external IT entity using a trusted channel according to FTP\_ITC, [assignment (a<sup>1</sup>): other storage location(s)].***

In assignment (a1), if additional or alternative storage locations for audit data need to be specified by the author of a PP, PP-Module, functional package or ST then this requirement can be specified in FAU\_STG.1.1 using the assignment found within the selection.

**ISO/IEC 15408-2:2026(en)**

In selection (s1), the author of a PP, PP-Module, functional package or ST should select where the audit data is stored. Audit data may be stored on the TOE itself, be transmitted to an external entity using a trusted channel, or other storage options can be specified in the assignment.

**8.8.7 FAU\_STG.2 Protected audit data storage****8.8.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FAU\_GEN.1 (Audit data generation) ([8.4.7](#))

**8.8.7.2 Component rationale**

In a distributed environment, as the location of the audit trail is in the TSF, but not necessarily co-located with the function generating the audit data, the author of a PP, PP-Module, functional package or ST can request authentication of the originator of the audit record, or non-repudiation of the origin of the record prior storing this record in the audit trail.

The TSF will protect the stored audit data in the audit trail from unauthorized deletion and modification. It is noted that in some TOEs it is possible that the auditor (role) is not authorized to delete the audit records for a certain period of time.

**8.8.7.3 Element FAU\_STG.2.1**

**The TSF shall protect the stored audit data in the audit trail from unauthorized deletion.**

**8.8.7.4 Element FAU\_STG.2.2**

**The TSF shall be able to [selection, choose one of <sup>(s1)</sup>: *prevent, detect*] unauthorized modifications to the stored audit data in the audit trail.**

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify whether the TSF shall prevent or only be able to detect modifications of the stored audit data in the audit trail. Only one of these options may be chosen.

**8.8.8 FAU\_STG.3 Guarantees of audit data availability****8.8.8.1 Component relationships**

Hierarchical to: FAU\_STG.2 (Protected audit data storage) ([8.8.7](#))

Dependencies: FAU\_GEN.1 (Audit data generation) ([8.4.7](#))

**8.8.8.2 Component rationale**

This component allows the author of a PP, PP-Module, functional package or ST to specify to which metrics the audit trail should conform.

In a distributed environment, as the location of the audit trail is in the TSF, but not necessarily co-located with the function generating the audit data, the author of a PP, PP-Module, functional package or ST can request authentication of the originator of the audit record, or non-repudiation of the origin of the record prior storing this record in the audit trail.

**8.8.8.3 Element FAU\_STG.3.1**

**The TSF shall protect the stored audit data in the audit trail from unauthorized deletion.**

**ISO/IEC 15408-2:2026(en)****8.8.8.4 Element FAU\_STG.3.2**

The TSF shall be able to [selection, choose one of <sup>(s1)</sup>: *prevent, detect*] unauthorized modifications to the stored audit data in the audit trail.

In selection (s1), the author of a PP, PP-module, functional package or ST should specify whether the TSF shall prevent or only be able to detect modifications of the stored audit data in the audit trail. Only one of these options may be chosen.

**8.8.8.5 Element FAU\_STG.3.3**

**The TSF shall ensure that [assignment <sup>(a1)</sup>: *metric for saving audit data*] stored audit data will be maintained when the following conditions occur: [selection <sup>(s1)</sup>: *audit data storage exhaustion, failure, attack*].**

In assignment (a1), PP, PP-Module, functional package or ST author should specify the metric that the TSF shall ensure with respect to the stored audit records. This metric limits the data loss by enumerating the number of records that must be kept, or the time that records are guaranteed to be maintained.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the condition under which the TSF shall still be able to maintain a defined amount of audit data. This condition can be any of the following: audit storage exhaustion, failure, attack.

**8.8.9 FAU\_STG.4 Action in case of possible audit data loss****8.8.9.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FAU\_STG.2 (Protected audit data storage) ([8.8.7](#))

**8.8.9.2 Component rationale**

This component requires that actions will be taken when the audit trail exceeds certain pre-defined limits.

**8.8.9.3 Element FAU\_STG.4.1**

**The TSF shall [assignment <sup>(a1)</sup>: *actions to be taken in case of possible audit data storage failure*] if the audit data storage exceeds [assignment <sup>(a2)</sup>: *pre-defined limit*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify actions that should be taken in case of imminent audit storage failure indicated by exceeding the threshold. Actions can include informing an authorized user.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should indicate the pre-defined limit. If the management functions indicate that this number can be changed by the authorized user, this value is the default value. The author of a PP, PP-Module, functional package or ST can choose to let the authorized user define this limit.

**EXAMPLE** In the case that an authorized user defines the limit, an example of the assignment can be "an authorized user set limit".

## ISO/IEC 15408-2:2026(en)

### 8.8.10 FAU\_STG.5 Prevention of audit data loss

#### 8.8.10.1 Component relationships

Hierarchical to: No other components.

Dependencies: FAU\_STG.2 (Protected audit data storage) (8.8.7)  
 FAU\_GEN.1 (Audit data generation) (8.4.7)

#### 8.8.10.2 Component rationale

This component specifies the behaviour of the TOE if the audit trail is full: either audit records are ignored, or the TOE is frozen such that no audited events can take place. The requirement also states that no matter how the requirement is instantiated, the authorized user with specific rights to this effect, can continue to generate audited events (actions). The reason is that otherwise the authorized user cannot 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.

#### 8.8.10.3 Element FAU\_STG.5.1

**The TSF shall [selection (s1): *ignore audited events, prevent audited events except those taken by the authorized user with special rights, overwrite the oldest stored audit records, [assignment (a1): other actions to be taken in case of audit storage failure and conditions for the actions]] if the audit data storage is full.***

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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".

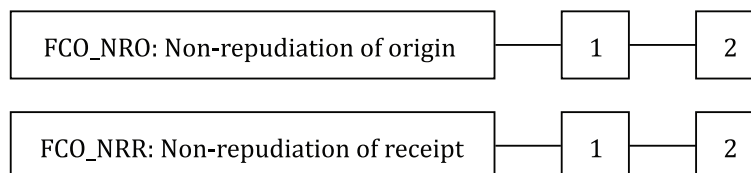
In selection (s1), the author of a PP, PP-Module, functional package or ST should select whether the TSF shall ignore audited actions, or whether it should prevent audited actions from happening, or whether the oldest audit records should be overwritten when the TSF can no longer store audit records. Only one of these options may be chosen.

## 9 Class FCO Communication

### 9.1 Introduction

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 15 illustrates the family hierarchy of class FCO.



**Key**  
 1,2 components

**Figure 15 — FCO: Communication, class decomposition**

Direct, indirect and optional dependencies for the components of class FCO are shown in [Table 2](#).

**Table 2 — Dependency table for class FCO: Communication**

	FIA_UID.1 (12.8.6)
FCO_NRO.1 ( <a href="#">9.3.6</a> )	X
FCO_NRO.2 ( <a href="#">9.3.7</a> )	X
FCO_NRR.1 ( <a href="#">9.4.6</a> )	X
FCO_NRR.2 ( <a href="#">9.4.7</a> )	X

## 9.2 Notes on class FCO

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 can 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" can 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.

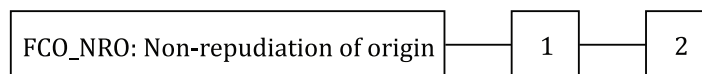
## 9.3 Non-repudiation of origin (FCO\_NRO)

### 9.3.1 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.

### 9.3.2 Component levelling and description

[Figure 16](#) illustrates the component levelling of family FCO\_NRO.



#### Key

1,2 components

**Figure 16 — FCO\_NRO: Component levelling**

FCO\_NRO.1 (Selective proof of origin) ([9.3.6](#)) requires the TSF to provide subjects with the capability to request evidence of the origin of transmitted information.

FCO\_NRO.2 (Enforced proof of origin) ([9.3.7](#)) requires that the TSF always generate evidence of origin for transmitted information.

**ISO/IEC 15408-2:2026(en)****9.3.3 Component management**

The following actions related to FCO\_NRO.1 and FCO\_NRO.2 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the management of changes to information types, fields, originator attributes and recipients of evidence.

**9.3.4 Component audit**

The following actions related to FCO\_NRO.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the identity of the user who requested that evidence of origin would be generated;
- minimal: the invocation of the non-repudiation service;
- basic: identification of the information, the destination, and a copy of the evidence provided;
- detailed: the identity of the user who requested a verification of the evidence.

The following actions related to FCO\_NRO.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the invocation of the non-repudiation service;
- basic: identification of the information, the destination, and a copy of the evidence provided;
- detailed: the identity of the user who requested a verification of the evidence.

**9.3.5 Application 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 can be a digital signature.

If the information or the associated attributes are altered in any way, validation of the evidence of origin can fail. Therefore, a PP, PP-Module, functional package or ST author should consider including integrity requirements such as FDP\_UIT.1 (Data exchange integrity) ([11.17.6](#)) in the PP, PP-Module, functional package or ST.

In non-repudiation, there are several different roles involved, each of which can 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) ([9.3.6](#))). 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 author of a PP, PP-Module, functional package or ST specifies the conditions that must be met to be able to verify the validity of the evidence.

**EXAMPLE 3** An example of a condition which can be specified is where the verification of evidence is required to occur within 24 h.

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 author of a PP, PP-Module, functional package or ST does not want the user identity to be exported. In that case, the author of a PP, PP-Module, functional package or ST considers whether it is

**ISO/IEC 15408-2:2026(en)**

appropriate to include this class, or whether the identity of the transport service provider or the identity of the host should be used.

In addition to (or instead of) the user identity, a PP, PP-Module, functional package or ST author can be more concerned about the time the information was transmitted.

EXAMPLE 4 Requests for proposals must be transmitted before a certain date in order to be considered.

In such instances, these requirements can be customized to provide a timestamp indication (time of origin).

### 9.3.6 FCO\_NRO.1 Selective proof of origin

#### 9.3.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

#### 9.3.6.2 Element FCO\_NRO.1.1

**The TSF shall be able to generate evidence of origin for transmitted [assignment <sup>(a1)</sup>: *list of information types*] at the request of the [selection <sup>(s1)</sup>: *originator, recipient, [assignment <sup>(a2)</sup>: *list of third parties*]].***

In assignment (a1), the author of a PP, PP-Module, functional package or ST should fill in the types of information subject to the evidence of origin function.

EXAMPLE 1 An example of the type of information is "electronic mail messages".

In assignment (a2), the author of a PP, PP-Module, functional package or ST, dependent on the selection, should specify the third parties that can request evidence of origin.

EXAMPLE 2 A third party can be an arbiter, judge, or legal body.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the user/subject who can request evidence of origin.

#### 9.3.6.3 Element FCO\_NRO.1.2

**The TSF shall be able to relate the [assignment <sup>(a1)</sup>: *list of attributes*] of the originator of the information, and the [assignment <sup>(a2)</sup>: *list of information fields*] of the information to which the evidence applies.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should fill in the list of the attributes that shall be linked to the information.

EXAMPLE Attributes include originator identity, time of origin, and location of origin.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should fill in the list of information fields within the information over which the attributes provide evidence of origin, such as the body of a message.

#### 9.3.6.4 Element FCO\_NRO.1.3

**The TSF shall provide a capability to verify the evidence of origin of information to [selection <sup>(s1)</sup>: *originator, recipient, [assignment <sup>(a1)</sup>: *list of third parties*]] given [assignment <sup>(a2)</sup>: *limitations on the evidence of origin*].***

In assignment (a1), the author of a PP, PP-Module, functional package or ST, dependent on the selection, should specify the third parties that can verify the evidence of origin.

## ISO/IEC 15408-2:2026(en)

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the user/subject who can verify the evidence of origin.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should fill in the list of limitations under which the evidence can be verified.

EXAMPLE An example of a limitation is "the evidence can only be verified within a 24-h time interval." An assignment of "immediate" or "indefinite" is acceptable.

### 9.3.7 FCO\_NRO.2 Enforced proof of origin

#### 9.3.7.1 Component relationships

Hierarchical to: FCO\_NRO.1 (Selective proof of origin) ([9.3.6](#))

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

#### 9.3.7.2 Element FCO\_NRO.2.1

The TSF shall **enforcethegenerationof** evidence of origin for transmitted [assignment <sup>(a1)</sup>: *list of information types*] at **alltimes**.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should fill in the types of information subject to the evidence of origin function.

#### 9.3.7.3 Element FCO\_NRO.2.2

The TSF shall be able to relate the [assignment <sup>(a1)</sup>: *list of attributes*] of the originator of the information, and the [assignment <sup>(a2)</sup>: *list of information fields*] of the information to which the evidence applies.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should fill in the list of the attributes that shall be linked to the information; for example, originator identity, time of origin, and location of origin.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should fill in the list of information fields within the information over which the attributes provide evidence of origin, such as the body of a message.

#### 9.3.7.4 Element FCO\_NRO.2.3

The TSF shall provide a capability to verify the evidence of origin of information to [selection <sup>(s1)</sup>: *originator, recipient, [assignment <sup>(a1)</sup>: list of third parties]*] given [assignment <sup>(a2)</sup>: *limitations on the evidence of origin*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the user/subject who can verify the evidence of origin.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should fill in the list of limitations under which the evidence can be verified.

EXAMPLE An example would be that the evidence can only be verified within a 24-h time interval.

## 9.4 Non-repudiation of receipt (FCO\_NRR)

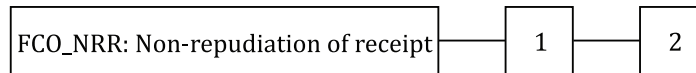
### 9.4.1 Family Behaviour

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

## ISO/IEC 15408-2:2026(en)

### 9.4.2 Component levelling and description

Figure 17 depicts the component levelling of family FCO\_NRR.



#### Key

1,2 components

**Figure 17 — FCO\_NRR: Component levelling**

FCO\_NRR.1 (Selective proof of receipt) (9.4.6) requires the TSF to provide subjects with a capability to request evidence of the receipt of information.

FCO\_NRR.2 (Enforced proof of receipt) (9.4.7) requires that the TSF always generate evidence of receipt for received information.

### 9.4.3 Component management

The following actions related to FCO\_NRR.1 and FCO\_NRR.2 can be considered for the management functions in class FMT (Security management) (Clause 13):

- the management of changes to information types, fields, originator attributes and third-party recipients of evidence.

### 9.4.4 Component audit

The following actions related to FCO\_NRR.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the identity of the user who requested that evidence of receipt would be generated;
- minimal: the invocation of the non-repudiation service;
- basic: identification of the information, the destination, and a copy of the evidence provided;
- detailed: the identity of the user who requested a verification of the evidence.

The following actions related to FCO\_NRR.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the invocation of the non-repudiation service;
- basic: identification of the information, the destination, and a copy of the evidence provided;
- detailed: the identity of the user who requested a verification of the evidence.

### 9.4.5 Application notes

Non-repudiation of receipt defines requirements to provide evidence to other users/subjects that the information was received by the recipient. The recipient cannot successfully deny having received the information because evidence of receipt provides evidence of the binding between the recipient attributes and the information. The originator or a third party can verify the evidence of receipt. This evidence should not be forgeable.

EXAMPLE 1 An example of evidence of receipt is a digital signature.

It should be noted that the provision of evidence that the information was received does not necessarily imply that the information was read or comprehended, but only delivered.

**ISO/IEC 15408-2:2026(en)**

If the information or the associated attributes are altered in any way, validation of the evidence of receipt with respect to the original information can fail. Therefore, a PP, PP-Module, functional package or ST author should consider including integrity requirements such as FDP\_UIT.1 (Data exchange integrity) (11.17.6) in the PP, PP-Module, functional package or ST.

In non-repudiation, there are several different roles involved, each of which can be combined in one or more subjects. The first role is a subject that requests evidence of receipt (only in FCO\_NRR.1 (Selective proof of receipt) (9.4.6)). 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 receipt, for example, an originator or a third party such as an arbiter.

EXAMPLE 2 A recipient or subject can be a notary.

The author of a PP, PP-Module, functional package or ST specifies the conditions that are required to be met to be able to verify the validity of the evidence.

EXAMPLE 3 An example of a condition which can be specified is where the verification of evidence must occur within 24 h.

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 author of a PP, PP-Module, functional package or ST does not want the user identity to be exported. In that case, the author of a PP, PP-Module, functional package or ST considers whether it is appropriate to include this class, or whether the identity of the transport service provider or the identity of the host should be used.

In addition to (or instead of) the user identity, a PP, PP-Module, functional package or ST author can be more concerned about the time the information was received.

EXAMPLE 4 When an offer expires at a certain date, orders must be received before a certain date in order to be considered.

In such instances, these requirements can be customized to provide a timestamp indication (time of receipt).

## 9.4.6 FCO\_NRR.1 Selective proof of receipt

### 9.4.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FIA\_UID.1 (Timing of identification) (12.8.6)

### 9.4.6.2 Element FCO\_NRR.1.1

**The TSF shall be able to generate evidence of receipt for received [assignment <sup>(a1)</sup>: list of information types] at the request of the [selection <sup>(s1)</sup>: originator, recipient, [assignment <sup>(a2)</sup>: list of third parties]].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should fill in the types of information subject to the evidence of receipt function, for example, electronic mail messages.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the user/subject who can request evidence of receipt.

In selection (s1), the author of a PP, PP-Module, functional package or ST, dependent on the selection, should specify the third parties that can request evidence of receipt.

**ISO/IEC 15408-2:2026(en)****9.4.6.3 Element FCO\_NRR.1.2**

**The TSF shall be able to relate the [assignment (a<sup>1</sup>): *list of attributes*] of the recipient of the information, and the [assignment (a<sup>2</sup>): *list of information fields*] of the information to which the evidence applies.**

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should fill in the list of the attributes that shall be linked to the information; for example, recipient identity, time of receipt, and location of receipt.

In assignment (a<sup>2</sup>), the author of a PP, PP-Module, functional package or ST should fill in the list of information fields with the fields within the information over which the attributes provide evidence of receipt, such as the body a message.

**9.4.6.4 Element FCO\_NRR.1.3**

**The TSF shall provide a capability to verify the evidence of receipt of information to [selection (s<sup>1</sup>): *originator, recipient, [assignment (a<sup>1</sup>): *list of third parties*]*] given [assignment (a<sup>2</sup>): *limitations on the evidence of receipt*].**

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST, dependent on the selection, should specify the third parties that can verify the evidence of receipt.

In selection (s<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should specify the user/subjects who can verify the evidence of receipt.

In assignment (a<sup>2</sup>), the author of a PP, PP-Module, functional package or ST should fill in the list of limitations under which the evidence can be verified. For example, the evidence can only be verified within a 24-hour time interval. An assignment of "immediate" or "indefinite" is acceptable.

**9.4.7 FCO\_NRR.2 Enforced proof of receipt****9.4.7.1 Component relationships**

Hierarchical to: FCO\_NRR.1 (Selective proof of receipt) ([9.4.6](#))

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

**9.4.7.2 Element FCO\_NRR.2.1**

The TSF shall **enforce the generation of** evidence of receipt for received [assignment (a<sup>1</sup>): *list of information types*] at **all times**.

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should fill in the types of information subject to the evidence of receipt function.

EXAMPLE Electronic mail messages.

**9.4.7.3 Element FCO\_NRR.2.2**

The TSF shall be able to relate the [assignment (a<sup>1</sup>): *list of attributes*] of the recipient of the information, and the [assignment (a<sup>2</sup>): *list of information fields*] of the information to which the evidence applies.

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should fill in the list of the attributes that shall be linked to the information.

EXAMPLE The recipient identity, time of receipt, and location of receipt.

In assignment (a<sup>2</sup>), the author of a PP, PP-Module, functional package or ST should fill in the list of information fields with the fields within the information over which the attributes provide evidence of receipt, such as the body of a message.

**ISO/IEC 15408-2:2026(en)****9.4.7.4 Element FCO\_NRR.2.3**

The TSF shall provide a capability to verify the evidence of receipt of information to [selection <sup>(s1)</sup>: *originator, recipient, [assignment <sup>(a1)</sup>: list of third parties]*] given [assignment <sup>(a2)</sup>: *limitations on the evidence of receipt*].

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the user/subjects who can verify the evidence of receipt.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should fill in the list of limitations under which the evidence can be verified. An assignment of "immediate" or "indefinite" is acceptable.

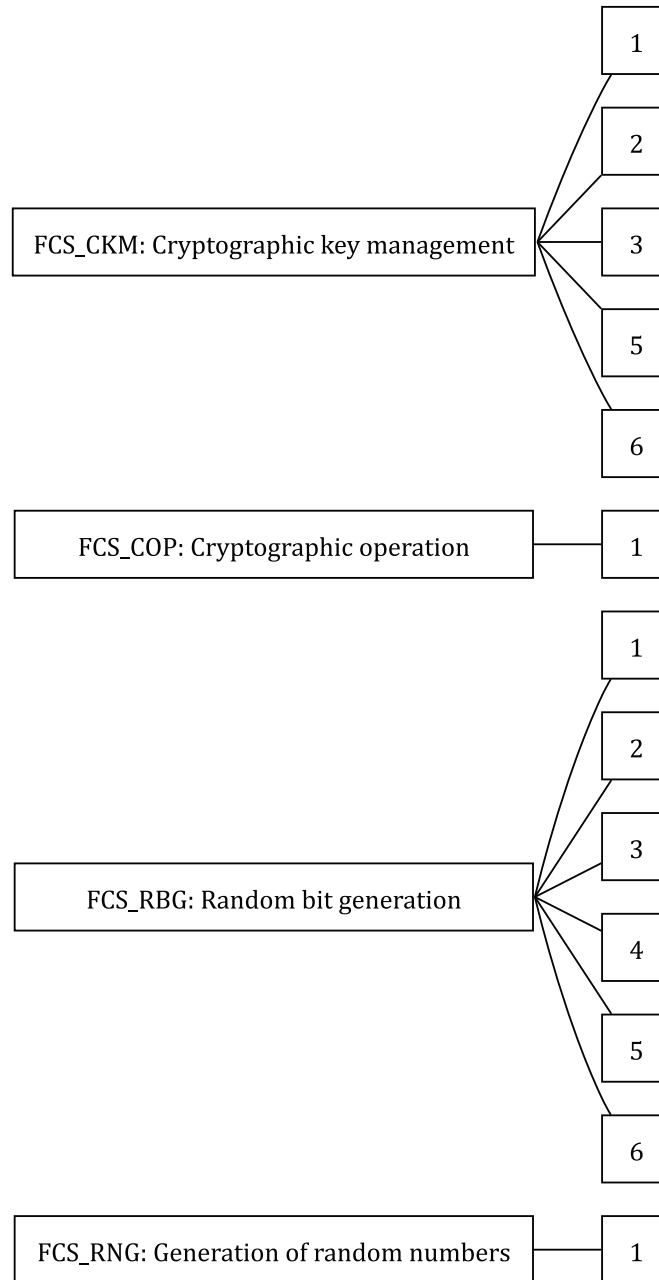
EXAMPLE When the evidence can only be verified within a 24 hour time interval.

**10 Class FCS Cryptographic support****10.1 Introduction**

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 can be in hardware, firmware and/or software.

[Figure 18](#) depicts the family hierarchy of class FCS.

**ISO/IEC 15408-2:2026(en)**



**Key**

1,2,3,4,5,6 components

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

Direct, indirect and optional dependencies for the components of class FCS are shown in [Table 3](#).

**ISO/IEC 15408-2:2026(en)**

**Table 3 — Dependency table for class FCS: Cryptographic support**

	FCS_CKM.1 (10.3.7)	FCS_CKM.2 (10.3.8)	FCS_CKM.5 (10.3.10)	FCS_CKM.6 (10.3.11)	FCS_COP.1 (10.4.6)	FCS_RBG.1 (10.5.6)	FCS_RBG.2 (10.5.7)	FCS_RBG.3 (10.5.8)	FCS_RBG.4 (10.5.9)	FCS_RBG.5 (10.5.10)	FCS_RNG.1 (10.6.6)	FDP_ITC.1 (11.10.6)	FDP_ITC.2 (11.10.7)	FPT_FLS.1 (15.4.6)	FPT_TST.1 (15.18.7)
FCS_CKM.1 (10.3.7)	-	0 <sup>1</sup>	0 <sup>1</sup>	X	0 <sup>1</sup>	0 <sup>2</sup>	-	-			0 <sup>2</sup>	-	-	-	-
FCS_CKM.2 (10.3.8)	0 <sup>1</sup>	-	0 <sup>1</sup>	-	-	-	-	-			-	0 <sup>1</sup>	0 <sup>1</sup>	-	-
FCS_CKM.3 (10.3.9)	0 <sup>1</sup>	-	0 <sup>1</sup>	-	-	-	-	-			-	0 <sup>1</sup>	0 <sup>1</sup>	-	-
FCS_CKM.5 (10.3.10)	-	0 <sup>1</sup>	-	X	0 <sup>1</sup>	-	-	-			-	-	-	-	-
FCS_CKM.6 (10.3.11)	0 <sup>1</sup>	-	0 <sup>1</sup>	-	-	-	-	-			-	0 <sup>1</sup>	0 <sup>1</sup>	-	-
FCS_COP.1 (10.4.6)	0 <sup>1</sup>	-	0 <sup>1</sup>	X	-	-	-	-			-	0 <sup>1</sup>	0 <sup>1</sup>	-	-
FCS_RBG.1 (10.5.6)						-	0 <sup>1</sup>	0 <sup>1</sup>						X	X
FCS_RBG.2 (10.5.7)						X	-	-						-	-
FCS_RBG.3 (10.5.8)						X	-	-						-	-
FCS_RBG.4 (10.5.9)						X	-	-	-	X				-	-
FCS_RBG.5 (10.5.10)						X	0 <sup>1</sup>	0 <sup>1</sup>	0 <sup>1</sup>	-				-	-
FCS_RBG.6 (10.5.11)						X	-	-						-	-
FCS_RNG.1 (10.6.6)															

**10.2 Notes on class FCS**

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;
- data separation.

This class is used when the TOE implements cryptographic functions, the implementation of which can be in hardware, firmware and/or software.

The FCS (Cryptographic support) class is composed of four families: FCS\_CKM (Cryptographic key management) (10.3), FCS\_COP (Cryptographic operation) (10.4), FCS\_RBG (Random bit generation) (10.5), and FCS\_RNG (Generation of random numbers) (10.6).

The FCS\_CKM (Cryptographic key management) (10.3) family addresses the management aspects of cryptographic keys; the FCS\_COP (Cryptographic operation) (10.4) family is concerned with the operational

**ISO/IEC 15408-2:2026(en)**

use of those cryptographic keys; the FCS\_RBG (Random bit generation) (10.5) family provides requirements for generating random bits; and the FCS\_RNG (Generation of random numbers) (10.6) is concerned with ensuring that random numbers meet defined quality metrics.

For each cryptographic key generation method implemented by the TOE, if any, the author of a PP, PP-Module, functional package or ST should select the FCS\_CKM.1 (Cryptographic key generation) (10.3.7) component.

For each cryptographic key distribution method implemented by the TOE, if any, the author of a PP, PP-Module, functional package or ST should select the FCS\_CKM.2 (Cryptographic key distribution) (10.3.8).

For each cryptographic key access method implemented by the TOE, if any, the author of a PP, PP-Module, functional package or ST should select the FCS\_CKM.3 (Cryptographic key access) (10.3.9).

For each cryptographic key derivation method implemented by the TOE, if any, the author of a PP, PP-Module, functional package or ST should select the FCS\_CKM.5 (Cryptographic key derivation) (10.3.10).

For each cryptographic key destruction method implemented by the TOE, if any, the author of a PP, PP-Module, functional package or ST should select the FCS\_CKM.6 (Timing and event of cryptographic key destruction) (10.3.11) component.

For each cryptographic operation (such as digital signature, data encryption, key agreement, secure hash, etc.) performed by the TOE, if any, the author of a PP, PP-Module, functional package or ST should select the FCS\_COP.1 (Cryptographic operation) (10.4.6) component.

For each deterministic random bit generation service implemented by the TOE, if any, the author of a PP, PP-Module, functional package or ST should select the FCS\_RBG.1 (Random bit generation (RBG)) (10.5.6) component.

For each external seeding source used by the TOE, if any, the author of a PP, PP-Module, functional package or ST should select the FCS\_RBG.2 (Random bit generation (external seeding)) (10.5.7) component.

For each internal seeding source (single) used by the TOE, if any, the author of a PP, PP-Module, functional package or ST should select the FCS\_RBG.3 (Random bit generation (internal seeding - single source)) (10.5.8) component.

Where internal seeding source (multiple) is intended to be specified, the author of a PP, PP-Module, functional package or ST should select the FCS\_RBG.4 (Random bit generation (internal seeding - multiple sources)) (10.5.9) component.

For cases where the TOE combines entropy sources, the FCS\_RBG.5 (Random bit generation (combining entropy sources)) (10.5.10) component should be specified by PP, PP-Module, functional package or ST author.

For each random bit generation service implemented by the TOE, the author of a PP, PP-Module, functional package or ST should specify the FCS\_RBG.6 (Random bit generation service) (10.5.11) component.

For each random number generation service implemented by the TOE, the author of a PP, PP-Module, functional package or ST should specify the FCS\_RNG.1 (Random number generation) (10.6.6) component.

To meet a variety of objectives, cryptographic functionality may be used to meet objectives specified in FCO (Communication), and in the following families FDP\_DAU (Data authentication) (11.5), FDP\_SDI (Stored data integrity) (11.15), FDP\_UCT (Inter-TSF user data confidentiality transfer protection) (11.16), FDP\_UIT (Inter-TSF user data integrity transfer protection) (11.17), FIA\_SOS (Specification of secrets) (12.6), FIA\_UAU (User authentication) (12.7), 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 assurance for the cryptographic functionality of the TOE is sought by consumers.

## 10.3 Cryptographic key management (FCS\_CKM)

### 10.3.1 Family Behaviour

Cryptographic keys should be managed throughout their life cycle. This family is intended to support that lifecycle and consequently defines requirements for the following activities:

- cryptographic key generation;
- cryptographic key distribution;
- cryptographic key access;
- cryptographic key derivation;
- timing and event of cryptographic key destruction.

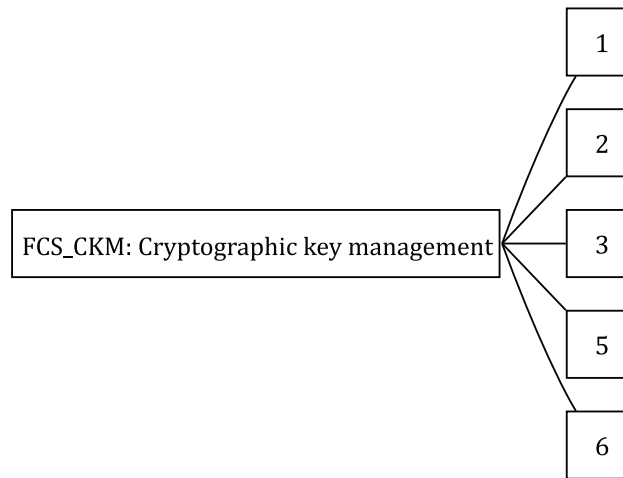
This family should be included whenever there are functional requirements for the management of cryptographic keys.

NOTE 1 Previous editions of this document specified FCS\_CKM (Cryptographic key management).4 which has been deprecated in this edition of this document. In order to preserve consistency when applying different editions of this document, the component number has not been re-used.

NOTE 2 Previous editions can refer to withdrawn and replaced documents.

### 10.3.2 Component levelling and description

[Figure 19](#) depicts the component levelling of family FCS\_CKM.



#### Key

1,2,3,5,6 components

**Figure 19 — FCS\_CKM: Component levelling**

FCS\_CKM.1 (Cryptographic key generation) ([10.3.7](#)) requires cryptographic keys to be generated in accordance with a specified algorithm and key sizes, which can be based on an assigned standard.

FCS\_CKM.2 (Cryptographic key distribution) ([10.3.8](#)) requires cryptographic keys to be distributed in accordance with a specified distribution method, which can be based on an assigned standard.

FCS\_CKM.3 (Cryptographic key access) ([10.3.9](#)) requires access to cryptographic keys to be performed in accordance with a specified access method, which can be based on an assigned standard.

FCS\_CKM.5 (Cryptographic key derivation) ([10.3.10](#)) requires that the methods, standards and parameters for key-derivation are specified.

**ISO/IEC 15408-2:2026(en)**

FCS\_CKM.6 (Timing and event of cryptographic key destruction) ([10.3.11](#)) requires cryptographic keys to be destroyed in accordance with specified destruction methods which can be based on an assigned standard.

**10.3.3 Component management**

There are no management activities foreseen related to FCS\_CKM.1, FCS\_CKM.2, FCS\_CKM.3, FCS\_CKM.5 or FCS\_CKM.6.

**10.3.4 Component audit**

The following actions related to FCS\_CKM.1, FCS\_CKM.2, FCS\_CKM.3, FCS\_CKM.5 and FCS\_CKM.6 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: success and failure of the activity;
- basic: the object attribute(s), and object value(s) excluding any sensitive information.

**10.3.5 Application notes**

Cryptographic keys are required to 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 1 Examples of key access include:

- backup;
- escrow;
- archive;
- recovery.

The inclusion of other stages is dependent on the key management strategy being implemented, as the TOE is not always involved in all of the key life-cycle phases.

EXAMPLE 2 The TOE can only generate and distribute cryptographic keys.

This family is intended to support the cryptographic key lifecycle and consequently defines requirements for the following activities:

- cryptographic key generation;
- cryptographic key derivation;
- cryptographic key distribution;
- cryptographic key access;
- cryptographic key destruction.

This family should be included whenever there are functional requirements for the management of cryptographic keys.

If FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST then, in the context of the events being audited:

- the object attributes may include the assigned user for the cryptographic key, the user role, the cryptographic operation for which the cryptographic key is used, the cryptographic key identifier and the cryptographic key validity period;

## ISO/IEC 15408-2:2026(en)

- the object value may include the values of cryptographic key(s) and parameters excluding any sensitive information (such as secret or private cryptographic keys).

Typically, FCS\_RNG.1 (Random number generation) (10.6.6) or FCS\_RBG.1 (Random bit generation (RBG)) (10.5.6) should be used respectively for random number/bit generation. Furthermore, in case of cryptographic key generation, FCS\_CKM.1 (Cryptographic key generation) (10.3.7) should be used. In cases where random number generation is required for purposes other than for the generation of cryptographic keys, the component FIA\_SOS.2 (TSF Generation of secrets) (12.6.7) can also be used.

### 10.3.6 Evaluator notes

Evaluators should refer to ISO/IEC 15408-1:2026, B.4 for information in regard to the use of standards specified in FCS\_CKM.5 (Cryptographic key derivation) (10.3.10).

FCS\_CKM.5 (Cryptographic key derivation) (10.3.10) has a dependency on FCS\_CKM.6 (Timing and event of cryptographic key destruction) (10.3.11). The dependency should be understood as the dependency of two directions, 1) destruction of key derivation key, and 2) destruction of derived keys. Evaluators should keep in mind that the dependency of two directions shall be fulfilled, and should also consider any intermediate values (such as those from key establishment) that should be destroyed in order to preserve the confidentiality of the key.

### 10.3.7 FCS\_CKM.1 Cryptographic key generation

#### 10.3.7.1 Component relationships

Hierarchical to: No other components.

Dependencies: [FCS\_CKM.2 (Cryptographic key distribution) (10.3.8), or FCS\_CKM.5 (Cryptographic key derivation) (10.3.10), or FCS\_COP.1 (Cryptographic operation) (10.4.6)]  
[FCS\_RBG.1 (Random bit generation (RBG)) (10.5.6), or FCS\_RNG.1 (Random number generation) (10.6.6)]  
FCS\_CKM.6 (Timing and event of cryptographic key destruction) (10.3.11)

#### 10.3.7.2 Component rationale

This component requires the cryptographic key sizes and method used to generate cryptographic keys to be specified, this may be in accordance with an assigned standard. It should be used to specify the cryptographic key sizes and the method used to generate the cryptographic keys. Only one instance of the component is needed for the same method and multiple key sizes. The key size may be common or different for the various entities and may be either the input to or the output from the method.

EXAMPLE An example of a method is an algorithm.

#### 10.3.7.3 Element FCS\_CKM.1.1

**The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment (a<sup>1</sup>): *cryptographic key generation algorithm*] and specified cryptographic key sizes [assignment (a<sup>2</sup>): *cryptographic key sizes*] that meet the following: [assignment (a<sup>3</sup>): *list of standards*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the cryptographic key generation algorithm to be used.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the cryptographic key sizes to be used. The key sizes specified should be appropriate for the algorithm and its intended use.

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify the assigned standard that documents the method used to generate cryptographic keys. The assigned standard may

**ISO/IEC 15408-2:2026(en)**

comprise none, one or more actual standards publications, for example, from international, national, industry or organizational standards.

**10.3.8 FCS\_CKM.2 Cryptographic key distribution****10.3.8.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FDP\_ITC.1 (Import of user data without security attributes) ([11.10.6](#)), or FDP\_ITC.2 (Import of user data with security attributes) ([11.10.7](#)), or FCS\_CKM.1 (Cryptographic key generation) ([10.3.7](#)), or FCS\_CKM.5 (Cryptographic key derivation) ([10.3.10](#))]

**10.3.8.2 Component rationale**

This component requires the method used to distribute cryptographic keys to be specified, this may be in accordance with an assigned standard. See ISO/IEC 15408-1 for information on using standards in PPs, PP-Modules, functional packages and STs.

**10.3.8.3 Element FCS\_CKM.2.1**

**The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [assignment <sup>(a1)</sup>: *cryptographic key distribution method*] that meets the following: [assignment <sup>(a2)</sup>: *list of standards*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the cryptographic key distribution method to be used.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the assigned standard that documents the method used to distribute cryptographic keys. The assigned standard may comprise none, one or more actual standards publications, for example, from international, national, industry or organizational standards.

**10.3.9 FCS\_CKM.3 Cryptographic key access****10.3.9.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FDP\_ITC.1 (Import of user data without security attributes) ([11.10.6](#)), or FDP\_ITC.2 (Import of user data with security attributes) ([11.10.7](#)), or FCS\_CKM.1 (Cryptographic key generation) ([10.3.7](#)), or FCS\_CKM.5 (Cryptographic key derivation) ([10.3.10](#))]

**10.3.9.2 Component rationale**

This component is intended to allow the specification of requirements on the usage of keys outside the TOE (e.g. backup, archival, escrow, recovery) and requires the methods used to access cryptographic keys be specified, this may be in accordance with an assigned standard.

FCS\_CKM.3 (Cryptographic key access) is not intended to postulate the access control on cryptographic keys.

**10.3.9.3 Element FCS\_CKM.3.1**

**The TSF shall perform [assignment <sup>(a1)</sup>: *type of cryptographic key access*] in accordance with a specified cryptographic key access method [assignment <sup>(a2)</sup>: *cryptographic key access method*] that meets the following: [assignment <sup>(a3)</sup>: *list of standards*].**

## ISO/IEC 15408-2:2026(en)

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the type of cryptographic key access being 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.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the cryptographic key access method to be used.

### 10.3.10 FCS\_CKM.5 Cryptographic key derivation

#### 10.3.10.1 Component relationships

Hierarchical to: No other components.

Dependencies: [FCS\_CKM.2 (Cryptographic key distribution) ([10.3.8](#)), or  
FCS\_COP.1 (Cryptographic operation) ([10.4.6](#))]  
FCS\_CKM.6 (Timing and event of cryptographic key destruction) ([10.3.11](#))

#### 10.3.10.2 Component rationale

This component requires the specification of the methods and parameters associated with the key derivation for a specified type of key.

FCS\_CKM.5 (Cryptographic key derivation) has a dependency on FCS\_CKM.6 (Timing and event of cryptographic key destruction) ([10.3.11](#)). The dependency should be understood as the dependency of two directions, 1) destruction of key derivation key, and 2) destruction of derived keys. PP, PP-Module, functional package and ST authors should keep in mind that the dependency of two directions shall be fulfilled and should also consider any intermediate values (such as those from key establishment) that should be destroyed in order to preserve the confidentiality of the key.

#### 10.3.10.3 Element FCS\_CKM.5.1

**The TSF shall derive cryptographic keys [assignment (a1): *key type*] from [assignment (a2): *input parameters*] in accordance with a specified key derivation algorithm [assignment (a3): *key derivation algorithm*] and specified cryptographic key sizes [assignment (a4): *list of key sizes*] that meet the following: [assignment (a5): *list of standards*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the type of cryptographic key to be derived.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify input parameters associated with the key derivation for a specified type of key.

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify key derivation algorithm to be used.

In assignment (a4), the author of a PP, PP-Module, functional package or ST should specify the cryptographic key sizes to be derived. The key sizes specified should be appropriate for the algorithm and its intended use.

In assignment (a5), the author of a PP, PP-Module, functional package or ST should specify the assigned standard that documents the method used to derive cryptographic keys. The assigned standard may comprise none, one or more actual standards publications, for example, from international, national, industry or organizational standards.

## ISO/IEC 15408-2:2026(en)

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

#### 10.3.11.1 Component relationships

Hierarchical to: No other components.

Dependencies: [FDP\_ITC.1 (Import of user data without security attributes) (11.10.6), or FDP\_ITC.2 (Import of user data with security attributes) (11.10.7), or FCS\_CKM.1 (Cryptographic key generation) (10.3.7), or FCS\_CKM.5 (Cryptographic key derivation) (10.3.10)]

#### 10.3.11.2 Component rationale

This component requires the list of keys, including any keying material and the method used to destroy cryptographic keys to be specified, this can be in accordance with an assigned standard.

The purpose of the destruction of cryptographic keys and keying material is to prevent their recovery in consideration of threats related to their compromise.

NOTE 1 Keying material includes keys and initialization vectors necessary to establish and maintain cryptographic keying relationships.

NOTE 2 When a DRBG is used to generate a cryptographic key or any keying material, and the PP/ST author desires to protect the DRBG state to avoid the possibility that knowledge of this state can compromise the key or keying material, then the PP/ST author includes DRBG entropy input, seed input, and internal state of the DRBG in the assignment in FCS\_CKM.6.1. See also FCS\_RBG.1 (Random bit generation (RBG)) (10.5.6) regarding the destruction of the DRBG state using the unstantiate operation.

#### 10.3.11.3 Element FCS\_CKM.6.1

**The TSF shall destroy [assignment <sup>(a1)</sup>: *list of cryptographic keys (including keying material)*] when [selection <sup>(s1)</sup>: *no longer needed*, [assignment <sup>(a2)</sup>: *other circumstances for key or keying material destruction*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should provide a list of cryptographic keys and keying material that should be destroyed under certain circumstances.

In selection (s1), and assignment (a2), the author of a PP, PP-Module, functional package or ST should select the circumstances of the destruction of keys or keying material. It can be chosen to destroy keys or keying material in case that these are no longer needed or to specify other circumstances for their destruction, e.g. the destruction of a key on reaching the limit of an error usage counter assigned to that key.

#### 10.3.11.4 Element 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 <sup>(a1)</sup>: *cryptographic key destruction method*] that meets the following: [assignment <sup>(a2)</sup>: *list of standards*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should provide the list of standards specifying the cryptographic key destruction method. The assigned list of standards may comprise none, one or more actual standards publications, for example, from international, national, industry or organisational standards.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should provide the cryptographic key destruction method.

## 10.4 Cryptographic operation (FCS\_COP)

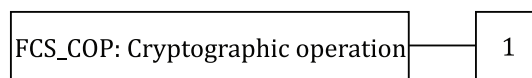
### 10.4.1 Family Behaviour

In order for a cryptographic operation to function correctly, the operation shall 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.

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

### 10.4.2 Component levelling and description

[Figure 20](#) shows the component levelling of family FCS\_COP.



#### Key

1 component

**Figure 20 — FCS\_COP: Component levelling**

FCS\_COP.1 (Cryptographic operation) ([10.4.6](#)) requires a cryptographic operation to be performed in accordance with a specified algorithm and with a cryptographic key of specified sizes. The specified algorithm and cryptographic key sizes can be based on an assigned standard.

### 10.4.3 Component management

There are no management activities foreseen related to FCS\_COP.1.

### 10.4.4 Component audit

The following actions related to FCS\_COP.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: success and failure, and the type of cryptographic operation;
- basic: any applicable cryptographic mode(s) of operation, subject attributes and object attributes.

### 10.4.5 Application notes

A cryptographic operation may have cryptographic mode(s) of operation associated with it. If this is the case, then the cryptographic mode(s) shall be specified.

**EXAMPLE 1** Examples of cryptographic modes of operation are cipher block chaining, output feedback mode, electronic code book mode, and cipher feedback mode.

Cryptographic operations may be used to support one or more TOE security services. The FCS\_COP (Cryptographic operation) component can need to be iterated more than once depending on:

- the user application for which the security service is being used;
- the use of different cryptographic algorithms and/or cryptographic key sizes;
- the type or sensitivity of the data being operated on.

## ISO/IEC 15408-2:2026(en)

If FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST then, in the context of the cryptographic operation events being audited:

- the types of cryptographic operation may include digital signature generation and/or verification, cryptographic checksum generation for integrity and/or for verification of checksum, secure hash (message digest) computation, data encryption and/or decryption, cryptographic key encryption and/or decryption, cryptographic key agreement, and random number generation;
- the subject attributes may include subject role(s) and user(s) associated with the subject;
- the object attributes may include the assigned user for the cryptographic key, user role, cryptographic operation for which the cryptographic key is used, cryptographic key identifier, and the cryptographic key validity period.

When specifying cryptographic operations, the author of a PP, PP-Module, functional package or ST should perform due diligence in order to have confidence that the specified cryptographic operations are appropriate for the selected assurance requirements and in consideration of the technology types, environment and use cases of the TOE.

NOTE In some cases, certification bodies can apply policies in regard to the selection of cryptographic operations (see ISO/IEC 18045:2026, A.6).

In case that a cryptographic algorithm operation is specified in FCS\_COP.1 (Cryptographic operation) (10.4.6) where a TOE-internal generation of random bits or random numbers is part of that cryptographic algorithm operation such generation of random bits or random numbers should be part of the TSF and covered by corresponding SFRs. This should either be done by FCS\_RBG.1 (Random bit generation (RBG)) (10.5.6) or FCS\_RNG.1 (Random number generation) (10.6.6) whereby a corresponding dependency on the chosen additional SFR should supplement the FCS\_COP.1 (Cryptographic operation) (10.4.6), or - if applicable - by using the dependency of FCS\_COP.1 (Cryptographic operation) (10.4.6) on FCS\_CKM.1 (Cryptographic key generation) (10.3.7) (that itself contains a dependency on FCS\_RBG.1 (Random bit generation (RBG)) (10.5.6) or FCS\_RNG.1 (Random number generation) (10.6.6) respectively).

EXAMPLE 2 Non-deterministic cryptographic algorithm like DSA signature generation, ECDSA signature generation, RSASSA-PSS signature generation.

### 10.4.6 FCS\_COP.1 Cryptographic operation

#### 10.4.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: [FDP\_ITC.1 (Import of user data without security attributes) (11.10.6), or FDP\_ITC.2 (Import of user data with security attributes) (11.10.7), or FCS\_CKM.1 (Cryptographic key generation) (10.3.7), or FCS\_CKM.5 (Cryptographic key derivation) (10.3.10)]  
FCS\_CKM.6 (Timing and event of cryptographic key destruction) (10.3.11)

#### 10.4.6.2 Component rationale

This component requires the cryptographic algorithm and key size used to perform specified cryptographic operation(s) which can be based on an assigned standard.

#### 10.4.6.3 Element FCS\_COP.1.1

The TSF shall perform [assignment <sup>(a1)</sup>: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment <sup>(a2)</sup>: *cryptographic algorithm*] and cryptographic key sizes [assignment <sup>(a3)</sup>: *cryptographic key sizes*] that meet the following: [assignment <sup>(a4)</sup>: *list of standards*].

## ISO/IEC 15408-2:2026(en)

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the cryptographic operations being performed. Typical cryptographic operations include digital signature generation and/or verification, cryptographic checksum generation for integrity and/or for verification of checksum, secure hash (message digest) computation, data encryption and/or decryption, cryptographic key encryption and/or decryption, cryptographic key agreement, and random number generation. The cryptographic operation may be performed on user data or TSF data.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the cryptographic algorithm to be used.

**EXAMPLE** Examples of typical cryptographic algorithms include, but are not limited to, DES, RSA and IDEA.

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify the cryptographic key sizes to be used. The key sizes specified should be appropriate for the algorithm and its intended use.

In assignment (a4), the author of a PP, PP-Module, functional package or ST should specify the assigned standard that documents how the identified cryptographic operation(s) are performed. The assigned standard may comprise none, one or more actual standards publications, for example, from international, national, industry or organisational standards.

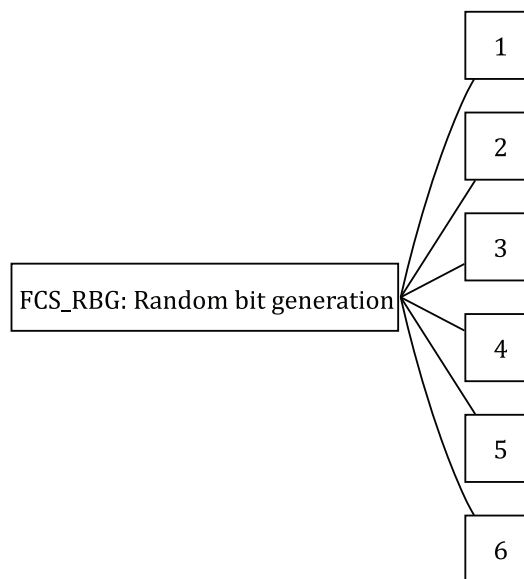
### 10.5 Random bit generation (FCS\_RBG)

#### 10.5.1 Family Behaviour

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

#### 10.5.2 Component levelling and description

[Figure 21](#) depicts the component levelling of family FCS\_RBG.



**Key**

1,2,3,4,5,6 components

**Figure 21 — FCS\_RBG: Component levelling**

FCS\_RBG.1 (Random bit generation (RBG)) ([10.5.6](#)) requires random bit generation to be performed in accordance with selected standards. It also specifies whether the initial seeding is done via an internal or external noise source, as well as when and how an RBG's state is updated.

FCS\_RBG.2 (Random bit generation (external seeding)) ([10.5.7](#)) gives requirements for seeding by an external (outside the TOE) entropy source.

**ISO/IEC 15408-2:2026(en)**

FCS\_RBG.3 (Random bit generation (internal seeding - single source)) ([10.5.8](#)) gives requirements for seeding using a TSF entropy source.

FCS\_RBG.4 (Random bit generation (internal seeding - multiple sources)) ([10.5.9](#)) gives requirements for seeding using multiple TSF entropy sources.

FCS\_RBG.5 (Random bit generation (combining entropy sources)) ([10.5.10](#)) gives requirements for combining multiple entropy sources (multiple internal sources, internal and external).

FCS\_RBG.6 (Random bit generation service) ([10.5.11](#)) requires random numbers to be supplied over an external interface as a service to other entities.

**10.5.3 Component management**

There are no management activities foreseen related to FCS\_RBG.1, FCS\_RBG.2, FCS\_RBG.3, FCS\_RBG.4, FCS\_RBG.5 or FCS\_RBG.6.

**10.5.4 Component audit**

The following action related to FCS\_RBG.1 and FCS\_RBG.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: failure of the randomization process, failure to initialize or reseed (as supported by the technology).

There are no auditable events foreseen related to FCS\_RBG.3, FCS\_RBG.4, FCS\_RBG.5 or FCS\_RBG.6.

**10.5.5 Application notes**

When specifying random bit generation methods, the author of a PP, PP-Module, functional package or ST should perform due diligence in order to have confidence that the specifications are appropriate for the selected assurance requirements and in consideration of the technology types, environment and use cases of the TOE.

NOTE In some cases, certification bodies can apply policies in regard to the selection of random bit generators (see ISO/IEC 18045:2026, A.6).

**10.5.6 FCS\_RBG.1 Random bit generation (RBG)****10.5.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FCS\_RBG.2 (Random bit generation (external seeding)) ([10.5.7](#)), or  
FCS\_RBG.3 (Random bit generation (internal seeding - single source)) ([10.5.8](#))  
FPT\_FLS.1 (Failure with preservation of secure state) ([15.4.6](#))  
FPT\_TST.1 (TSF self-testing) ([15.18.7](#))

**10.5.6.2 Component rationale**

For FCS\_RBG.1 (Random bit generation (RBG)), these dependencies shall always be met.

ISO/IEC 15408-1:2026, 8.3 allows a justification to be provided if a dependency is not met for this component.

The entropy source can be a raw noise source or conditioned entropy.

Reseeding is the typical mechanism for updating the DRBG state and means that additional entropy is provided to the DRBG. If reseeding is not feasible, the TSF should uninstantiate and re-instantiate DRBGs rather than produce output that is of insufficient quality.

**ISO/IEC 15408-2:2026(en)**

"Uninstantiate" means that the internal state of the DRBG is no longer available for use.

The situation "never" should be selected only if the DRBG cannot be reseeded or uninstantiated.

The situation "on demand" indicates that there is an interface to trigger reseeding or uninstantiating of the DRBG, whether internal to the TOE or presented as a TSFI (e.g. an API call).

The situation "on the condition" allows the PP/ST author to specify application-specific conditions for reseeding.

The list of standards should specify the reseed interval, and the process for reseeding. This assignment should be "None" if the situation is "never".

Health tests for the DRBG are specified in FPT\_TST.1 (TSF self-testing) ([15.18.7](#)).

**NOTE** If a TOE is expected to protect the DRBG state to avoid the possibility that knowledge of this state can compromise a key or keying material derived from its output, then the PP/ST author will include DRBG entropy input, seed input, and internal state of the DRBG in the assignment in an instance of FCS\_CKM.6.1. This applies particularly where neither "reseeding" nor "re-instantiating" apply in the last selection of FCS\_RBG.1.3 (and therefore where a different method of destruction is specified).

**10.5.6.3 Element FCS\_RBG.1.1**

**The TSF shall perform deterministic random bit generation services using [assignment (a1): *DRBG algorithm*] in accordance with [assignment (a2): *list of standards*] after initialization.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the DRBG algorithm to be used.

**EXAMPLE** Examples of typical DRBG algorithms include, but are not limited to, CTR\_DRBG, Hash\_DRBG, HMAC\_DRBG.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the assigned standard that documents how the identified DRBG algorithm is performed. The assigned standard may comprise one or more actual standards publications, these may include standards from international, national, industry or organizational standards.

**10.5.6.4 Element FCS\_RBG.1.2**

**The TSF shall use a [selection (s1): *TSF entropy source [assignment (a1): name of entropy source], TSF interface for obtaining entropy*] for initialization and reseeding.**

In selection (s1), and assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the entropy source or the TSF interface for obtaining entropy by providing the name.

**EXAMPLE** Examples of typical TSF entropy sources include, but are not limited to, Jitter, IRQs, CPU timing differences.

**10.5.6.5 Element FCS\_RBG.1.3**

**The TSF shall update the DRBG state by [selection (s1): *reseeding, uninstantiating and re-instantiating*] using a [selection (s2): *TSF entropy source [assignment (a1): name of entropy source], TSF interface for obtaining entropy [assignment (a2): name of the interface]*] in the following situations: [selection (s3): *never, on demand, on the condition: [assignment (a3): condition], after [assignment (a4): time]*] in accordance with [assignment (a5): *list of standards*].**

In selection (s2), and in assignments (a1) and (a2), the author of a PP, PP-Module, functional package or ST should specify the entropy source or the TSF interface for obtaining entropy by providing the name.

In selection (s3), and assignment (a4), the author of a PP, PP-Module, functional package or ST should specify the condition or the time interval for the update of the DRBG state.

**ISO/IEC 15408-2:2026(en)**

In assignment (a5), the author of a PP, PP-Module, functional package or ST should specify the assigned standard that documents how the identified DRBG state is updated. The assigned standard may comprise one or more actual standards publications, these may include standards from international, national, industry or organizational standards. When the first selection in FCS\_RBG.1.3 is completed with "reseeding", the standard should match the one specified in FCS\_RBG.1.1. When the first selection in FCS\_RBG.1.3 is completed with "uninstantiating and re-instantiating", the assignment should be completed with "none".

**10.5.7 FCS\_RBG.2 Random bit generation (external seeding)****10.5.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FCS\_RBG.1 (Random bit generation (RBG)) ([10.5.6](#))

**10.5.7.2 Component rationale**

For this component, the interface to obtain the entropy noise source can be used multiple times to provide input. For instance, if the input length is 128 bits, it can be used twice to gather 256 bits. In this instance, the 128 bits would not be provided to the DRBG, since the DRBG can only be instantiated once, rather a function would gather the 128 bits twice and provide the DRBG with 256 bits of entropy noise source.

This component does not describe requirements on seed quality: It is the responsibility of the operational environment to define their requirement in this regard and to ensure that it is met by the external source.

Guidance in the introduction to PP, PP-Module, functional package or ST authors should address protection from modification and disclosure of the value from the external noise source, as well as the leaking of any pertinent information (e.g. internal state) regarding the DRBG.

**10.5.7.3 Element FCS\_RBG.2.1**

**The TSF shall be able to accept a minimum input of [assignment (a1): *minimum input length greater than zero*] from a TSF interface for obtaining entropy.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the minimum input length greater than zero in bits by providing a number.

**10.5.8 FCS\_RBG.3 Random bit generation (internal seeding - single source)****10.5.8.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FCS\_RBG.1 (Random bit generation (RBG)) ([10.5.6](#))

**10.5.8.2 Component rationale**

If the author of a PP, PP-Module, functional package or ST wishes to use multiple internal noise sources, they iterate this requirement for each noise source being used by the TSF.

Hardware-based noise sources are sources whose primary function is noise generation, such as ring oscillators, diodes, and thermal noise. While software is used to collect the noise from these hardware sources, these are not software-based. Software-based noise sources are those sources that have some other primary function and the noise is a byproduct of their normal operation.

**EXAMPLE** Examples of software-based noise sources are user or system-based events, reading the least significant bits from an event timer.

## ISO/IEC 15408-2:2026(en)

Hardware-based noise sources may be stochastically modelled, in which case the amount of entropy is well understood. Software-based noise sources are usually less well understood and therefore will typically take a more conservative approach, gathering larger numbers of bits than required and then performing a compression function to derive the final output. Software-based noise sources often rely on an entropy estimator.

### 10.5.8.3 Element FCS\_RBG.3.1

**The TSF shall be able to seed the DRBG using a [selection, choose one of <sup>(s1)</sup>: *TSF software-based entropy source, TSF hardware-based entropy source*][assignment <sup>(a1)</sup>: *name of entropy source*] with [assignment <sup>(a2)</sup>: *number of bits*] bits of min-entropy.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the TSF entropy source by providing the name.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the number of bits of min-entropy for seeding the DRBG. The bits of min-entropy are required to be declared regardless of the number of bits supplied.

### 10.5.9 FCS\_RBG.4 Random bit generation (internal seeding - multiple sources)

#### 10.5.9.1 Component relationships

Hierarchical to: No other components.

Dependencies: FCS\_RBG.1 (Random bit generation (RBG)) ([10.5.6](#))  
FCS\_RBG.5 (Random bit generation (combining entropy sources)) ([10.5.10](#))

#### 10.5.9.2 Component rationale

The minimum entropy is defined per source/iteration of FCS\_RBG.3.1. The resulting minimum entropy is covered by FCS\_RBG.5.1 which is a dependency of FCS\_RBG.4.1.

#### 10.5.9.3 Element FCS\_RBG.4.1

**The TSF shall be able to seed the DRBG using [selection <sup>(s1)</sup>: *[assignment <sup>(a1)</sup>: number] TSF software-based entropy source(s), [assignment <sup>(a2)</sup>: number] TSF hardware-based entropy source(s)*].**

In assignment (a1), and assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the number of TSF software-based and/or TSF hardware-based entropy sources.

### 10.5.10 FCS\_RBG.5 Random bit generation (combining entropy sources)

#### 10.5.10.1 Component relationships

Hierarchical to: No other components.

Dependencies: FCS\_RBG.1 (Random bit generation (RBG)) ([10.5.6](#))  
[FCS\_RBG.2 (Random bit generation (external seeding)) ([10.5.7](#)), or  
FCS\_RBG.3 (Random bit generation (internal seeding - single source)) ([10.5.8](#)), or  
FCS\_RBG.4 (Random bit generation (internal seeding - multiple sources)) ([10.5.9](#))]

#### 10.5.10.2 Component rationale

This component addresses operations used for combining multiple entropy sources to obtain min-entropy for further processing. Input to the combining operation can come from the TSF entropy source(s) specified in FCS\_RBG.3.1 and/or FCS\_RBG.4.1 and/or from the TSF interface(s) for obtaining entropy specified in FCS\_RBG.2.1.

## ISO/IEC 15408-2:2026(en)

### 10.5.10.3 Element FCS\_RBG.5.1

The TSF shall [assignment (a1): *combining operation*][selection (s1): *output from TSF entropy source(s), input from TSF interface(s) for obtaining entropy*] resulting in a minimum of [assignment (a2): *number of bits*] bits of min-entropy to create the entropy input into the derivation function as defined in [assignment (a3): *list of standards*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the operation to combine the output from the TSF entropy source(s) and/or input from the TSF interface(s) for obtaining entropy.

EXAMPLE Examples of typical combining operations include, but are not limited to, XORing or hashing.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the number of bits of the resulting min-entropy.

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify the assigned standard that documents how the entropy input is combined. The assigned standard may comprise one or more actual standards publications, these may include standards from international, national, industry or organizational standards.

### 10.5.11 FCS\_RBG.6 Random bit generation service

#### 10.5.11.1 Component relationships

Hierarchical to: No other components.

Dependencies: FCS\_RBG.1 (Random bit generation (RBG)) ([10.5.6](#))

#### 10.5.11.2 Component rationale

Specifying the interface type is important for developing evaluation activities and important information for an external instance requesting the DRBG service from the TOE.

#### 10.5.11.3 Element FCS\_RBG.6.1

The TSF shall provide a [selection (s1): *hardware, software, [assignment (a1): other interface type]*] interface to make the DRBG output, as specified in FCS\_RBG.1 (Random bit generation (RBG)) ([10.5.6](#)), available as a service to entities outside of the TOE.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify other interface types, if applicable. The assignment inside of the selection is introduced to allow other or more specific interface types than just "hardware" or "software". Other interface types can be a service over a network interface.

EXAMPLE Ethernet, wireless.

In selection (s1), the author of a PP, PP-Module, functional package or ST should either select the interface type "hardware" or "software" or should specify another interface type.

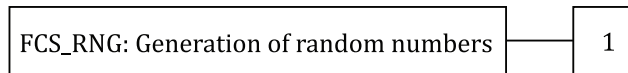
## 10.6 Generation of random numbers (FCS\_RNG)

### 10.6.1 Family Behaviour

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

## 10.6.2 Component levelling and description

[Figure 22](#) describes the component levelling of family FCS\_RNG.



### Key

1 component

**Figure 22 — FCS\_RNG: Component levelling**

FCS\_RNG.1 (Random number generation) ([10.6.6](#)) requires that random numbers meet a defined quality metric.

## 10.6.3 Component management

There are no management activities foreseen related to FCS\_RNG.1.

## 10.6.4 Component audit

There are no auditable events foreseen related to FCS\_RNG.1.

## 10.6.5 Application notes

When specifying random number generation methods, the author of a PP, PP-Module, functional package or ST should perform due diligence in order to have confidence that the specifications are appropriate for the selected assurance requirements and in consideration of the technology types, environment and use cases of the TOE.

NOTE In some cases, certification bodies can apply policies in regard to the selection of random number generators (see ISO/IEC 18045:2026, A.6).

## 10.6.6 FCS\_RNG.1 Random number generation

### 10.6.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

### 10.6.6.2 Component rationale

The random number generator is expected to comply with up-to-date standards/specifications. The evaluation of the random number generator should follow a recognized evaluation methodology.

EXAMPLE 1 Some users of Generation of random numbers (FCS\_RNG (Generation of random numbers)) can find useful NIST SP 800-90A<sup>[9]</sup> and NIST SP 800-90B<sup>[10]</sup> published by the National Institute of Standards and Technology (NIST).

EXAMPLE 2 An example of a recognized methodology of evaluation is AIS 31<sup>[1]</sup> published by the Bundesamt für Sicherheit in der Informationstechnik (BSI), Germany.

### 10.6.6.3 Element FCS\_RNG.1.1

The TSF shall provide a [selection <sup>(s1)</sup>: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*] random number generator that implements: [assignment <sup>(a1)</sup>: *list of security capabilities*].

**ISO/IEC 15408-2:2026(en)**

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the type of random number generator as physical, non-physical true, deterministic, hybrid physical or hybrid deterministic.

A physical random number generator (RNG) produces random numbers using a noise source based on physical random processes. A non-physical true RNG uses a noise source based on non-physical random processes like human interaction (key strokes, mouse movement). A deterministic RNG uses a random seed to produce a pseudorandom output. A hybrid physical or hybrid deterministic RNG combines the principles of physical and deterministic RNGs. A hybrid physical RNG produces at least the amount of entropy the RNG output may contain while the internal state of a hybrid deterministic RNG output contains fresh entropy but less than the output of an RNG may contain.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of security capabilities provided by the random number generator of the TOE.

In the case of a PP, PP-Module or functional package, FCS\_RNG.1.1 can be completed with a more restrictive language such as:

- [assignment: list of additional security capabilities];
- [selection: security\_capability\_1, ..., security\_capability\_n];
- mixtures of such selections and assignments within the list of security capabilities.

**EXAMPLE** Examples of security capabilities include:

- a total failure test detects a total failure of entropy source immediately when the RNG has started. When a total failure is detected, no random numbers will be output;
- if a total failure of the entropy source occurs while the RNG is being operated, the RNG [selection: prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source, generates the internal random numbers with a post-processing algorithm of class DRG.2 (as defined in AIS 31 [\[1\]](#)) as long as its internal state entropy guarantees the claimed output entropy];
- the online test detects non-tolerable statistical defects of the raw random number sequence (i) immediately when the RNG has started, and (ii) while the RNG is being operated. The TSF must not output any random numbers before the power-up online test has finished successfully or when a defect has been detected;
- the online test procedure be effective to detect non-tolerable weaknesses of the random numbers soon.
- the online test procedure checks the quality of the raw random number sequence. It is triggered [selection: externally, at regular intervals, continuously, applied upon specified internal events]. The online test is suitable for detecting non-tolerable statistical defects of the statistical properties of the raw random numbers within an acceptable period of time;
- failure or severe degradation of the noise source be detectable;
- continuous tests or other mechanisms in the entropy source protect against producing output during malfunctions.

#### 10.6.6.4 Element FCS\_RNG.1.2

**The TSF shall provide [selection (s1): *bits, octets of bits, numbers* [assignment (a1): *format of the numbers*]] that meet [assignment (a2): *a defined quality metric*].**

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the type of random output as bits, octets of bits or numbers whereby in the latter case the format of the numbers is required to be specified.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify an appropriate quality metric for the random output.

**NOTE 1** In the case of a PP, PP-Module or functional package, FCS\_RNG.1.2 can be completed with a more restrictive language such as:

- [selection: defined\_quality\_metric\_1, ..., defined\_quality\_metric\_n];

**ISO/IEC 15408-2:2026(en)**

- [assignment: a defined quality metric];
- mixtures of such selections and assignments

within the quality metric.

NOTE 2 The "quality metric" can include both qualitative metric and quantitative metric.

EXAMPLE Examples of quality metrics include

- test procedure A [assignment: additional standard test suites] does not distinguish the internal random numbers from output sequences of an ideal RNG;

NOTE 3 The assignment for additional standard statistical test suite can be empty.

- the average Shannon entropy per internal random bit exceeds 0.998;
- each output bit is independent of all other output bits;
- [selection: full entropy output, [assignment: bias and entropy rate of the output]].

## 11 Class FDP User data protection

### 11.1 Introduction

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

The families in this class are organized into four groups:

- user data protection SFPs:
  - FDP\_ACC (Access control policy) ([11.3](#));
  - FDP\_IFC (Information flow control policy) ([11.7](#)).

Components in these families permit the author of a PP, PP-Module, functional package or ST to name the user data protection SFPs and define the scope of control of the policy, necessary to address the security objectives. The names of these policies 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 "access control SFP" or an "information flow control SFP". The rules that define the functionality of the named access control and information flow control SFPs will be defined in the FDP\_ACF (Access control functions) ([11.4](#)) and FDP\_IFF (Information flow control functions) ([11.8](#)) families (respectively).

- forms of user data protection:
  - FDP\_ACF (Access control functions) ([11.4](#));
  - FDP\_IFF (Information flow control functions) ([11.8](#));
  - FDP\_ITT (Internal TOE transfer) ([11.11](#));
  - FDP\_IRC (Information retention control) ([11.9](#));
  - FDP\_RIP (Residual information protection) ([11.12](#));
  - FDP\_ROL (Rollback) ([11.13](#));
  - FDP\_SDC (Stored data confidentiality) ([11.14](#));

## ISO/IEC 15408-2:2026(en)

- FDP\_SDI (Stored data integrity) ([11.15](#)).
- off-line storage, import and export:
  - FDP\_DAU (Data authentication) ([11.5](#));
  - FDP\_ETC (Export from the TOE) ([11.6](#));
  - FDP\_ITC (Import from outside of the TOE) ([11.10](#)).

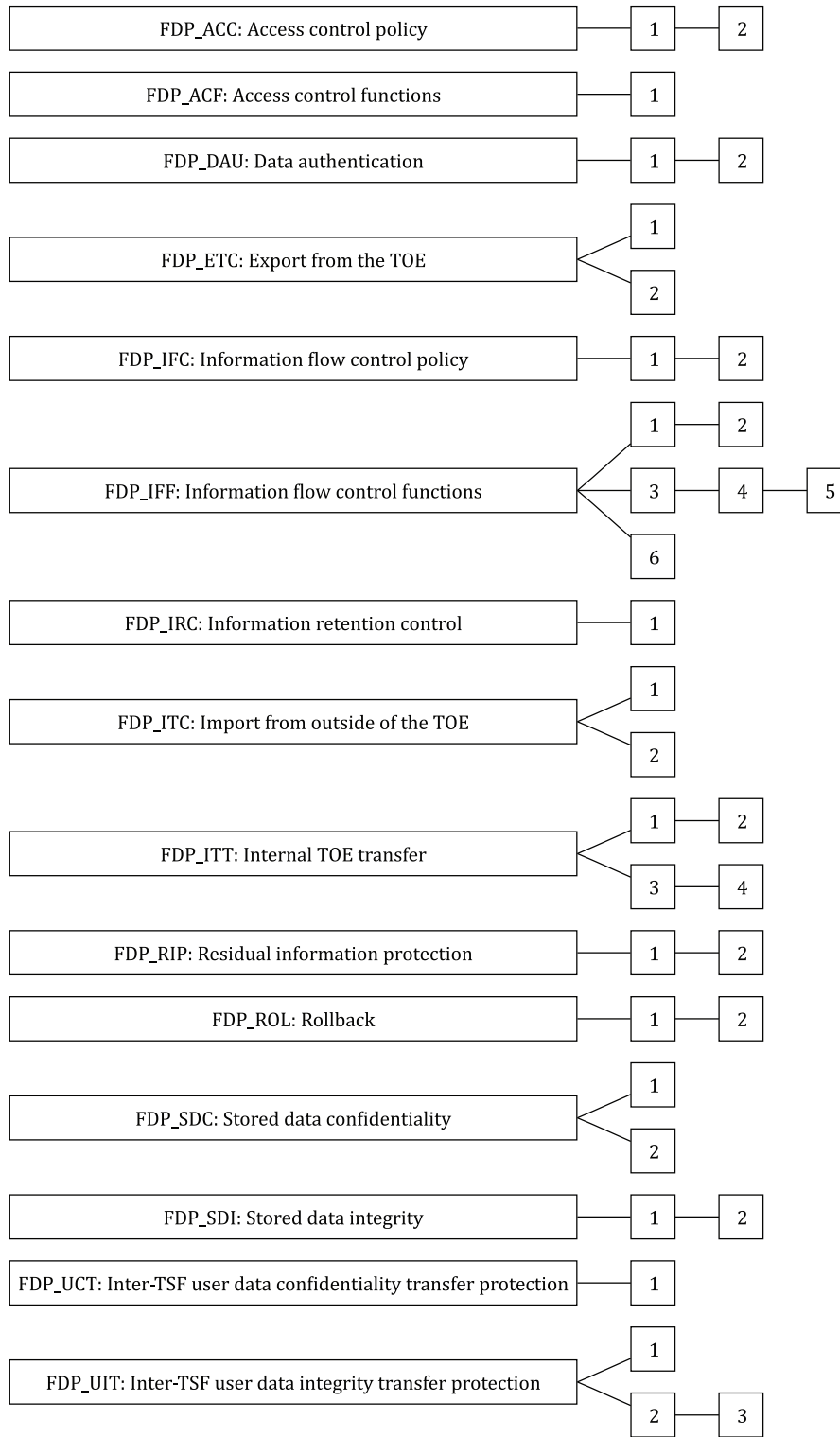
Components in these families address the trustworthy transfer into or out of the TOE.

- inter-TSF communication:
  - FDP\_UCT (Inter-TSF user data confidentiality transfer protection) ([11.16](#));
  - FDP\_UIT (Inter-TSF user data integrity transfer protection) ([11.17](#)).

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

[Figure 23](#) describes the family hierarchy of class FDP.

**ISO/IEC 15408-2:2026(en)**



**Key**

1,2,3,4,5,6 components

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

Direct, indirect and optional dependencies for the components of class FDP are shown in [Table 4](#).

**ISO/IEC 15408-2:2026(en)**

**Table 4 — Dependency table for class FDP: User data protection**

	FCS_COP.1 (10.4.6)	FDP_ACC.1 (11.3.6)	FDP_ACF.1 (11.4.6)	FDP_IFC.1 (11.7.6)	FDP_IFF.1 (11.8.6)	FDP_ITT.1 (11.11.6)	FDP_ITT.2 (11.11.7)	FDP_UIT.1 (11.17.6)	FIA_UID.1 (12.8.6)	FMT_MSA.3 (13.5.8)	FPT_TDC.1 (15.15.6)	FPT_ITC.1 (18.3.6)	FPT_TRP.1 (18.5.6)
FDP_ACC.1 (11.3.6)		-	X							-			
FDP_ACC.2 (11.3.7)		-	X							-			
FDP_ACF.1 (11.4.6)		X	-							X			
FDP_DAU.1 (11.5.6)													
FDP_DAU.2 (11.5.7)									X				
FDP_ETC.1 (11.6.6)		O <sup>1</sup>	-	O <sup>1</sup>	-					-			
FDP_ETC.2 (11.6.7)		O <sup>1</sup>	-	O <sup>1</sup>	-					-			
FDP_IFC.1 (11.7.6)				-	X					-			
FDP_IFC.2 (11.7.7)				-	X					-			
FDP_IFF.1 (11.8.6)				X	-					X			
FDP_IFF.2 (11.8.7)				X	-					X			
FDP_IFF.3 (11.8.8)				X	-					-			
FDP_IFF.4 (11.8.9)				X	-					-			
FDP_IFF.5 (11.8.10)				X	-					-			
FDP_IFF.6 (11.8.11)				X	-					-			
FDP_IRC.1 (11.9.6)													
FDP_ITC.1 (11.10.6)		O <sup>1</sup>	-	O <sup>1</sup>	-					X			
FDP_ITC.2 (11.10.7)		O <sup>1</sup>	-	O <sup>1</sup>	-					-	X	O <sup>2</sup>	O <sup>2</sup>
FDP_ITT.1 (11.11.6)		O <sup>1</sup>	-	O <sup>1</sup>	-					-			
FDP_ITT.2 (11.11.7)		O <sup>1</sup>	-	O <sup>1</sup>	-					-			
FDP_ITT.3 (11.11.8)		O <sup>1</sup>	-	O <sup>1</sup>	-	X				-			
FDP_ITT.4 (11.11.9)		O <sup>1</sup>	-	O <sup>1</sup>	-		X			-			
FDP_RIP.1 (11.12.6)													
FDP_RIP.2 (11.12.7)													
FDP_ROL.1 (11.13.6)		O <sup>1</sup>	-	O <sup>1</sup>	-					-			
FDP_ROL.2 (11.13.7)		O <sup>1</sup>	-	O <sup>1</sup>	-					-			
FDP_SDC.1 (11.14.7)													
FDP_SDC.2 (11.14.8)	X												
FDP_SDI.1 (11.15.6)													
FDP_SDI.2 (11.15.7)													
FDP_UCT.1 (11.16.6)		O <sup>2</sup>	-	O <sup>2</sup>	-					-		O <sup>1</sup>	O <sup>1</sup>
FDP_UIT.1 (11.17.6)		O <sup>1</sup>	-	O <sup>1</sup>	-					-		O <sup>2</sup>	O <sup>2</sup>
FDP_UIT.2 (11.17.7)		O <sup>1</sup>	-	O <sup>1</sup>	-			O <sup>2</sup>		-		O <sup>2</sup>	-
FDP_UIT.3 (11.17.8)		O <sup>1</sup>	-	O <sup>1</sup>	-			O <sup>2</sup>		-		O <sup>2</sup>	-

**11.2 Notes on class FDP**

This class contains families specifying requirements related to protecting user data. This class differs from class FIA (Identification and authentication) (Clause 12) and class FPT (Protection of the TSF) (Clause 15). This class specifies components to protect user data, while class FIA (Identification and authentication) (Clause 12) specifies components to protect attributes associated with the user, and class FPT (Protection of the TSF) (Clause 15) specifies components to protect TSF information.

**ISO/IEC 15408-2:2026(en)**

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 shall adequately cover these three objectives in the PP, PP-Module, functional package or 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 author of a PP, PP-Module, functional package or ST 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 policy controls access to the information, independent of the container. The attributes of the information, which can be associated with the attributes of the container (or as in the case of a multi-level database possibly are not, ) 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 in this class 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 1** 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 can 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 can serve another purpose.

**EXAMPLE 2** It is possible to build an access control policy FDP\_ACC (Access control policy) (11.3) that uses labels FDP\_IFF.1 (Simple security attributes) (11.8.6) as the basis of the access control mechanism.

A set of SFRs may encompass many SFPs, each to be identified by the two policy-oriented components FDP\_ACC (Access control policy) (11.3), and FDP\_IFC (Information flow control policy) (11.7). These policies will typically take confidentiality, integrity, and availability aspects into consideration as required, to satisfy the TOE requirements. Care should be taken to ensure that all objects are covered by at least one SFP and that there are no conflicts arising from implementing the multiple SFPs.

When building a PP, PP-Module, functional package or ST using components from the FDP (User data protection) class, the following information provides guidance on where to look and what to select from the class.

The requirements in the FDP (User data protection) class are defined in terms of a set of SFRs that will implement an SFP. Since a TOE may implement multiple SFPs simultaneously, the author of a PP, PP-Module, functional package or ST shall specify the name for each SFP, so it can be referenced in other families. This name will then be used in each component selected to indicate that it is being used as part of the definition of requirements for that SFP. This allows the author to easily indicate the scope for operations, e.g. objects covered, operations covered, authorized users.

Each instantiation of a component can apply to only one SFP. Therefore, if an SFP is specified in a component then this SFP will apply to all the elements in this component. The components may be instantiated multiple times within a PP, PP-Module, functional package or ST to account for different policies if requested.

The key to selecting components from this family is to have a well-defined set of TOE security objectives to enable proper selection of the components from the two policy components; FDP\_ACC (Access control policy) (11.3) and FDP\_IFC (Information flow control policy) (11.7). In FDP\_ACC (Access control policy) (11.3) and

## ISO/IEC 15408-2:2026(en)

FDP\_IFC (Information flow control policy) (11.7) respectively, all access control policies and all information flow control policies are named. Furthermore, the scope of control of these components is defined by the subjects, objects and operations covered by this security functionality. The names of these policies 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 "access control SFP" or an "information flow control SFP". The rules that define the functionality of the named access control and information flow control SFPs will be defined in the FDP\_ACF (Access control functions) (11.4) and FDP\_IFF (Information flow control functions) (11.8) families (respectively).

The following steps are guidance on how this class is applied in the construction of a PP, PP-Module, functional package or ST:

- identify the policies to be enforced from the FDP\_ACC (Access control policy) (11.3), and FDP\_IFC (Information flow control policy) (11.7) families. These families define scope of control for the policy, granularity of control and may identify some rules to go with the policy;
- identify the components and perform any applicable operations in the policy components. The assignment operations may be performed generally (such as with a statement "All files") or specifically (The files "A", "B", etc.) depending upon the level of detail known;
- identify any applicable function components from the FDP\_ACF (Access control functions) (11.4) and FDP\_IFF (Information flow control functions) (11.8) families to address the named policy families from FDP\_ACC (Access control policy) (11.3) and FDP\_IFC (Information flow control policy) (11.7). Perform the operations to make the components define the rules to be enforced by the named policies. This should make the components fit the requirements of the selected function envisioned or to be built;
- identify who shall have the ability to control and change security attributes under the function, such as only a security administrator, only the owner of the object, etc. Select the appropriate components from class FMT (Security management) (Clause 13) and perform the operations. Refinements can be useful here to identify missing features, such as that some or all changes shall be done via trusted path;
- identify any appropriate components from the class FMT (Security management) (Clause 13) for initial values for new objects and subjects;
- identify any applicable rollback components from the FDP\_ROL (Rollback) (11.13) family;
- identify any applicable residual information protection requirements from the FDP\_RIP (Residual information protection) (11.12) family;
- identify any applicable import or export components, and how security attributes should be handled during import and export, from the FDP\_ITC (Import from outside of the TOE) (11.10) and FDP\_ETC (Export from the TOE) (11.6) families;
- identify any applicable internal TOE communication components from the FDP\_ITT (Internal TOE transfer) (11.11) family;
- identify any requirements for integrity protection of stored information from the FDP\_SDI (Stored data integrity) (11.15);
- identify any applicable inter-TSF communication components from the FDP\_UCT (Inter-TSF user data confidentiality transfer protection) (11.16) or FDP\_UIT (Inter-TSF user data integrity transfer protection) (11.17) families.

### 11.3 Access control policy (FDP\_ACC)

#### 11.3.1 Family Behaviour

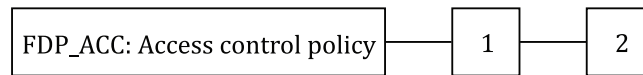
This family identifies the access control SFPs (by name) and defines the scope of control of the policies that form the identified access control portion of the SFRs related to the SFP. This scope of control is characterized by three sets: the subjects under control of the policy, the objects under control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

## ISO/IEC 15408-2:2026(en)

allow multiple policies to exist, each having a unique name. This is accomplished by iterating components from this family once for each named access control policy. The rules that define the functionality of an access control SFP will be defined by other families such as FDP\_ACF (Access control functions) (11.4) and FDP\_ETC (Export from the TOE) (11.6). The names of the access control SFPs identified here in FDP\_ACC (Access control policy) 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 "access control SFP".

### 11.3.2 Component levelling and description

Figure 24 describes the component levelling of family FDP\_ACC.



#### Key

1,2 components

**Figure 24 — FDP\_ACC: Component levelling**

FDP\_ACC.1 (Subset access control) (11.3.6) requires that each identified access control SFP be in place for a subset of the possible operations on a subset of the objects in the TOE.

FDP\_ACC.2 (Complete access control) (11.3.7) requires that each identified access control SFP cover all operations on subjects and objects covered by that SFP. It further requires that all objects and operations protected by the TSF are covered by at least one identified access control SFP.

### 11.3.3 Component management

There are no management activities foreseen related to FDP\_ACC.1 or FDP\_ACC.2.

### 11.3.4 Component audit

There are no auditable events foreseen related to FDP\_ACC.1 or FDP\_ACC.2.

### 11.3.5 Application notes

This family is based upon the concept of arbitrary controls on the interaction of subjects and objects. The scope and purpose of the controls is based upon the attributes of the accessor (subject), the attributes of the container being accessed (object), the actions (operations) and any associated access control rules.

The components in this family are capable of identifying the access control SFPs (by name) to be enforced by the traditional DAC mechanisms. It further defines the subjects, objects and operations that are covered by identified access control SFPs. The rules that define the functionality of an access control SFP will be defined by other families, such as FDP\_ACF (Access control functions) (11.4) and FDP\_ETC (Export from the TOE) (11.6). The names of the access control SFPs defined in FDP\_ACC (Access control policy) 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 "access control SFP".

The access control SFP covers a set of triplets: subject, object, and operations. Therefore, a subject can be covered by multiple access control SFPs but only with respect to a different operation or a different object. Of course, the same applies to objects and operations.

A critical aspect of an access control function that enforces an access control SFP is the ability for users to modify the attributes involved in access control decisions. The FDP\_ACC (Access control policy) family does not address these aspects. Some of these requirements are left undefined, but can be added as refinements, while others are covered elsewhere in other families and classes such as class FMT (Security management) (Clause 13).

**ISO/IEC 15408-2:2026(en)**

There are no audit requirements in FDP\_ACC (Access control policy) as this family specifies access control SFP requirements. Audit requirements will be found in families specifying functions to satisfy the access control SFPs identified in this family.

This family provides a PP, PP-Module, functional package or ST author the capability to specify several policies, for example, a fixed access control SFP to be applied to one scope of control, and a flexible access control SFP to be defined for a different scope of control. To specify more than one access control policy, the components from this family can be iterated multiple times in a PP, PP-Module, functional package or ST to different subsets of operations and objects. This will accommodate TOEs that contain multiple policies, each addressing a particular set of operations and objects. In other words, the author of a PP, PP-Module, functional package or ST should specify the required information in the ACC component for each of the access control SFPs that the TSF will enforce. For example, a TOE incorporating three access control SFPs, each covering only a subset of the objects, subjects, and operations within the TOE, will contain one FDP\_ACC.1 (Subset access control) ([11.3.6](#)) component for each of the three access-control SFPs, necessitating a total of three FDP\_ACC.1 (Subset access control) ([11.3.6](#)) components.

**11.3.6 FDP\_ACC.1 Subset access control****11.3.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FDP\_ACF.1 (Security attribute-based access control) ([11.4.6](#))

**11.3.6.2 Component rationale**

The terms object and subject refer to generic elements in the TOE. For a policy to be implementable, the entities shall be clearly identified. For a PP, the objects and operations can be expressed as types such as: named objects, data repositories, observe accesses, etc. For a specific TOE these generic terms (subject, object) shall be refined.

EXAMPLE Examples of objects and subjects include files, registers, ports, daemons, and open calls.

This component specifies that the policy cover some well-defined set of operations on some subset of the objects. It places no constraints on any operations outside the set - including operations on objects for which other operations are controlled.

**11.3.6.3 Element FDP\_ACC.1.1**

**The TSF shall enforce the [assignment <sup>(a1)</sup>: access control SFP] on [assignment <sup>(a2)</sup>: list of subjects, objects, and operations among subjects and objects covered by the SFP].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify a uniquely named access control SFP to be enforced by the TSF.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the list of subjects, objects, and operations among subjects and objects covered by the SFP.

**11.3.7 FDP\_ACC.2 Complete access control****11.3.7.1 Component relationships**

Hierarchical to: FDP\_ACC.1 (Subset access control) ([11.3.6](#))

Dependencies: FDP\_ACF.1 (Security attribute-based access control) ([11.4.6](#))

## ISO/IEC 15408-2:2026(en)

### 11.3.7.2 Component rationale

This component requires that all possible operations on objects, that are included in the SFP, are covered by an access control SFP.

The author of a PP, PP-Module, functional package or ST shall demonstrate that each combination of objects and subjects is covered by an access control SFP.

### 11.3.7.3 Element FDP\_ACC.2.1

The TSF shall enforce the [assignment <sup>(a1)</sup>: *access control SFP*] on [assignment <sup>(a2)</sup>: *list of subjects and objects*] **and all** operations among subjects and objects covered by the **SFP**.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify a uniquely named access control SFP to be enforced by the TSF.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the list of subjects and objects covered by the SFP. All operations among those subjects and objects will be covered by the SFP.

### 11.3.7.4 Element 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.**

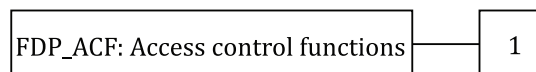
## 11.4 Access control functions (FDP\_ACF)

### 11.4.1 Family Behaviour

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

### 11.4.2 Component levelling and description

[Figure 25](#) describes the component levelling of family FDP\_ACF.



#### Key

1 component

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

FDP\_ACF.1 (Security attribute-based access control) ([11.4.6](#)) allows the TSF to enforce access control 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.

### 11.4.3 Component management

The following actions related to FDP\_ACF.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- managing the attributes used to make explicit access or denial-based decisions.

**ISO/IEC 15408-2:2026(en)****11.4.4 Component audit**

The following actions related to FDP\_ACF.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: successful requests to perform an operation on an object covered by the SFP;
- basic: all requests to perform an operation on an object covered by the SFP;
- detailed: the specific security attributes used in making an access check.

**11.4.5 Application notes**

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

This family provides a PP, PP-Module, functional package or ST author the capability to describe the rules for access control. This results in a TOE where the access to objects will not change.

**EXAMPLE 1** An example of such an object is "Message of the Day", which is readable by all, and changeable only by the authorized administrator.

This family also provides the author of a PP, PP-Module, functional package or ST with the ability to describe rules that provide for exceptions to the general access control rules. Such exceptions would either explicitly allow or deny authorization to access an object.

There are no explicit components to specify other possible functions such as two-person control, sequence rules for operations, or exclusion controls. However, these mechanisms, as well as traditional DAC mechanisms, can be represented with the existing components, by careful drafting of the access control rules.

**EXAMPLE 2** A variety of acceptable access control functionality can be specified in this family.

- access control lists (ACLs);
- time-based access control specifications;
- origin-based access control specifications;
- owner-controlled access control attributes.

**11.4.6 FDP\_ACF.1 Security attribute-based access control****11.4.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FDP\_ACC.1 (Subset access control) ([11.3.6](#))  
FMT\_MSA.3 (Static attribute initialization) ([13.5.8](#))

**11.4.6.2 Component rationale**

This component provides requirements for a mechanism that mediates access control based on security attributes associated with subjects and objects. Each object and subject have a set of associated attributes, such as location, time of creation, access rights such as Access Control Lists (ACLs). This component allows the author of a PP, PP-Module, functional package or ST to specify the attributes that will be used for the access control mediation. This component allows access control rules, using these attributes, to be specified.

**EXAMPLE** Examples of the attributes that a PP, PP-Module, functional package or ST author can assign are:

**ISO/IEC 15408-2:2026(en)**

- an identity attribute can be associated with users, subjects, or objects to be used for mediation. Examples of such attributes can 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 can 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 can 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 is 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.

This component also provides requirements for the access control security functions to be able to explicitly authorize or deny access to an object based upon security attributes. This can be used to provide privilege, access rights, or access authorizations within the TOE. Such privileges, rights, or authorizations can apply to users, subjects (representing users or applications), and objects.

**11.4.6.3 Element FDP\_ACF.1.1**

**The TSF shall enforce the [assignment (a1): *access control SFP*] to objects based on the following: [assignment (a2): *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify an access control SFP name that the TSF is to enforce. The name of the access control SFP, and the scope of control for that policy are defined in components from FDP\_ACC (Access control policy).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify, for each controlled subject and object, the security attributes and/or named groups of security attributes that the function will use in the specification of the rules.

**11.4.6.4 Element FDP\_ACF.1.2**

**The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment (a1): *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the SFP rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects. These rules specify when access is granted or denied. It can specify general access control functions or granular access control functions.

**11.4.6.5 Element FDP\_ACF.1.3**

**The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment (a1): *rules, based on security attributes, that explicitly authorize access of subjects to objects*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the rules, based on security attributes, that explicitly authorize access of subjects to objects that will be used to explicitly authorize access. These rules are in addition to those specified in FDP\_ACF.1.1. They are included in FDP\_ACF.1.3 as they are intended to contain exceptions to the rules in FDP\_ACF.1.1.

**11.4.6.6 Element FDP\_ACF.1.4**

**The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [assignment (a1): *rules, based on security attributes, that explicitly deny access of subjects to objects*].**

## ISO/IEC 15408-2:2026(en)

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the rules, based on security attributes, that explicitly deny access of subjects to objects. These rules are in addition to those specified in FDP\_ACF.1.1. They are included in FDP\_ACF.1.4 as they are intended to contain exceptions to the rules in FDP\_ACF.1.1. An example of rules to explicitly deny access is based on a privilege vector associated with a subject that always denies access to objects covered by the access control SFP that has been specified. If such a capability is not desired, then the author of a PP, PP-Module, functional package or ST should specify "none".

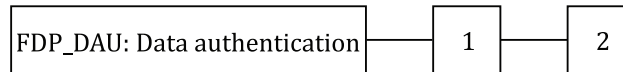
### 11.5 Data authentication (FDP\_DAU)

#### 11.5.1 Family Behaviour

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

#### 11.5.2 Component levelling and description

[Figure 26](#) depicts the component levelling of family FDP\_DAU.



#### Key

1,2 components

**Figure 26 — FDP\_DAU: Component levelling**

FDP\_DAU.1 (Basic Data Authentication) ([11.5.6](#)) requires that the TSF is capable of generating a guarantee of authenticity of the information content of objects.

FDP\_DAU.2 (Data Authentication with Identity of Guarantor) ([11.5.7](#)) additionally requires that the TSF is capable of establishing the identity of the subject who provided the guarantee of authenticity.

#### 11.5.3 Component management

The following actions related to FDP\_DAU.1 and FDP\_DAU.2 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the assignment or modification of the objects for which data authentication may apply can be configurable.

#### 11.5.4 Component audit

The following actions related to FDP\_DAU.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: successful generation of validity evidence;
- basic: unsuccessful generation of validity evidence;
- detailed: the identity of the subject that requested the evidence.

The following actions related to FDP\_DAU.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: successful generation of validity evidence;
- basic: unsuccessful generation of validity evidence;

## ISO/IEC 15408-2:2026(en)

- detailed: the identity of the subject that requested the evidence;
- detailed: the identity of the subject that generated the evidence.

### 11.5.5 Application notes

This family describes specific functions that can be used to authenticate "static" data.

Components in this family should be used when there is a requirement for "static" data authentication, i.e. where data is required to be signed but not transmitted.

NOTE The non-repudiation of origin FCO\_NRO (Non-repudiation of origin) family provides for non-repudiation of origin of information received during a data exchange.

### 11.5.6 FDP\_DAU.1 Basic Data Authentication

#### 11.5.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 11.5.6.2 Component rationale

This component may be satisfied by one-way hash functions to generate a hash value for a definitive document that may be used as verification of the validity or authenticity of its information content.

EXAMPLE Cryptographic checksum, fingerprint, message digest.

#### 11.5.6.3 Element FDP\_DAU.1.1

**The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of [assignment <sup>(a1)</sup>: *list of objects or information types*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of objects or information types for which the TSF shall be capable of generating data authentication evidence.

#### 11.5.6.4 Element FDP\_DAU.1.2

**The TSF shall provide [assignment <sup>(a1)</sup>: *list of subjects*] with the ability to verify evidence of the validity of the indicated information.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of subjects that shall have the ability to verify data authentication evidence for the objects identified in the previous element. The list of subjects can be very specific, if the subjects are known, or it can be more generic and refer to a "type" of subject such as an identified role.

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

#### 11.5.7.1 Component relationships

Hierarchical to: FDP\_DAU.1 (Basic Data Authentication) ([11.5.6](#))

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

### 11.5.7.2 Component rationale

This component additionally requires the ability to verify the identity of the user that provided the guarantee of authenticity.

EXAMPLE A trusted third party.

### 11.5.7.3 Element FDP\_DAU.2.1

The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of [assignment <sup>(a1)</sup>: *list of objects or information types*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of objects or information types for which the TSF shall be capable of generating data authentication evidence.

### 11.5.7.4 Element FDP\_DAU.2.2

The TSF shall provide [assignment <sup>(a1)</sup>: *list of subjects*] with the ability to verify evidence of the validity of the indicated information **and the identity of the user that generated the evidence.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of subjects that shall have the ability to verify data authentication evidence for the objects identified in the previous element as well as the identity of the user that created the data authentication evidence.

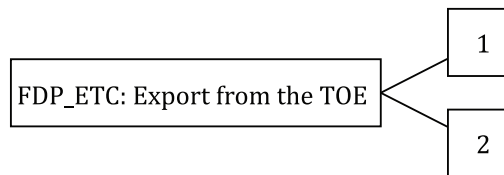
## 11.6 Export from the TOE (FDP\_ETC)

### 11.6.1 Family Behaviour

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

### 11.6.2 Component levelling and description

[Figure 27](#) describes the component levelling of family FDP\_ETC.



#### Key

1,2 components

**Figure 27 — FDP\_ETC: Component levelling**

FDP\_ETC.1 (Export of user data without security attributes) ([11.6.6](#)) requires that the TSF enforces the appropriate SFPs when exporting user data outside the TSF. User data that is exported by this function is exported without its associated security attributes.

FDP\_ETC.2 (Export of user data with security attributes) ([11.6.7](#)) requires that the TSF enforces the appropriate SFPs using a function that accurately and unambiguously associates security attributes with the user data that is exported.

**ISO/IEC 15408-2:2026(en)****11.6.3 Component management**

The following actions related to FDP\_ETC.2 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the additional exportation control rules can be configurable by a user in a defined role.

There are no management activities foreseen related to FDP\_ETC.1.

**11.6.4 Component audit**

The following actions related to FDP\_ETC.1 and FDP\_ETC.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: successful export of information;
- basic: all attempts to export information.

**11.6.5 Application notes**

This family defines functions for TSF-mediated exporting of user data from the TOE such that its security attributes either can be explicitly preserved or can be ignored once it has been exported. Consistency of these security attributes are addressed by FPT\_TDC (Inter-TSF TSF data consistency) ([15.15](#)).

FDP\_ETC (Export from the TOE) is concerned with limitations on export and association of security attributes with the exported user data.

This family, and the corresponding Import family FDP\_ITC (Import from outside of the TOE) ([11.10](#)), address how the TOE deals with user data transferred into and outside its control. In principle, this family is concerned with the TSF-mediated exporting of user data and its related security attributes.

A variety of activities can be involved here:

- exporting of user data without any security attributes;
- exporting user data including security attributes where the two are associated with one another and the security attributes unambiguously represent the exported user data.

If there are multiple SFPs (access control and/or information flow control) then it may be appropriate to iterate these components once for each named SFP.

**11.6.6 FDP\_ETC.1 Export of user data without security attributes****11.6.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]

**11.6.6.2 Component rationale**

This component is used to specify the TSF-mediated exporting of user data without the export of its security attributes.

**11.6.6.3 Element FDP\_ETC.1.1**

**The TSF shall enforce the [assignment (a<sup>1</sup>): *access control SFP(s) and/or information flow control SFP(s)*] when exporting user data, controlled under the SFP(s), outside of the TOE.**

**ISO/IEC 15408-2:2026(en)**

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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.

**11.6.6.4 Element FDP\_ETC.1.2**

**The TSF shall export the user data without the user data's associated security attributes.**

**11.6.7 FDP\_ETC.2 Export of user data with security attributes****11.6.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]

**11.6.7.2 Component rationale**

The user data is exported together with its security attributes. The security attributes are unambiguously associated with the user data. There are several ways of achieving this association. One way that this can be achieved is by physically collocating the user data and the security attributes.

EXAMPLE On the same external media.

An alternative way is by using cryptographic techniques such as secure signatures to associate the attributes and the user data. FTP\_ITC (Inter-TSF trusted channel) ([18.3](#)) can be used to assure that the attributes are correctly received at the other trusted IT product while FPT\_TDC (Inter-TSF TSF data consistency) ([15.15](#)) can be used to make sure that those attributes are properly interpreted. Furthermore, FTP\_TRP (Trusted path) ([18.5](#)) can be used to make sure that the export is being initiated by the proper user.

**11.6.7.3 Element FDP\_ETC.2.1**

**The TSF shall enforce the [assignment (a1): *access control SFP(s) and/or information flow control SFP(s)*] when exporting user data, controlled under the SFP(s), outside of the TOE.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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.

**11.6.7.4 Element FDP\_ETC.2.2**

**The TSF shall export the user data with the user data's associated security attributes.**

**11.6.7.5 Element FDP\_ETC.2.3**

**The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.**

**11.6.7.6 Element FDP\_ETC.2.4**

**The TSF shall ensure that interpretation of the security attributes of the exported user data is as intended by the owner of the user data.**

**11.6.7.7 Element FDP\_ETC.2.5**

**The TSF shall enforce the following rules when user data is exported from the TOE: [assignment (a1): *additional exportation control rules*].**

## ISO/IEC 15408-2:2026(en)

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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.

### 11.7 Information flow control policy (FDP\_IFC)

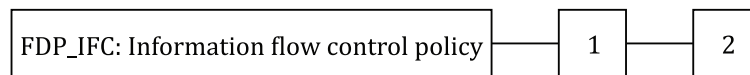
#### 11.7.1 Family Behaviour

This family identifies the information flow control SFPs (by name) and defines the scope of control for each named information flow control SFP. This scope of control is characterized by three sets: 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 covered by the policy. The criteria allow multiple policies to exist, each having a unique name. This is accomplished by iterating components from this family once for each named information flow control policy. The rules that define the functionality of an information flow control SFP will be defined by other families such as FDP\_IFF (Information flow control functions) (11.8) and FDP\_ETC (Export from the TOE). The names of the information flow control SFPs identified here in FDP\_IFC (Information flow control policy) 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".

The TSF mechanism controls the flow of information in accordance with the information flow control SFP. Operations that would change the security attributes of information are not generally permitted as this would be in violation of an information flow control SFP. However, such operations can be permitted as exceptions to the information flow control SFP if explicitly specified.

#### 11.7.2 Component levelling and description

Figure 28 describes the component levelling of family FDP\_IFC.



#### Key

1,2 components

Figure 28 — FDP\_IFC: Component levelling

FDP\_IFC.1 (Subset information flow control) (11.7.6) requires that each identified information flow control SFPs be in place for a subset of the possible operations on a subset of information flows in the TOE.

FDP\_IFC.2 (Complete information flow control) (11.7.7) requires that each identified information flow control SFP cover all operations on subjects and information covered by that SFP. It further requires that all information flows and operations controlled by the TSF are covered by at least one identified information flow control SFP.

#### 11.7.3 Component management

There are no management activities foreseen related to FDP\_IFC.1 or FDP\_IFC.2.

#### 11.7.4 Component audit

There are no auditable events foreseen related to FDP\_IFC.1 or FDP\_IFC.2.

#### 11.7.5 Application notes

This family covers the identification of information flow control SFPs and, for each, specifies the scope of control of the SFP.

**ISO/IEC 15408-2:2026(en)**

The components in this family are capable of identifying the information flow control SFPs to be enforced by the traditional MAC 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 FDP\_IFF (Information flow control functions) ([11.8](#)) and FDP\_ETC (Export from the TOE). The information flow control SFPs named here in FDP\_IFC (Information flow control policy) are intended 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 can 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 can 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 is possible that it does 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. In this case the desired IT security functions can be specified as supportive of those objectives.

In the second component FDP\_IFC.2 (Complete information flow control) ([11.7.7](#)), 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 are required to be covered by an 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 shall 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 might 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 can 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 can be expressed as types, e.g. email, data repositories, observe accesses.

The components in this family can be applied multiple times in a PP, PP-Module, functional package or 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.

## **11.7.6 FDP\_IFC.1 Subset information flow control**

### **11.7.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FDP\_IFF.1 (Simple security attributes) ([11.8.6](#))

**ISO/IEC 15408-2:2026(en)****11.7.6.2 Component rationale**

This component requires that an information flow control policy apply to a subset of the possible operations in the TOE.

**11.7.6.3 Element FDP\_IFC.1.1**

**The TSF shall enforce the [assignment (a1): *information flow control SFP*] on [assignment (a2): *list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify a uniquely named information flow control SFP to be enforced by the TSF.

In assignment (a2), the author of a PP, PP-Module, functional package or ST 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 can be at various levels of detail depending on the needs of the author of a PP, PP-Module, functional package or ST.

**11.7.7 FDP\_IFC.2 Complete information flow control****11.7.7.1 Component relationships**

Hierarchical to: FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))

Dependencies: FDP\_IFF.1 (Simple security attributes) ([11.8.6](#))

**11.7.7.2 Component rationale**

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 author of a PP, PP-Module, functional package or ST shall demonstrate that each combination of information flows and subjects is covered by an information flow control SFP.

**11.7.7.3 Element FDP\_IFC.2.1**

**The TSF shall enforce the [assignment (a1): *information flow control SFP*] on [assignment (a2): *list of subjects and information*] and all operations that cause **that** information to flow to and from subjects covered by the SFP.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify a uniquely named information flow control SFP to be enforced by the TSF.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the list of subjects and information that will be covered by the SFP. All operations that cause that information to flow to and from subjects will be covered by the SFP. As mentioned above, the list of subjects can be at various levels of detail depending on the needs of the author of a PP, PP-Module, functional package or ST.

**11.7.7.4 Element FDP\_IFC.2.2**

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

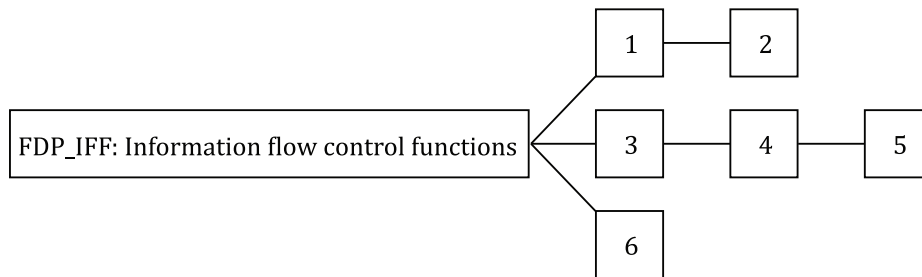
## 11.8 Information flow control functions (FDP\_IFF)

### 11.8.1 Family Behaviour

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

### 11.8.2 Component levelling and description

[Figure 29](#) depicts the component levelling of family FDP\_IFF.



#### Key

1,2,3,4,5,6 components

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

FDP\_IFF.1 (Simple security attributes) ([11.8.6](#)) 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) ([11.8.7](#)) expands on the requirements of FDP\_IFF.1 (Simple security attributes) ([11.8.6](#)) 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 than or equal to (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 than or equal to than any other element of the lattice.

FDP\_IFF.3 (Limited illicit information flows) ([11.8.8](#)) requires the SFP to cover illicit information flows but does not necessarily eliminate them.

FDP\_IFF.4 (Partial elimination of illicit information flows) ([11.8.9](#)) requires the SFP to cover the elimination of some (but does not necessarily all) illicit information flows.

FDP\_IFF.5 (No illicit information flows) ([11.8.10](#)) requires SFP to cover the elimination of all illicit information flows.

FDP\_IFF.6 (Illicit information flow monitoring) ([11.8.11](#)) requires the SFP to monitor illicit information flows for specified and maximum capacities.

## ISO/IEC 15408-2:2026(en)

### 11.8.3 Component management

The following actions related to FDP\_IFF.1 and FDP\_IFF.2 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- managing the attributes used to make explicit access-based decisions.

The following actions related to FDP\_IFF.6 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the enabling or disabling of the monitoring function;
- modification of the maximum capacity at which the monitoring occurs.

There are no management activities foreseen related to FDP\_IFF.3, FDP\_IFF.4 or FDP\_IFF.5.

### 11.8.4 Component audit

The following actions related to FDP\_IFF.1, FDP\_IFF.2 and FDP\_IFF.5 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: decisions to permit requested information flows;
- basic: all decisions on requests for information flow;
- detailed: the specific security attributes used in making an information flow enforcement decision;
- detailed: some specific subsets of the information that has flowed based upon policy goals.

The following actions related to FDP\_IFF.3, FDP\_IFF.4 and FDP\_IFF.6 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: decisions to permit requested information flows;
- basic: all decisions on requests for information flow;
- basic: the use of identified illicit information flow channels;
- detailed: the specific security attributes used in making an information flow enforcement decision;
- detailed: some specific subsets of the information that has flowed based upon policy goals;
- detailed: the use of identified illicit information flow channels with estimated maximum capacity exceeding a specified value.

### 11.8.5 Application notes

This family describes the rules for the specific functions that can implement the information flow control SFPs named in FDP\_IFC (Information flow control policy), which also specifies the scope of control of the policies. It consists of two "trees": one addressing the common information flow control function issues, and a second addressing illicit information flows (i.e. covert channels) with respect to one or more information flow control SFPs. This division arises because the issues concerning illicit information flows are, in some sense, orthogonal to the rest of an SFP. Illicit information flows are flows in violation of policy; thus, they are not a policy issue.

In order to implement strong protection against disclosure or modification in the face of untrusted software, controls on information flow are required. Access controls alone are not sufficient because they only control access to containers, allowing the information they contain to flow, without controls, throughout a system.

In this family, the phrase "types of illicit information flows" is used. This phrase may be used to refer to the categorization of flows as "Storage Channels" or "Timing Channels", or it can refer to improved categorizations reflective of the needs of a PP, PP-Module, functional package or ST author.

**ISO/IEC 15408-2:2026(en)**

The flexibility of these components allows the definition of a privilege policy within FDP\_IFF.1 (Simple security attributes) (11.8.6) and FDP\_IFF.2 (Hierarchical security attributes) (11.8.7) to allow the controlled bypass of all or part of a particular SFP. If there is a need for a predefined approach to SFP bypass, the author of a PP, PP-Module, functional package or ST should consider incorporating a privilege policy.

**11.8.6 FDP\_IFF.1 Simple security attributes****11.8.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FDP\_IFC.1 (Subset information flow control) (11.7.6)  
FMT\_MSA.3 (Static attribute initialization) (13.5.8)

**11.8.6.2 Component rationale**

This component requires security attributes on information, and on subjects that cause that information to flow and subjects that act as recipients of that information. The attributes of the containers of the information should also be considered if it is desired that they should play a part in information flow control decisions or if they are covered by an access control policy. This component specifies the key rules that are enforced and describes how security attributes are derived.

This component does not specify the details of how a security attribute is assigned (i.e. user versus process). Flexibility in policy is provided by having assignments that allow specification of additional policy and function requirements, as necessary.

This component also provides requirements for the information flow control functions to be able to explicitly authorize and deny an information flow based upon security attributes. This can be used to implement a privilege policy that covers exceptions to the basic policy defined in this component.

**11.8.6.3 Element FDP\_IFF.1.1**

**The TSF shall enforce the [assignment (a1): *information flow control SFP*] based on the following types of subject and information security attributes: [assignment (a2): *list of subjects and information controlled under the indicated SFP, and for each, the security attributes*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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 FDP\_IFC (Information flow control policy).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify, for each type of controlled subject and information, the security attributes that are relevant to the specification of the SFP rules.

**11.8.6.4 Element FDP\_IFF.1.2**

**The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [assignment (a1): *for each operation, the security attribute-based relationship that hold between subject and information security attributes*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify for each operation, the security attribute-based relationship that holds between subject and information security attributes that the TSF will enforce.

**11.8.6.5 Element FDP\_IFF.1.3**

**The TSF shall enforce the [assignment (a1): *additional information flow control SFP rules*].**

**ISO/IEC 15408-2:2026(en)**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify any additional information flow control SFP rules that the TSF is to enforce. This includes all rules of the SFP that are either not based on the security attributes of the information and the subject or rules that automatically modify the security attributes of information or subjects as a result of an access operation. An example for the first case is a rule of the SFP controlling a threshold value for specific types of information. 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. Some information flow policies for example may limit the number of access operations to information with specific security attributes. If there are no additional rules then the author of a PP, PP-Module, functional package or ST should specify "none".

**11.8.6.6 Element FDP\_IFF.1.4**

**The TSF shall explicitly authorize an information flow based on the following rules: [assignment (a1): rules, based on security attributes, that explicitly authorize information flows].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the rules, based on security attributes, that explicitly authorize information flows. These rules are in addition to those specified in the preceding elements. They are included in FDP\_IFF.1.4 as they are intended to contain exceptions to the rules in the preceding elements.

**11.8.6.7 Element FDP\_IFF.1.5**

**The TSF shall explicitly deny an information flow based on the following rules: [assignment (a1): rules, based on security attributes, that explicitly deny information flows].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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.1.5 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 author of a PP, PP-Module, functional package or ST should specify "none".

**11.8.7 FDP\_IFF.2 Hierarchical security attributes****11.8.7.1 Component relationships**

Hierarchical to: FDP\_IFF.1 (Simple security attributes) ([11.8.6](#))

Dependencies: FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))  
FMT\_MSA.3 (Static attribute initialization) ([13.5.8](#))

**11.8.7.2 Component rationale**

This component requires that the named information flow control SFP uses hierarchical security attributes that form a lattice.

It is important to note that the hierarchical relationship requirements identified in FDP\_IFF.2.4 need only apply to the information flow control security attributes for the information flow control SFPs that have been identified in FDP\_IFF.2.1. This component is not meant to apply to other SFPs such as access control SFPs.

FDP\_IFF.2.6 phrases the requirements for the set of security attributes to form a lattice. A number of information flow policies defined in the literature and implemented in IT products are based on a set of security attributes that form a lattice. FDP\_IFF.2.6 is specifically included to address this type of information flow policies.

**ISO/IEC 15408-2:2026(en)**

If it is the case that multiple information flow control SFPs are specified, and that each of these SFPs have their own security attributes that are not related to one another, then the author of a PP, PP-Module, functional package or ST shall iterate this component once for each of those SFPs. Otherwise, a conflict can arise with the sub-items of FDP\_IFF.2.4 since the required relationships will not exist.

**11.8.7.3 Element FDP\_IFF.2.1**

The TSF shall enforce the [assignment <sup>(a1)</sup>: *information flow control SFP*] based on the following types of subject and information security attributes: [assignment <sup>(a2)</sup>: *list of subjects and information controlled under the indicated SFP, and for each, the security attributes*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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 FDP\_IFC (Information flow control policy).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify, for each type of controlled subject and information, the security attributes that are relevant to the specification of the SFP rules. For example, such security attributes may be things such the subject identifier, subject sensitivity label, subject clearance label, information sensitivity label, etc. The types of security attributes should be sufficient to support the environmental needs.

**11.8.7.4 Element FDP\_IFF.2.2**

The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules, **based on the ordering relationships between security attributes** hold: [assignment <sup>(a1)</sup>: *for each operation, the security attribute-based relationship that shall hold between subject and information security attributes*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify for each operation, the security attribute-based relationship that holds between a subject and the information security attributes that the TSF will enforce. These relationships should be based upon the ordering relationships between the security attributes.

**11.8.7.5 Element FDP\_IFF.2.3**

The TSF shall enforce the [assignment <sup>(a1)</sup>: *additional information flow control SFP rules*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify any additional information flow control SFP rules that the TSF is to enforce. This includes all rules of the SFP that are either not based on the security attributes of the information and the subject or rules that automatically modify the security attributes of information or subjects as a result of an access operation. An example for the first case is a rule of the SFP controlling a threshold value for specific types of information.

**11.8.7.6 Element FDP\_IFF.2.4**

The TSF shall explicitly authorize an information flow based on the following rules: [assignment <sup>(a1)</sup>: *rules, based on security attributes, that explicitly authorize information flows*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the rules, based on security attributes, that explicitly authorize information flows. These rules are in addition to those specified in the preceding elements. They are included in FDP\_IFF.2.4 as they are intended to contain exceptions to the rules in the preceding elements.

**11.8.7.7 Element FDP\_IFF.2.5**

The TSF shall explicitly deny an information flow based on the following rules: [assignment <sup>(a1)</sup>: *rules, based on security attributes, that explicitly deny information flows*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the rules, based on security attributes, that explicitly deny information flows. These rules are in addition to those specified

## ISO/IEC 15408-2:2026(en)

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 author of a PP, PP-Module, functional package or ST should specify "none".

### 11.8.7.8 Element FDP\_IFF.2.6

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

- **there exists an ordering function that, given two valid security attributes, determines if the security attributes are equal, if one security attribute is greater than the other, or if the security attributes are incomparable;**
- **there exists a "least upper bound" in the set of security attributes, such that, given any two valid security attributes, there is a valid security attribute that is greater than or equal to the two valid security attributes;**
- **there exists a "greatest lower bound" in the set of security attributes, such that, given any two valid security attributes, there is a valid security attribute that is not greater than the two valid security attributes.**

### 11.8.8 FDP\_IFF.3 Limited illicit information flows

#### 11.8.8.1 Component relationships

Hierarchical to: No other components.

Dependencies: FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))

#### 11.8.8.2 Component rationale

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, PP-Module, functional package or ST author has the ability to specify whether the illicit information flows must be audited.

#### 11.8.8.3 Element FDP\_IFF.3.1

**The TSF shall enforce the [assignment <sup>(a1)</sup>: *information flow control SFP*] to limit the capacity of [assignment <sup>(a2)</sup>: *types of illicit information flows*] to a [assignment <sup>(a3)</sup>: *maximum capacity*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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 FDP\_IFC (Information flow control policy).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the types of illicit information flows that are subject to a maximum capacity limitation.

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify the maximum capacity permitted for any identified illicit information flows.

## ISO/IEC 15408-2:2026(en)

### 11.8.9 FDP\_IFF.4 Partial elimination of illicit information flows

#### 11.8.9.1 Component relationships

Hierarchical to: FDP\_IFF.3 (Limited illicit information flows) ([11.8.8](#))

Dependencies: FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))

#### 11.8.9.2 Component rationale

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.

#### 11.8.9.3 Element FDP\_IFF.4.1

The TSF shall enforce the [assignment <sup>(a1)</sup>: *information flow control SFP*] to limit the capacity of [assignment <sup>(a2)</sup>: *types of illicit information flows*] to a [assignment <sup>(a3)</sup>: *maximum capacity*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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 FDP\_IFC (Information flow control policy).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the types of illicit information flows which are subject to a maximum capacity limitation.

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify the maximum capacity permitted for any identified illicit information flows.

#### 11.8.9.4 Element FDP\_IFF.4.2

**The TSF shall prevent [assignment <sup>(a1)</sup>: *types of illicit information flows*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the types of illicit information flows to be eliminated. This list shall not be empty as this component requires that some illicit information flows be eliminated.

### 11.8.10 FDP\_IFF.5 No illicit information flows

#### 11.8.10.1 Component relationships

Hierarchical to: FDP\_IFF.4 (Partial elimination of illicit information flows) ([11.8.9](#))

Dependencies: FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))

#### 11.8.10.2 Component rationale

This component should be used when the SFPs that require control of illicit information flows require elimination of all illicit information flows. However, the author of a PP, PP-Module, functional package or ST should carefully consider the potential impact that eliminating all illicit information flows can have on the normal functional operation of the TOE. Many practical applications have shown that there is an indirect relationship between illicit information flows and normal functionality within a TOE and eliminating all illicit information flows may result in less than desired functionality.

#### 11.8.10.3 Element FDP\_IFF.5.1

The TSF shall **ensure that no illicit information flows exist to circumvent** [assignment <sup>(a1)</sup>: *name of information flow control SFP*].

**ISO/IEC 15408-2:2026(en)**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the information flow control SFP for which illicit information flows are eliminated. The name of the information flow control SFP, and the scope of control for that policy are defined in components from FDP\_IFC (Information flow control policy).

**11.8.11 FDP\_IFF.6 Illicit information flow monitoring****11.8.11.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))

**11.8.11.2 Component rationale**

This component should be used when it is desired that the TSF provide the ability to monitor the use of illicit information flows that exceed a specified capacity. If it is desired that such flows be audited, then this component can serve as the source of audit events to be used by components from the FAU\_GEN (Security audit data generation) family.

**11.8.11.3 Element FDP\_IFF.6.1**

**The TSF shall enforce the [assignment <sup>(a1)</sup>: *information flow control SFP*] to monitor [assignment <sup>(a2)</sup>: *types of illicit information flows*] when it exceeds the [assignment <sup>(a3)</sup>: *maximum capacity*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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 FDP\_IFC (Information flow control policy).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the types of illicit information flows that will be monitored for exceeding a maximum capacity.

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify the maximum capacity above which illicit information flows will be monitored by the TSF.

**11.9 Information retention control (FDP\_IRC)****11.9.1 Family Behaviour**

The "Information retention control" family addresses a basic need in secure information processing and storage applications for the secure management of data no longer needed by the TOE to perform its operations, but that is still stored in the TOE.

The historical view of IT systems as data storage systems suggested that once entered, data would seldom be deleted from the system, and if it was deleted, this would mainly be because of storage exhaustion problems.

However, in a multilateral or high security environment it is important to minimize the replication of data, as well as the time period during which data is stored in the system. It is also possible that users can want their IT products to avoid retaining sensitive data that they consider to be exploitable by third parties or that can threaten privacy. FDP\_IRC (Information retention control) may help users to gain confidence that the product is secure by deleting every copy of the data when it is no longer needed.

The FDP\_RIP (Residual information protection) ([11.12](#)) family addresses one side of this problem, but an explicit requirement on the management of data that is no longer needed is missing.

Of course, competing requirements can arise, since some data can be needed by the system for more operations over a longer time period. Possible solutions to this problem are:

- better protecting the information objects stored in the TOE from access;

## ISO/IEC 15408-2:2026(en)

- re-requesting the protected information from the user each time it is needed.

Information retention control ensures, that data no longer necessary for the operation of the TOE is deleted by the TOE. Components of this family require the author of a PP, PP-Module, functional package or ST to identify the TOE operations, including both simple and complex processing, and the information objects not to be kept in the TOE subject to those operations.

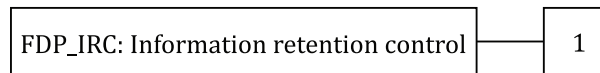
The TOE is also required to keep track of such stored information objects, and to delete both the on-line and the off-line information objects that are no longer required.

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 are detailed elsewhere, and are in the domain of the environmental objectives and organizational policies, not of the PP, PP-Module, functional package or ST.

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

### 11.9.2 Component levelling and description

[Figure 30](#) illustrates the component levelling of family FDP\_IRC.



#### Key

1 component

**Figure 30 — FDP\_IRC: Component levelling**

FDP\_IRC.1 (Information retention control) ([11.9.6](#)) requires that the TSF ensure that any copy of a defined set of objects in the TOE is deleted when no longer strictly necessary for the operation of the TOE.

### 11.9.3 Component management

There are no management activities foreseen related to FDP\_IRC.1.

### 11.9.4 Component audit

There are no auditable events foreseen related to FDP\_IRC.1.

### 11.9.5 Application notes

While a great aspect of the elimination of the objects as required by FDP\_IRC (Information retention control) refers to the information stored within the object as a container, it also includes all attributes (also in the meaning of metadata) that may be associated with the object.

In this aspect, the focus of FDP\_IRC (Information retention control) differs from other components related to access or information flow control policies, such as FDP\_IFF (Information flow control functions) and FDP\_IFC (Information flow control policy). More important, objects here are always considered in the context of selected activities that are performed on these objects. In contrast to residual information protection FDP\_RIP (Residual information protection) ([11.12](#)), FDP\_IRC (Information retention control) excludes objects from any access or information flow and deletes them, irreversibly and untraceably when they are no longer needed by a set of activities.

While it is possible that it is not completely clear, which objects to consider, it is essential that the list of objects is assigned by the author of a PP, PP-Module, functional package or ST at the very latest in order to allow for concrete tests. In any case the list of objects shall be derived from a structured analysis.

**ISO/IEC 15408-2:2026(en)****11.9.6 FDP\_IRC.1 Information retention control****11.9.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**11.9.6.2 Component rationale**

The information erasure policy as defined in FDP\_IRC.1 (Information retention control) serves to protect all information that is contained in the assigned objects from being misused, regardless of whether the information is primary content or any kind of attribute. The policy covers combinations of objects and activities. The policy's coverage may be "complete" with respect to all the objects related to one or more activities, or it may address only some of the objects related to one or more activities.

The term "promptly" in FDP\_IRC.1 (Information retention control) specifically refers to the fact that the objects shall be terminated in a manner that ensures that they cannot be accessed as before.

**11.9.6.3 Element FDP\_IRC.1.1**

**The TSF shall enforce the [assignment (a1): *information erasure policy*] on a [assignment (a2): *list of objects*] required for [assignment (a3): *list of operations*] so that the selected objects are deleted irreversibly and untraceably from the TOE promptly upon termination of the selected operations.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify a uniquely named information erasure policy to be enforced by the TSF.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the list of objects that are required for the respective list of activities, e.g. "all message objects".

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify the list of activities that the information erasure policy is concerned with, e.g. "all activities related to passing a message on, such as receiving a message, cryptographic handling of a message, sending a message".

**11.9.6.4 Element FDP\_IRC.1.2**

**The TSF shall ensure that [assignment (a1): *list of objects*] cannot be accessed after their release and prior to their irreversible and untraceable deletion.**

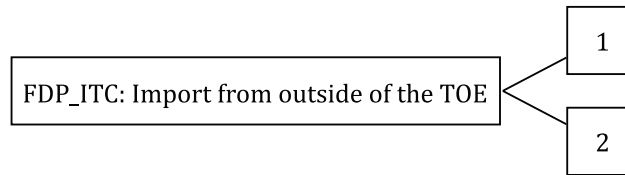
In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of objects that are required for the respective list of activities. This assignment shall be identical to the assigned objects in FDP\_IRC.1.1.

**11.10 Import from outside of the TOE (FDP\_ITC)****11.10.1 Family Behaviour**

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

**11.10.2 Component levelling and description**

[Figure 31](#) shows the component levelling of family FDP\_ITC.

**ISO/IEC 15408-2:2026(en)****Key**

1,2 components

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

FDP\_ITC.1 (Import of user data without security attributes) ([11.10.6](#)) 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) ([11.10.7](#)) requires that security attributes correctly represent the user data and are accurately and unambiguously associated with the user data imported from outside the TOE.

**11.10.3 Component management**

The following action related to FDP\_ITC.1 and FDP\_ITC.2 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the modification of the additional control rules used for import.

**11.10.4 Component audit**

The following actions related to FDP\_ITC.1 and FDP\_ITC.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: successful import of user data, including any security attributes;
- basic: all attempts to import user data, including any security attributes;
- detailed: the specification of security attributes for imported user data supplied by an authorized user.

**11.10.5 Application notes**

This family defines mechanisms for TSF-mediated importing of user data from outside the TOE into the TOE such that the user data security attributes can be preserved. Consistency of these security attributes are addressed by FPT\_TDC (Inter-TSF TSF data consistency) ([15.15](#)).

FDP\_ITC (Import from outside of the TOE) is concerned with limitations on import, user specification of security attributes, and association of security attributes with the user data.

This family, and the corresponding export family FDP\_ETC (Export from the TOE), address how the TOE deals with user data outside its control. This family is concerned with assigning and abstraction of the user data security attributes.

**EXAMPLE 1** A variety of activities can be involved here:

- importing user data from an unformatted medium (e.g. tape, scanner, video or audio signal), without including any security attributes, and physically marking the medium to indicate its contents;
- importing user data, including security attributes, from a medium and verifying that the object security attributes are appropriate;
- importing user data, including security attributes, from a medium using a cryptographic sealing technique to protect the association of user data and security attributes.

**ISO/IEC 15408-2:2026(en)**

This family is not concerned with the determination of whether the user data may be imported. It is concerned with the values of the security attributes to associate with the imported user data.

There are two possibilities for the import of user data: either the user data is unambiguously associated with reliable object security attributes (values and meaning of the security attributes is not modified), or no reliable security attributes (or no security attributes at all) are available from the import source. This family addresses both cases.

If there are reliable security attributes available, they may have been associated with the user data by physical means (the security attributes are on the same media), or by logical means (the security attributes are distributed differently but include unique object identification).

EXAMPLE 2 An example of security attribute is a cryptographic checksum.

This family is concerned with TSF-mediated importing of user data and maintaining the association of security attributes as required by the SFP. Other families are concerned with other import aspects such as consistency, trusted channels, and integrity that are beyond the scope of this family. Furthermore, FDP\_ITC (Import from outside of the TOE) is only concerned with the interface to the import medium. FDP\_ETC (Export from the TOE) is responsible for the other end point of the medium (the source).

Some of the well-known import requirements are:

- importing of user data without any security attributes;
- importing of user data including security attributes where the two are associated with one another and the security attributes unambiguously represent the information being imported.

These import requirements may be handled by the TSF with or without human intervention, depending on the IT limitations and the organizational security policy. For example, if user data is received on a "confidential" channel, the security attributes of the objects will be set to "confidential".

If there are multiple SFPs (access control and/or information flow control) then it may be appropriate to iterate these components once for each named SFP.

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

#### **11.10.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]  
FMT\_MSA.3 (Static attribute initialization) ([13.5.8](#))

#### **11.10.6.2 Component rationale**

This component is used to specify the import of user data that does not have reliable (or any) security attributes associated with it. This function requires that the security attributes for the imported user data be initialized within the TSF. It can also be the case that the author of a PP, PP-Module, functional package or ST specifies the rules for import. It may be appropriate, in some environments, to require that these attributes be supplied via a trusted path or a trusted channel mechanism.

#### **11.10.6.3 Element FDP\_ITC.1.1**

**The TSF shall enforce the [assignment (a1): *access control SFP(s) and/or information flow control SFP(s)*] when importing user data, controlled under the SFP, from outside of the TOE.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) that will be enforced when importing user data from outside of the TOE. The user data that this function imports is scoped by the assignment of these SFPs.

**ISO/IEC 15408-2:2026(en)****11.10.6.4 Element FDP\_ITC.1.2**

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

**11.10.6.5 Element FDP\_ITC.1.3**

**The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: [assignment (a1): *additional importation control rules*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify any additional importation control rules or "none" if there are no additional importation 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\_ITC.1.1.

**11.10.7 FDP\_ITC.2 Import of user data with security attributes****11.10.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]  
[FTP\_ITC.1 (Inter-TSF trusted channel) ([18.3.6](#)), or  
FTP\_TRP.1 (Trusted path) ([18.5.6](#))]  
FPT\_TDC.1 (Inter-TSF basic TSF data consistency) ([15.15.6](#))

**11.10.7.2 Component rationale**

This component is used to specify the import of user data that has reliable security attributes associated with it. This function relies upon the security attributes that are accurately and unambiguously associated with the objects on the import medium. Once imported, those objects shall have those same attributes. This requires FPT\_TDC (Inter-TSF TSF data consistency) ([15.15](#)) to ensure the consistency of the data. It can also be the case that the author of a PP, PP-Module, functional package or ST specifies the rules for import.

**11.10.7.3 Element FDP\_ITC.2.1**

**The TSF shall enforce the [assignment (a1): *access control SFP(s) and/or information flow control SFP(s)*] when importing user data, controlled under the SFP, from outside of the TOE.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) that will be enforced when importing user data from outside of the TOE. The user data that this function imports is scoped by the assignment of these SFPs.

**11.10.7.4 Element FDP\_ITC.2.2**

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

**11.10.7.5 Element FDP\_ITC.2.3**

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

**11.10.7.6 Element FDP\_ITC.2.4**

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

### 11.10.7.7 Element FDP\_ITC.2.5

The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: [assignment <sup>(a1)</sup>: *additional importation control rules*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify any additional importation control rules or "none" if there are no additional importation 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\_ITC.2.1.

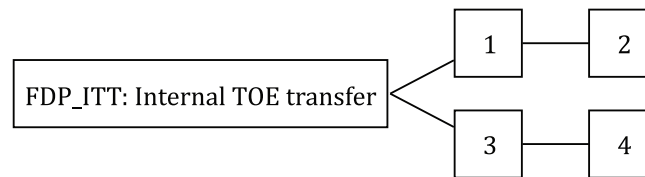
## 11.11 Internal TOE transfer (FDP\_ITT)

### 11.11.1 Family Behaviour

This family provides requirements that address protection of user data when it is transferred between separated parts of a TOE across an internal channel. This may be contrasted with the FDP\_UCT (Inter-TSF user data confidentiality transfer protection) (11.16) and FDP\_UIT (Inter-TSF user data integrity transfer protection) (11.17) families, which provide protection for user data when it is transferred between distinct TSFs across an external channel, and FDP\_ETC (Export from the TOE) and FDP\_ITC (Import from outside of the TOE), which address TSF-mediated transfer of data to or from outside the TOE.

### 11.11.2 Component levelling and description

Figure 32 depicts the component levelling of family FDP\_ITT.



#### Key

1,2,3,4 components

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

FDP\_ITT.1 (Basic internal transfer protection) (11.11.6) requires that user data be protected when transmitted between parts of the TOE.

FDP\_ITT.2 (Transmission separation by attribute) (11.11.7) requires separation of data based on the value of SFP-relevant attributes in addition to the first component.

FDP\_ITT.3 (Integrity monitoring) (11.11.8) requires that the TSF monitor user data transmitted between parts of the TOE for identified integrity errors.

FDP\_ITT.4 (Attribute-based integrity monitoring) (11.11.9) expands on the third component by allowing the form of integrity monitoring to differ by SFP-relevant attributes.

### 11.11.3 Component management

The following action related to FDP\_ITT.1 and FDP\_ITT.2 can be considered for the management functions in class FMT (Security management) (Clause 13):

- if the TSF provides multiple methods to protect user data during transmission between physically separated parts of the TOE, the TSF can provide a pre-defined role with the ability to select the method that will be used.

**ISO/IEC 15408-2:2026(en)**

The following action related to FDP\_ITT.3 and FDP\_ITT.4 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the specification of the actions to be taken upon detection of an integrity error can be configurable.

**11.11.4 Component audit**

The following actions related to FDP\_ITT.1 and FDP\_ITT.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: successful transfers of user data, including identification of the protection method used;
- basic: all attempts to transfer user data, including the protection method used and any errors that occurred.

The following actions related to FDP\_ITT.3 and FDP\_ITT.4 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: successful transfers of user data, including identification of the integrity protection method used;
- basic: all attempts to transfer user data, including the integrity protection method used and any errors that occurred;
- basic: unauthorized attempts to change the integrity protection method;
- detailed: the action taken upon detection of an integrity error.

**11.11.5 Application notes**

This family provides requirements that address protection of user data when it is transferred between parts of a TOE across an internal channel. This may be contrasted with the FDP\_UCT (Inter-TSF user data confidentiality transfer protection) ([11.16](#)) and FDP\_UIT (Inter-TSF user data integrity transfer protection) ([11.17](#)) family, which provide protection for user data when it is transferred between distinct TSFs across an external channel, and FDP\_ETC (Export from the TOE) and FDP\_ITC (Import from outside of the TOE), which address TSF-mediated transfer of data to or from outside the TOE.

The requirements in this family allow a PP, PP-Module, functional package or ST author to specify the desired security for user data while in transit within the TOE. This security can be protection against disclosure, modification, or loss of availability.

The determination of the degree of physical separation above which this family should apply 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. In more benign environments, the transfers may be across more traditional network media.

If there are multiple SFPs (access control and/or information flow control) then it may be appropriate to iterate these components once for each named SFP.

**11.11.6 FDP\_ITT.1 Basic internal transfer protection****11.11.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]

### **11.11.6.2 Element FDP\_ITT.1.1**

**The TSF shall enforce the [assignment <sup>(a1)</sup>: *access control SFP(s) and/or information flow control SFP(s)*] to prevent the [selection <sup>(s1)</sup>: *disclosure, modification, loss of use*] of user data when it is transmitted between physically-separated parts of the TOE.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) covering the information being transferred.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the types of transmission errors that the TSF should prevent occurring for user data while in transport. The options are disclosure, modification, loss of use.

### **11.11.7 FDP\_ITT.2 Transmission separation by attribute**

#### **11.11.7.1 Component relationships**

Hierarchical to: FDP\_ITT.1 (Basic internal transfer protection) ([11.11.6](#))

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]

#### **11.11.7.2 Component rationale**

This component can, for example, be used to provide different forms of protection to information with different clearance levels.

One of the ways to achieve separation of data when it is transmitted is through the use of separate logical or physical channels.

#### **11.11.7.3 Element FDP\_ITT.2.1**

**The TSF shall enforce the [assignment <sup>(a1)</sup>: *access control SFP(s) and/or information flow control SFP(s)*] to prevent the [selection <sup>(s1)</sup>: *disclosure, modification, loss of use*] of user data when it is transmitted between physically-separated parts of the TOE.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) covering the information being transferred.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the types of transmission errors that the TSF should prevent occurring for user data while in transport. The options are disclosure, modification, loss of use.

#### **11.11.7.4 Element FDP\_ITT.2.2**

**The TSF shall separate data controlled by the SFP(s) when transmitted between physically-separated parts of the TOE, based on the values of the following: [assignment <sup>(a1)</sup>: *security attributes that require separation*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the security attributes, the values of which the TSF will use to determine when to separate data that is being transmitted between physically-separated parts of the TOE. An example is that user data associated with the identity of one owner is transmitted separately from the user data associated with the identify of a different owner. In this case, the value of the identity of the owner of the data is what is used to determine when to separate the data for transmission.

**ISO/IEC 15408-2:2026(en)****11.11.8 FDP\_ITT.3 Integrity monitoring****11.11.8.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]  
FDP\_ITT.1 (Basic internal transfer protection) ([11.11.6](#))

**11.11.8.2 Component rationale**

This component is used in combination with either FDP\_ITT.1 (Basic internal transfer protection) ([11.11.6](#)) or FDP\_ITT.2 (Transmission separation by attribute) ([11.11.7](#)). It ensures that the TSF checks received user data (and their attributes) for integrity. FDP\_ITT.1 (Basic internal transfer protection) ([11.11.6](#)) or FDP\_ITT.2 (Transmission separation by attribute) ([11.11.7](#)) will provide the data in a manner such that it is protected from modification (so that FDP\_ITT.3 (Integrity monitoring) can detect any modifications).

The author of a PP, PP-Module, functional package or ST shall specify the types of errors that must be detected. The author of a PP, PP-Module, functional package or ST should consider: modification of data, substitution of data, unrecoverable ordering change of data, replay of data, incomplete data, in addition to other integrity errors.

The author of a PP, PP-Module, functional package or ST specifies the actions that the TSF should take on detection of a failure.

**EXAMPLE** Ignoring the user data, requesting the data again, informing the authorized administrator, rerouting traffic for other lines.

**11.11.8.3 Element FDP\_ITT.3.1**

**The TSF shall enforce the [assignment <sup>(a1)</sup>: *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 <sup>(a2)</sup>: *integrity errors*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) covering the information being transferred and monitored for integrity errors.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the type of possible integrity errors to be monitored during transmission of the user data.

**11.11.8.4 Element FDP\_ITT.3.2**

**Upon detection of a data integrity error, the TSF shall [assignment <sup>(a1)</sup>: *specify the action to be taken upon integrity error*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the action to be taken by the TSF when an integrity error is encountered.

**11.11.9 FDP\_ITT.4 Attribute-based integrity monitoring****11.11.9.1 Component relationships**

Hierarchical to: FDP\_ITT.3 (Integrity monitoring) ([11.11.8](#))

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]  
FDP\_ITT.2 (Transmission separation by attribute) ([11.11.7](#))

**ISO/IEC 15408-2:2026(en)****11.11.9.2 Component rationale**

This component is used in combination with FDP\_ITT.2 (Transmission separation by attribute) ([11.11.7](#)). It ensures that the TSF checks received user data, that has been transmitted by separate channels (based on values of specified security attributes), for integrity. It allows the author of a PP, PP-Module, functional package or ST to specify actions to be taken upon detection of an integrity error.

**EXAMPLE 1** This component can be used to provide different integrity error detection and action for information at different integrity levels.

The author of a PP, PP-Module, functional package or ST shall specify the types of errors that must be detected. The author of a PP, PP-Module, functional package or ST should consider: modification of data, substitution of data, unrecoverable ordering change of data, replay of data, incomplete data, in addition to other integrity errors.

The author of a PP, PP-Module, functional package or ST should specify the attributes (and associated transmission channels) that necessitate integrity error monitoring.

The author of a PP, PP-Module, functional package or ST specifies the actions that the TSF should take on detection of a failure.

**EXAMPLE 2** Ignore the user data, request the data again, inform the authorized administrator, reroute traffic for other lines.

**11.11.9.3 Element FDP\_ITT.4.1**

The TSF shall enforce the [assignment <sup>(a1)</sup>: *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 <sup>(a2)</sup>: *integrity errors*], **based on the following attributes: [assignment <sup>(a3)</sup>: *security attributes that require separate transmission channels*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) covering the information being transferred and monitored for integrity errors.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the type of possible integrity errors to be monitored during transmission of the user data.

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify a list of security attributes that require separate transmission channels. This list is used to determine which user data to monitor for integrity errors., based on its security attributes and its transmission channel. This element is directly related to FDP\_ITT.2 (Transmission separation by attribute) ([11.11.7](#)).

**11.11.9.4 Element FDP\_ITT.4.2**

Upon detection of a data integrity error, the TSF shall [assignment <sup>(a1)</sup>: *specify the action to be taken upon integrity error*].

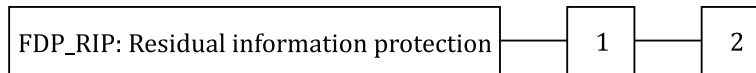
In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the action to be taken by the TSF when an integrity error is encountered. An example can be that the TSF should request the resubmission of the user data. The SFP(s) specified in FDP\_ITT.4.1 will be enforced as the actions are taken by the TSF.

**11.12 Residual information protection (FDP\_RIP)****11.12.1 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.

### 11.12.2 Component levelling and description

[Figure 33](#) illustrates the component levelling of family FDP\_RIP.



#### Key

1,2 components

**Figure 33 — FDP\_RIP: Component levelling**

FDP\_RIP.1 (Subset residual information protection) ([11.12.6](#)) 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) ([11.12.7](#)) 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.

### 11.12.3 Component management

The following action related to FDP\_RIP.1 and FDP\_RIP.2 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

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

### 11.12.4 Component audit

There are no auditable events foreseen related to FDP\_RIP.1 or FDP\_RIP.2.

### 11.12.5 Application notes

Residual information protection ensures that TSF-controlled resources when de-allocated from an object and before they are reallocated to another object are treated by the TSF in a way that it is not possible to reconstruct all or part of the data contained in the resource before it was de-allocated.

A TOE usually has a number of functions that potentially de-allocate resources from an object and potentially re-allocate those resources to objects. Some, but not all of those resources may have been used to store critical data from the previous use of the resource and for those resources FDP\_RIP (Residual information protection) requires that they are prepared for reuse. Object reuse applies to explicit requests of a subject or user to release resources as well as implicit actions of the TSF that 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.

The requirement for object reuse is related to the content of the resource belonging to an object, not all information about the resource or object that may be stored elsewhere in the TSF. As an example, to satisfy the FDP\_RIP (Residual information protection) requirement for files as objects requires that all sectors that make up the file be prepared for re-use.

It also applies to resources that are serially reused by different subjects within the system. For example, most operating systems typically rely upon hardware registers (resources) to support processes within the system. As processes are swapped from a "run" state to a "sleep" state (and vice versa), these registers are serially reused by different subjects. While this "swapping" action is not required to be considered an allocation or deallocation of a resource, FDP\_RIP (Residual information protection) can apply to such events and resources.

## ISO/IEC 15408-2:2026(en)

FDP\_RIP (Residual information protection) typically controls access to information that is not part of any currently defined or accessible object; however, in certain cases this is not necessarily true. For example, object "A" is a file and object "B" is the disk upon which that file resides. If object "A" is deleted, the information from object "A" is under the control of FDP\_RIP (Residual information protection) even though it is still part of object "B".

It is important to note that FDP\_RIP (Residual information protection) applies only to on-line objects and not off-line objects such as those backed-up on tapes. For example, if a file is deleted in the TOE, FDP\_RIP (Residual information protection) can be instantiated to require that no residual information exists upon deallocation; however, the TSF cannot extend this enforcement to that same file that exists on the off-line back-up. Therefore, that same file is still available. If this is a concern, then the author of a PP, PP-Module, functional package or ST should make sure that the proper environmental objectives are in place to support operational user guidance to address off-line objects.

FDP\_RIP (Residual information protection) and FDP\_ROL (Rollback) (11.13) can conflict when FDP\_RIP (Residual information protection) is instantiated to require that residual information be cleared at the time the application releases the object to the TSF (i.e. upon deallocation). Therefore, the FDP\_RIP (Residual information protection) selection of "deallocation" should not be used with FDP\_ROL (Rollback) (11.13) since there would be no information to roll back. The other selection, "unavailability upon allocation", may be used with FDP\_ROL (Rollback) (11.13), but there is the risk that the resource which held the information has been allocated to a new object before the roll back took place. If that were to occur, then the roll back would not be possible.

There are no audit requirements in FDP\_RIP (Residual information protection) because this is not a user-invokable function. Auditing of allocated or deallocated resources would be auditable as part of the access control SFP or the information flow control SFP operations.

This family should apply to the objects specified in the access control SFP(s) or the information flow control SFP(s) as specified by the author of a PP, PP-Module, functional package or ST.

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

#### 11.12.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 11.12.6.2 Component rationale

This component requires that, for a subset of the objects in the TOE, the TSF will ensure that there is no available residual information contained in a resource allocated to those objects or deallocated from those objects.

#### 11.12.6.3 Element FDP\_RIP.1.1

**The TSF shall ensure that any previous information content of a resource is made unavailable upon the [selection (s1): *allocation of the resource to, deallocation of the resource from*] the following objects: [assignment (a1): *list of objects*].**

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the event, allocation of the resource to or deallocation of the resource from, that invokes the residual information protection function.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of objects subject to residual information protection.

## ISO/IEC 15408-2:2026(en)

### 11.12.7 FDP\_RIP.2 Full residual information protection

#### 11.12.7.1 Component relationships

Hierarchical to: FDP\_RIP.1 (Subset residual information protection) ([11.12.6](#))

Dependencies: No dependencies.

#### 11.12.7.2 Component rationale

This component requires that for all objects in the TOE, the TSF will ensure that there is no available residual information contained in a resource allocated to those objects or deallocated from those objects.

#### 11.12.7.3 Element FDP\_RIP.2.1

The TSF shall ensure that any previous information content of a resource is made unavailable upon the [selection <sup>(s1)</sup>: *allocation of the resource to, deallocation of the resource from*] **all** objects.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the event, allocation of the resource to or deallocation of the resource from, that invokes the residual information protection function.

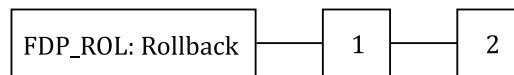
### 11.13 Rollback (FDP\_ROL)

#### 11.13.1 Family Behaviour

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

#### 11.13.2 Component levelling and description

[Figure 34](#) illustrates the component levelling of family FDP\_ROL.



#### Key

1, 2 components

**Figure 34 — FDP\_ROL: Component levelling**

FDP\_ROL.1 (Basic rollback) ([11.13.6](#)) addresses a need to roll back or undo a limited number of operations within the defined bounds.

FDP\_ROL.2 (Advanced rollback) ([11.13.7](#)) addresses the need to roll back or undo all operations within the defined bounds.

#### 11.13.3 Component management

The following actions related to FDP\_ROL.1 and FDP\_ROL.2 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the boundary limit to which rollback may be performed can be a configurable item within the TOE;
- permission to perform a rollback operation can be restricted to a well-defined role.

## ISO/IEC 15408-2:2026(en)

### 11.13.4 Component audit

The following actions related to FDP\_ROL.1 and FDP\_ROL.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: all successful rollback operations;
- basic: all attempts to perform rollback operations;
- detailed: all attempts to perform rollback operations, including identification of the types of operations rolled back.

### 11.13.5 Application notes

This family addresses the need to return to a well-defined valid state, such as the need of a user to undo modifications to a file or to undo transactions in case of an incomplete series of transaction as in the case of databases.

This family is intended to assist a user in returning to a well-defined valid state after the user undoes the last set of actions, or, in distributed databases, the return of all of the distributed copies of the databases to the state before an operation failed.

FDP\_RIP (Residual information protection) and FDP\_ROL (Rollback) conflict when FDP\_RIP (Residual information protection) enforces that the contents will be made unavailable at the time that a resource is deallocated from an object. Therefore, this use of FDP\_RIP (Residual information protection) cannot be combined with FDP\_ROL (Rollback) as there would be no information to roll back. FDP\_RIP (Residual information protection) can be used only with FDP\_ROL (Rollback) when it enforces that the contents will be unavailable at the time that a resource is allocated to an object. This is because the FDP\_ROL (Rollback) mechanism will have an opportunity to access the previous information that can still be present in the TOE in order to successfully roll back the operation.

The rollback requirement is bounded by certain limits.

EXAMPLE 1 A text editor typically only allows you roll back up to a certain number of commands.

EXAMPLE 2 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.

### 11.13.6 FDP\_ROL.1 Basic rollback

#### 11.13.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) (11.3.6), or  
FDP\_IFC.1 (Subset information flow control) (11.7.6)]

#### 11.13.6.2 Component rationale

This component allows a user or subject to undo a set of operations on a predefined set of objects. The undo is only possible within certain limits, for example up to a number of characters or up to a time limit.

#### 11.13.6.3 Element FDP\_ROL.1.1

**The TSF shall enforce [assignment <sup>(a1)</sup>: *access control SFP(s) and/or information flow control SFP(s)*] to permit the rollback of the [assignment <sup>(a2)</sup>: *list of operations*] on the [assignment <sup>(a3)</sup>: *information and/or list of objects*].**

**ISO/IEC 15408-2:2026(en)**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) that will be enforced when performing rollback operations. This is necessary to make sure that roll back is not used to circumvent the specified SFPs.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the list of operations that can be rolled back.

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify the information and/or list of objects that are subjected to the rollback policy.

**11.13.6.4 Element FDP\_ROL.1.2**

**The TSF shall permit operations to be rolled back within the [assignment <sup>(a1)</sup>: *boundary limit to which rollback may be performed*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the boundary limit to which rollback operations may be performed. The boundary may be specified as a predefined period of time,

**11.13.7 FDP\_ROL.2 Advanced rollback****11.13.7.1 Component relationships**

Hierarchical to: FDP\_ROL.1 (Basic rollback) ([11.13.6](#))

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]

**11.13.7.2 Component rationale**

This component enforces that the TSF provide the capability to rollback all operations; however, the user can choose to rollback only a part of them.

**11.13.7.3 Element FDP\_ROL.2.1**

The TSF shall enforce [assignment <sup>(a1)</sup>: *access control SFP(s) and/or information flow control SFP(s)*] to permit the rollback of **all the operations on the** [assignment <sup>(a2)</sup>: *list of objects*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) that will be enforced when performing rollback operations. This is necessary to make sure that roll back is not used to circumvent the specified SFPs.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the list of objects that are subjected to the rollback policy.

**11.13.7.4 Element FDP\_ROL.2.2**

The TSF shall permit operations to be rolled back within the [assignment <sup>(a1)</sup>: *boundary limit to which rollback may be performed*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the boundary limit to which rollback operations may be performed. The boundary may be specified as a predefined period of time,

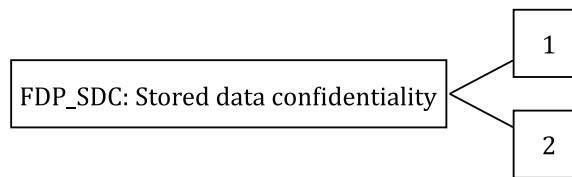
## 11.14 Stored data confidentiality (FDP\_SDC)

### 11.14.1 Family Behaviour

This family provides requirements that address protection of user data confidentiality while the data is stored within memory areas protected by the TSF. The TSF provides access to the data in the memory through the specified interfaces only and prevents compromise of their information bypassing these interfaces. It complements the family FDP\_SDI (Stored data integrity) (11.15) which protects the user data from integrity errors while being stored in the memory.

### 11.14.2 Component levelling and description

Figure 35 illustrates the component levelling of family FDP\_SDC.



#### Key

1,2 components

**Figure 35 — FDP\_SDC: Component levelling**

FDP\_SDC.1 (Stored data confidentiality) (11.14.7) requires the TSF to protect the confidentiality of information of the user data in specified memory areas.

FDP\_SDC.2 (Stored data confidentiality with dedicated method) (11.14.8) requires the TSF to protect the confidentiality of the user data according to data characteristics leading to specify a dedicated method of protection of confidentiality.

### 11.14.3 Component management

There are no management activities foreseen related to FDP\_SDC.1 or FDP\_SDC.2.

### 11.14.4 Component audit

There are no auditable events foreseen related to FDP\_SDC.1 or FDP\_SDC.2.

### 11.14.5 Application notes

This family provides requirements that address protection of user data confidentiality while the data is stored within memory areas protected by the TSF. The TSF provides access to the data in the memory through the specified interfaces only and prevents compromise of their information bypassing these interfaces. It complements the family FDP\_SDI (Stored data integrity) (11.15) which protects the user data from integrity errors while being stored in the memory.

### 11.14.6 Evaluator notes

In practice, the dependency to FCS\_COP.1 (Cryptographic operation) (10.4.6) may be satisfied by a PP, PP-Module, functional package or ST author by providing a rationale explaining an alternative method to cryptography is used in some dedicated cases.

**ISO/IEC 15408-2:2026(en)****11.14.7 FDP\_SDC.1 Stored data confidentiality****11.14.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**11.14.7.2 Component rationale**

In FDP\_SDC.1 (Stored data confidentiality), the author of a PP, PP-Module, functional package or ST specifies which user data is intended to be protected and in which type of memory the user data is requested to be protected. In the second selection the author of a PP, PP-Module, functional package or ST provides the memory type where the user data is intended to be protected.

**11.14.7.3 Element FDP\_SDC.1.1**

**The TSF shall ensure the confidentiality of [selection <sup>(s1)</sup>: *all user data, the following user data [assignment <sup>(a1)</sup>: list of user data]*] while it is stored in the [selection <sup>(s2)</sup>: *temporary memory, persistent memory, any memory*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST provides a list of the user data that is intended to be protected in memory.

In selection (s1), the author of a PP, PP-Module, functional package or ST shall select either "all user data" or provide a list of user data using the assignment below.

In selection (s2), the author of a PP, PP-Module, functional package or ST can specify either temporary memory, persistent memory or any memory. "Any memory" includes both temporary (volatile) and persistent (non-volatile) memory.

**11.14.8 FDP\_SDC.2 Stored data confidentiality with dedicated method****11.14.8.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FCS\_COP.1 (Cryptographic operation) ([10.4.6](#))

**11.14.8.2 Component rationale**

FDP\_SDC.2 (Stored data confidentiality with dedicated method) refines the FDP\_SDC.1.1 element by allowing the author of a PP, PP-Module, functional package or ST to refine the list of user data using a variety of data characteristics.

**11.14.8.3 Element FDP\_SDC.2.1**

**The TSF shall ensure the confidentiality of the [selection <sup>(s1)</sup>: *all user data, the following user data [assignment <sup>(a1)</sup>: list of user data]*] according to [assignment <sup>(a2)</sup>: *data characteristics*] while it is stored under the control of the TSF.**

In selection (s1), (s2), and (a1), the operations of selection and the first assignment are the same as that in FDP\_SDC.1 (Stored data confidentiality) ([11.14.7](#))

In assignment (a2), the author of a PP, PP-Module, functional package or ST provides the data characteristics. Data characteristics can include items such as data length (shorter or longer than a threshold), data type (binary, text, image, sound, video), and data representation (binary, vector, character, frame).

#### 11.14.8.4 Element FDP\_SDC.2.2

The TSF shall ensure the confidentiality of the user data specified in FDP\_SDC.2.1 without user intervention.

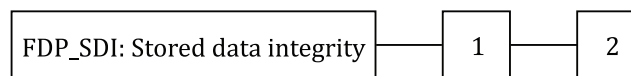
### 11.15 Stored data integrity (FDP\_SDI)

#### 11.15.1 Family Behaviour

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

#### 11.15.2 Component levelling and description

Figure 36 illustrates the component levelling of family FDP\_SDI.



#### Key

1,2 components

**Figure 36 — FDP\_SDI: Component levelling**

FDP\_SDI.1 (Stored data integrity monitoring) (11.15.6) requires that the TSF monitor user data stored within containers controlled by the TSF for identified integrity errors.

FDP\_SDI.2 (Stored data integrity monitoring and action) (11.15.7) adds the additional capability to the first component by allowing for actions to be taken as a result of an error detection.

#### 11.15.3 Component management

The following action related to FDP\_SDI.2 can be considered for the management functions in class FMT (Security management) (Clause 13):

- the actions to be taken upon the detection of an integrity error can be configurable.

There are no management activities foreseen related to FDP\_SDI.1.

#### 11.15.4 Component audit

The following actions related to FDP\_SDI.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: successful attempts to check the integrity of user data, including an indication of the results of the check;
- basic: all attempts to check the integrity of user data, including an indication of the results of the check, if performed;
- detailed: the type of integrity error that occurred.

The following actions related to FDP\_SDI.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: successful attempts to check the integrity of user data, including an indication of the results of the check;

**ISO/IEC 15408-2:2026(en)**

- basic: all attempts to check the integrity of user data, including an indication of the results of the check, if performed;
- detailed: the type of integrity error that occurred;
- detailed: the action taken upon detection of an integrity error.

**11.15.5 Application notes**

This family provides requirements that address protection of user data while it is stored within containers controlled by the TSF.

Hardware glitches or errors may affect data stored in memory. This family provides requirements to detect these unintentional errors. The integrity of user data while stored on storage devices controlled by the TSF are also addressed by this family.

To prevent a subject from modifying the data, the FDP\_IFF (Information flow control functions) or FDP\_ACF (Access control functions) families are required (rather than this family).

This family differs from FDP\_ITT (Internal TOE transfer) that protects the user data from integrity errors while being transferred within the TOE.

**11.15.6 FDP\_SDI.1 Stored data integrity monitoring****11.15.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**11.15.6.2 Component rationale**

This component monitors data stored on media for integrity errors. The author of a PP, PP-Module, functional package or ST can specify different kinds of user data attributes that will be used as the basis for monitoring.

**11.15.6.3 Element FDP\_SDI.1.1**

**The TSF shall monitor user data stored in containers controlled by the TSF for [assignment (a1): *integrity errors*] on all objects, based on the following attributes: [assignment (a2): *user data attributes*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the integrity errors that the TSF will detect.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the user data attributes that will be used as the basis for the monitoring.

**11.15.7 FDP\_SDI.2 Stored data integrity monitoring and action****11.15.7.1 Component relationships**

Hierarchical to: FDP\_SDI.1 (Stored data integrity monitoring) ([11.15.6](#))

Dependencies: No dependencies.

**11.15.7.2 Component rationale**

This component monitors data stored on media for integrity errors. The author of a PP, PP-Module, functional package or ST can specify which action should be taken in case an integrity error is detected.

## ISO/IEC 15408-2:2026(en)

### 11.15.7.3 Element FDP\_SDI.2.1

The TSF shall monitor user data stored in containers controlled by the TSF for [assignment <sup>(a1)</sup>: *integrity errors*] on all objects, based on the following attributes: [assignment <sup>(a2)</sup>: *user data attributes*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the integrity errors that the TSF will detect.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the user data attributes that will be used as the basis for the monitoring.

### 11.15.7.4 Element FDP\_SDI.2.2

**Upon detection of a data integrity error, the TSF shall [assignment <sup>(a1)</sup>: *action to be taken*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the actions to be taken in case an integrity error is detected.

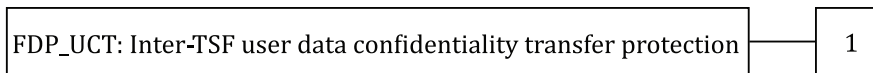
## 11.16 Inter-TSF user data confidentiality transfer protection (FDP\_UCT)

### 11.16.1 Family Behaviour

This family defines the requirements for ensuring the confidentiality of user data when it is transferred using an external channel between the TOE and another trusted IT product.

### 11.16.2 Component levelling and description

[Figure 37](#) depicts the component levelling of family FDP\_UCT.



#### Key

1 component

**Figure 37 — FDP\_UCT: Component levelling**

In FDP\_UCT.1 (Basic data exchange confidentiality) ([11.16.6](#)), the goal is to provide protection from unauthorized disclosure of user data while in transit.

### 11.16.3 Component management

There are no management activities foreseen related to FDP\_UCT.1.

### 11.16.4 Component audit

The following actions related to FDP\_UCT.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the identity of any user or subject using the data exchange mechanisms;
- basic: the identity of any unauthorized user or subject attempting to use the data exchange mechanisms;
- basic: a reference to the names or other indexing information useful in identifying the user data that was transmitted or received. this can include security attributes associated with the information.

**ISO/IEC 15408-2:2026(en)****11.16.5 Application notes**

This family defines the requirements for ensuring the confidentiality of user data when it is transferred using an external channel between the TOE and another trusted IT product. Confidentiality is enforced by preventing unauthorized disclosure of user data in transit between the two end points. The end points may be a TSF or a user.

This family provides a requirement for the protection of user data during transit. In contrast, FPT\_ITC (Confidentiality of exported TSF data) ([15.7](#)) handles TSF data.

**11.16.6 FDP\_UCT.1 Basic data exchange confidentiality****11.16.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FTP\_ITC.1 (Inter-TSF trusted channel) ([18.3.6](#)), or  
FTP\_TRP.1 (Trusted path) ([18.5.6](#))  
[FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]

**11.16.6.2 Component rationale**

Depending on the access control or information flow policies the TSF is required to send or receive user data in a manner such that the confidentiality of the user data is protected.

**11.16.6.3 Element FDP\_UCT.1.1**

**The TSF shall enforce the [assignment <sup>(a1)</sup>: *access control SFP(s) and/or information flow control SFP(s)*] to [selection <sup>(s1)</sup>: *transmit, receive*] user data in a manner protected from unauthorized disclosure.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) that will be enforced when exchanging user data. The specified policies will be enforced to make decisions about who can exchange data and which data can be exchanged.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify whether this element applies to a mechanism that transmits or receives user data.

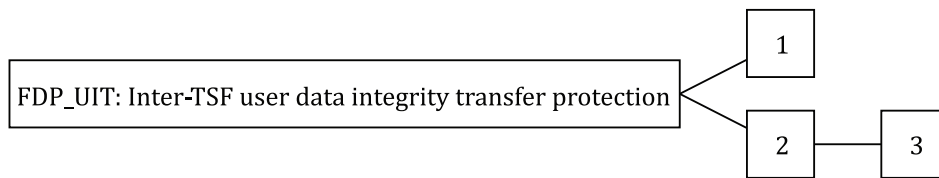
**11.17 Inter-TSF user data integrity transfer protection (FDP\_UIT)****11.17.1 Family Behaviour**

This family defines the requirements for providing integrity for user data in transit between the TOE and another trusted IT product and recovering from detectable errors. At a minimum, this family monitors the integrity of user data for modifications. Furthermore, this family supports different ways of correcting detected integrity errors.

**11.17.2 Component levelling and description**

[Figure 38](#) illustrates the component levelling of family FDP\_UIT.

## ISO/IEC 15408-2:2026(en)



### Key

1,2,3 components

**Figure 38 — FDP\_UIT: Component levelling**

FDP\_UIT.1 (Data exchange integrity) (11.17.6) addresses detection of modifications, deletions, insertions, and replay errors of the user data transmitted.

FDP\_UIT.2 (Source data exchange recovery) (11.17.7) addresses recovery of the original user data by the receiving TSF with help from the source trusted IT product.

FDP\_UIT.3 (Destination data exchange recovery) (11.17.8) addresses recovery of the original user data by the receiving TSF on its own without any help from the source trusted IT product.

### 11.17.3 Component management

There are no management activities foreseen related to FDP\_UIT.1, FDP\_UIT.2 or FDP\_UIT.3.

### 11.17.4 Component audit

The following actions related to FDP\_UIT.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the identity of any user or subject using the data exchange mechanisms;
- basic: the identity of any user or subject attempting to use the user data exchange mechanisms, but who is unauthorized to do so;
- basic: a reference to the names or other indexing information useful in identifying the user data that was transmitted or received. this can include security attributes associated with the user data;
- basic: any identified attempts to block transmission of user data;
- detailed: the types and/or effects of any detected modifications of transmitted user data.

The following actions related to FDP\_UIT.2 and FDP\_UIT.3 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the identity of any user or subject using the data exchange mechanisms;
- minimal: successful recovery from errors including the type of error that was detected;
- basic: the identity of any user or subject attempting to use the user data exchange mechanisms, but who is unauthorized to do so;
- basic: a reference to the names or other indexing information useful in identifying the user data that was transmitted or received. this can include security attributes associated with the user data;
- basic: any identified attempts to block transmission of user data;
- detailed: the types and/or effects of any detected modifications of transmitted user data.

**ISO/IEC 15408-2:2026(en)****11.17.5 Application notes**

This family defines the requirements for providing integrity for user data in transit between the TSF and another trusted IT product and recovering from detectable errors. At a minimum, this family monitors the integrity of user data for modifications. Furthermore, this family supports different ways of correcting detected integrity errors.

This family defines the requirements for providing integrity for user data in transit; while FPT\_ITI (Integrity of exported TSF data) (15.8) handles TSF data.

FDP\_UIT (Inter-TSF user data integrity transfer protection) and FDP\_UCT (Inter-TSF user data confidentiality transfer protection) are duals of each other, as FDP\_UCT (Inter-TSF user data confidentiality transfer protection) addresses user data confidentiality. Therefore, the same mechanism that implements FDP\_UIT (Inter-TSF user data integrity transfer protection) can possibly be used to implement other families such as FDP\_UCT (Inter-TSF user data confidentiality transfer protection) and FDP\_ITC (Import from outside of the TOE).

**11.17.6 FDP\_UIT.1 Data exchange integrity****11.17.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) (11.3.6), or  
FDP\_IFC.1 (Subset information flow control) (11.7.6)]  
[FTP\_ITC.1 (Inter-TSF trusted channel) (18.3.6), or  
FTP\_TRP.1 (Trusted path) (18.5.6)]

**11.17.6.2 Component rationale**

Depending on the access control or information flow policies the TSF is required to send or receive user data in a manner such that modification of the user data is detected. There is no requirement for a TSF mechanism to attempt to recover from the modification.

**11.17.6.3 Element FDP\_UIT.1.1**

**The TSF shall enforce the [assignment (a<sup>1</sup>): *access control SFP(s) and/or information flow control SFP(s)*] to [selection (s<sup>1</sup>): *transmit, receive*] user data in a manner protected from [selection (s<sup>2</sup>): *modification, deletion, insertion, replay*] errors.**

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) that will be enforced on the transmitted data or on the received data. The specified policies will be enforced to make decisions about who can transmit or who can receive data, and which data can be transmitted or received.

In selection (s<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should specify whether this element applies to a TSF that is transmitting or receiving objects.

In selection (s<sup>2</sup>), the author of a PP, PP-Module, functional package or ST should specify whether the data should be protected from modification, deletion, insertion, or replay.

**11.17.6.4 Element FDP\_UIT.1.2**

**The TSF shall be able to determine on receipt of user data, whether [selection (s<sup>1</sup>): *modification, deletion, insertion, replay*] has occurred.**

In selection (s<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should specify whether the errors of the type: modification, deletion, insertion, or replay are detected.

## ISO/IEC 15408-2:2026(en)

### 11.17.7 FDP\_UIT.2 Source data exchange recovery

#### 11.17.7.1 Component relationships

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]  
[FDP\_UIT.1 (Data exchange integrity) ([11.17.6](#)), or  
FTP\_ITC.1 (Inter-TSF trusted channel) ([18.3.6](#))]

#### 11.17.7.2 Component rationale

This component provides the ability to recover from a set of identified transmission errors, if required, with the help of the other trusted IT product. As the other trusted IT product is outside the TOE, the TSF cannot control its behaviour. However, it can provide functions that have the ability to cooperate with the other trusted IT product for the purposes of recovery.

**EXAMPLE** The TSF can include functions that depend upon the source trusted IT product to re-send the data in the event that an error is detected.

This component deals with the ability of the TSF to handle such an error recovery.

#### 11.17.7.3 Element FDP\_UIT.2.1

**The TSF shall enforce the [assignment <sup>(a1)</sup>: *access control SFP(s) and/or information flow control SFP(s)*] to be able to recover from [assignment <sup>(a2)</sup>: *list of recoverable errors*] with the help of the source trusted IT product.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) that will be enforced when recovering user data. The specified policies will be enforced to make decisions about which data can be recovered and how it can be recovered.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the list of integrity errors from which the TSF, with the help of the source trusted IT product, is able to recover the original user data.

### 11.17.8 FDP\_UIT.3 Destination data exchange recovery

#### 11.17.8.1 Component relationships

Hierarchical to: FDP\_UIT.2 (Source data exchange recovery) ([11.17.7](#))

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]  
[FDP\_UIT.1 (Data exchange integrity) ([11.17.6](#)), or  
FTP\_ITC.1 (Inter-TSF trusted channel) ([18.3.6](#))]

#### 11.17.8.2 Component rationale

This component provides the ability to recover from a set of identified transmission errors. It accomplishes this task without help from the source trusted IT product.

**EXAMPLE** If certain errors are detected, the transmission protocol must be robust enough to allow the TSF to recover from the error based on checksums and other information available within that protocol.

### **11.17.8.3 Element FDP\_UIT.3.1**

The TSF shall enforce the [assignment <sup>(a1)</sup>: *access control SFP(s) and/or information flow control SFP(s)*] to be able to recover from [assignment <sup>(a2)</sup>: *list of recoverable errors*] **without any help from** the source trusted IT product.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the access control SFP(s) and/or information flow control SFP(s) that will be enforced when recovering user data. The specified policies will be enforced to make decisions about which data can be recovered and how it can be recovered.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the list of integrity errors from which the receiving TSF, alone, is able to recover the original user data.

## **12 Class FIA Identification and authentication**

### **12.1 Introduction**

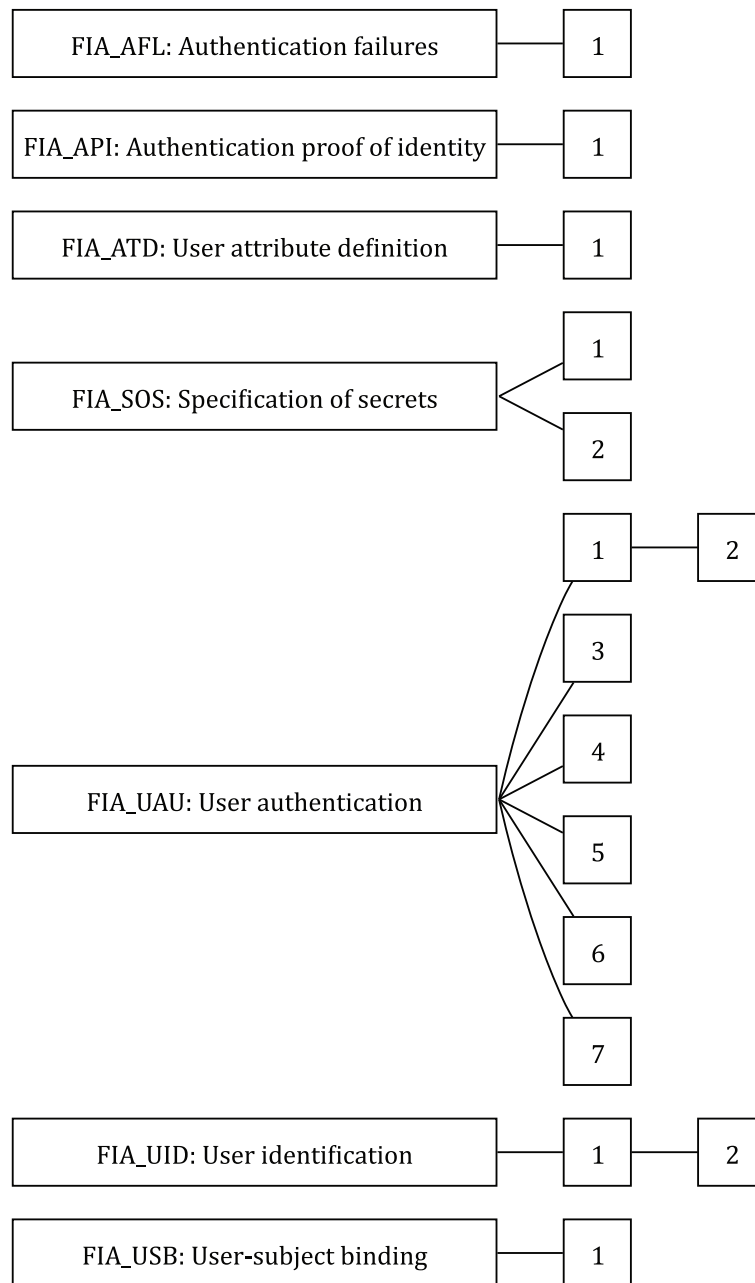
Families in this class address the requirements for functions to establish and verify a claimed user identity.

Identification and authentication are 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](#) illustrates the family hierarchy of class FIA.

**ISO/IEC 15408-2:2026(en)**



**Key**

1,2,3,4,5,6,7 components

**Figure 39 — FIA: Identification and authentication, class decomposition**

Direct, indirect and optional dependencies for the components of class FIA are shown in [Table 5](#).

**ISO/IEC 15408-2:2026(en)**

**Table 5 — Dependency table for class FIA: Identification and authentication**

	FIA_ATD.1 (12.5.6)	FIA_UAU.1 (12.7.6)	FIA_UID.1 (12.8.6)
FIA_AFL.1 (12.3.6)		X	-
FIA_API.1 (12.4.6)			
FIA_ATD.1 (12.5.6)			
FIA_SOS.1 (12.6.6)			
FIA_SOS.2 (12.6.7)			
FIA_UAU.1 (12.7.6)			X
FIA_UAU.2 (12.7.7)			X
FIA_UAU.3 (12.7.8)			
FIA_UAU.4 (12.7.9)			
FIA_UAU.5 (12.7.10)			
FIA_UAU.6 (12.7.11)			
FIA_UAU.7 (12.7.12)		X	-
FIA_UID.1 (12.8.6)			
FIA_UID.2 (12.8.7)			
FIA_USB.1 (12.9.6)	X		

**12.2 Notes on class FIA**

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 FIA\_AFL (Authentication failures) (12.3) family addresses defining limits on repeated unsuccessful authentication attempts.

The FIA\_API (Authentication proof of identity) (12.4) family addresses defining the functionality provided by the TOE to prove its identity and to be verified by an external entity in the TOE IT environment.

The FIA\_ATD (User attribute definition) (12.5) family addresses the definition of user attributes that are used in the enforcement of the SFRs.

The FIA\_SOS (Specification of secrets) (12.6) family addresses the generation and verification of secrets that satisfy a defined metric.

The FIA\_UAU (User authentication) (12.7) family addresses verifying the identity of a user.

The FIA\_UID (User identification) (12.8) family addresses determining the identity of a user.

The FIA\_USB (User-subject binding) (12.9) family addresses the correct association of security attributes for each authorized user.

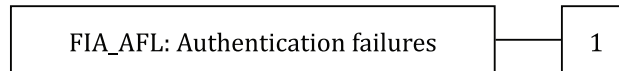
## 12.3 Authentication failures (FIA\_AFL)

### 12.3.1 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.

### 12.3.2 Component levelling and description

[Figure 40](#) illustrates the component levelling of family FIA\_AFL.



#### Key

1 component

**Figure 40 — FIA\_AFL: Component levelling**

FIA\_AFL.1 (Authentication failure handling) ([12.3.6](#)) 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.

### 12.3.3 Component management

The following actions related to FIA\_AFL.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- management of the threshold for unsuccessful authentication attempts;
- management of actions to be taken in the event of an authentication failure.

### 12.3.4 Component audit

The following action related to FIA\_AFL.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- 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.

### 12.3.5 Application 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.

## 12.3.6 FIA\_AFL.1 Authentication failure handling

### 12.3.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FIA\_UAU.1 (Timing of authentication) ([12.7.6](#))

### 12.3.6.2 Component rationale

The author of a PP, PP-Module, functional package or ST may define the number of unsuccessful authentication attempts or may choose to let the TOE developer or the authorized user to define this number. The unsuccessful authentication attempts are not required to be consecutive, but rather related to an authentication event. Such an authentication event can be the count from the last successful session establishment at a given terminal.

The author of a PP, PP-Module, functional package or ST can specify a list of actions that the TSF shall take in the case of authentication failure. An authorized administrator can also be allowed to manage the events, if deemed opportune by the author of a PP, PP-Module, functional package or ST. These actions can be, among other things, terminal deactivation, user account deactivation, or administrator alarm. The conditions under which the situation will be restored to normal be specified on the action.

In order to prevent denial of service, TOEs usually ensure that there is at least one user account that cannot be disabled.

Further actions for the TSF can be stated by the author of a PP, PP-Module, functional package or ST, including rules for re-enabling 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).

### 12.3.6.3 Element FIA\_AFL.1.1

**The TSF shall detect when [selection (s<sup>1</sup>): *assignment (a<sup>1</sup>): positive integer number*], *an administrator configurable positive integer within [assignment (a<sup>2</sup>): range of acceptable values]* unsuccessful authentication attempts occur related to [assignment (a<sup>3</sup>): *list of authentication events*].**

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should specify the authentication events.

In assignment (a<sup>2</sup>), if the assignment of a positive integer is selected, the author of a PP, PP-Module, functional package or ST should specify the default number (positive integer) of unsuccessful authentication attempts that, when met or surpassed, will trigger the events.

EXAMPLE Examples of these authentication events are:

- the unsuccessful authentication attempts since the last successful authentication for the indicated user identity;
- the unsuccessful authentication attempts since the last successful authentication for the current terminal;
- the number of unsuccessful authentication attempts in the last 10 min.

At least one authentication event shall be specified.

In selection (s<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should select either the assignment of a positive integer, or the phrase "an administrator configurable positive integer" specifying the range of acceptable values.

## ISO/IEC 15408-2:2026(en)

In assignment (a3), if an administrator configurable positive integer is selected, the author of a PP, PP-Module, functional package or ST should specify the range of acceptable values from which the administrator of the TOE may configure the number of unsuccessful authentication attempts. The number of authentication attempts should be less than or equal to the upper bound and greater or equal to the lower bound values.

### 12.3.6.4 Element FIA\_AFL.1.2

**When the defined number of unsuccessful authentication attempts has been [selection <sup>(s1)</sup>: *met, surpassed*], the TSF shall [assignment <sup>(a1)</sup>: *list of actions*].**

In selection (s1), the author of a PP, PP-Module, functional package or ST should select whether the event of meeting or surpassing the defined number of unsuccessful authentication attempts shall trigger an action by the TSF.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the actions to be taken in case the threshold is met or surpassed, as selected. These actions can be disabling of an account for 5 minutes, disabling the terminal for an increasing amount of time (2 to the power of the number of unsuccessful attempts in seconds), or disabling of the account until unlocked by the administrator and simultaneously informing the administrator. The actions should specify the measures and if applicable the duration of the measure (or the conditions under which the measure will be ended).

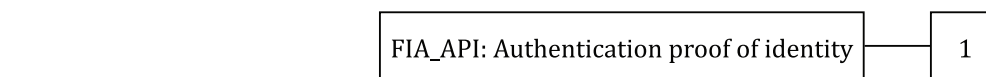
## 12.4 Authentication proof of identity (FIA\_API)

### 12.4.1 Family Behaviour

This family defines functions provided by the TOE to prove its identity and hence allow for verification of the TOE by an external entity in the TOE's IT environment.

### 12.4.2 Component levelling and description

[Figure 41](#) depicts the component levelling of family FIA\_API.



#### Key

1 component

**Figure 41 — FIA\_API: Component levelling**

FIA\_API.1 (Authentication proof of identity) ([12.4.6](#)) provides proof of the identity of the TOE to an external entity.

### 12.4.3 Component management

The following action related to FIA\_API.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- management of authentication information used to prove the claimed identity.

### 12.4.4 Component audit

There are no auditable events foreseen related to FIA\_API.1.

### 12.4.5 Application notes

The other families of FIA (Identification and authentication) describe only the authentication verification of users' identity performed by the TOE and do not describe the functionality of the user to prove their identity.

## ISO/IEC 15408-2:2026(en)

The family FIA\_API (Authentication proof of identity) allows the specification the functionality allowing a TOE to prove its own identity.

### 12.4.6 FIA\_API.1 Authentication proof of identity

#### 12.4.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 12.4.6.2 Component rationale

FIA\_API.1 (Authentication proof of identity) allows the specification of the authentication mechanism used to support proving the identity of the TOE to external entities.

#### 12.4.6.3 Element FIA\_API.1.1

**The TSF shall provide an [assignment <sup>(a1)</sup>: *authentication mechanism*] to prove the identity of [assignment <sup>(a2)</sup>: *entity*] by including the following properties [assignment <sup>(a3)</sup>: *list of properties*] to an external entity.**

In assignment (a1), the first assignment is where a PP, PP-Module, functional package or ST author specifies the authentication mechanism to be used.

EXAMPLE Examples of such an authentication method is "an Authentication Mechanism based on Triple-DES" and "Chip Authentication Protocol according to TR-03110".

In assignment (a2), the second assignment allows the author of a PP, PP-Module, functional package or ST to specify to what entity the proof of identity is associated with.

In assignment (a3), the third assignment is used to provide a list of properties. The property list may include roles or credentials.

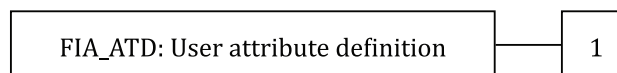
### 12.5 User attribute definition (FIA\_ATD)

#### 12.5.1 Family Behaviour

All authorized users may have a set of security attributes, other than the user's identity, that are used to enforce the SFRs. This family defines the requirements for associating user security attributes with users as needed to support the TSF in making security decisions.

#### 12.5.2 Component levelling and description

[Figure 42](#) illustrates the component levelling of family FIA\_ATD.



#### Key

1 component

**Figure 42 — FIA\_ATD: Component levelling**

FIA\_ATD.1 (User attribute definition) ([12.5.6](#)), allows user security attributes for each user to be maintained individually.

**ISO/IEC 15408-2:2026(en)****12.5.3 Component management**

The following action related to FIA\_ATD.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- if indicated in the assignment, the authorized administrator can be able to define additional security attributes for users.

**12.5.4 Component audit**

There are no auditable events foreseen related to FIA\_ATD.1.

**12.5.5 Application notes**

All authorized users may have a set of security attributes, other than the user's identity, that are used to enforce the SFRs. This family defines the requirements for associating user security attributes with users as needed to support the TSF in making security decisions.

There are dependencies on the individual security policy (SFP) definitions. These individual definitions should contain the listing of attributes that are necessary for policy enforcement.

**12.5.6 FIA\_ATD.1 User attribute definition****12.5.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**12.5.6.2 Component rationale**

This component specifies the security attributes that should be maintained at the level of the user. This means that the security attributes listed are assigned to the user and can be changed at the level of the user. In other words, changing a security attribute in this list associated with a user should have no impact on the security attributes of any other user.

In case security attributes belong to a group of users (such as a capability list for a group), the user is required to have a reference (as a security attribute) to the relevant group.

**12.5.6.3 Element FIA\_ATD.1.1**

**The TSF shall maintain the following list of security attributes belonging to individual users: [assignment <sup>(a1)</sup>: *list of security attributes*].**

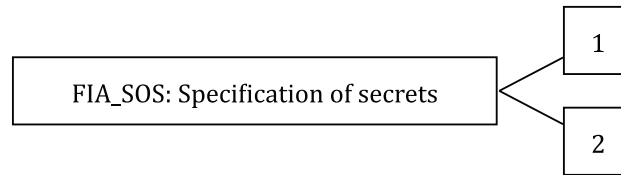
In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the security attributes that are associated to an individual user.

**12.6 Specification of secrets (FIA\_SOS)****12.6.1 Family Behaviour**

This family defines requirements for mechanisms that enforce defined quality metrics on provided secrets and generate secrets to satisfy the defined metric.

**12.6.2 Component levelling and description**

[Figure 43](#) illustrates the component levelling of family FIA\_SOS.

**ISO/IEC 15408-2:2026(en)****Key**

1,2 components

**Figure 43 — FIA\_SOS: Component levelling**

FIA\_SOS.1 (Verification of secrets) ([12.6.6](#)) requires the TSF to verify that secrets meet defined quality metrics.

FIA\_SOS.2 (TSF Generation of secrets) ([12.6.7](#)) requires the TSF to be able to generate secrets that meet defined quality metrics.

**12.6.3 Component management**

The following action related to FIA\_SOS.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the management of the metric used to verify the secrets.

The following action related to FIA\_SOS.2 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the management of the metric used to generate the secrets.

**12.6.4 Component audit**

The following actions related to FIA\_SOS.1 and FIA\_SOS.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: rejection by the TSF of any tested secret;
- basic: rejection or acceptance by the TSF of any tested secret;
- detailed: identification of any changes to the defined quality metrics.

**12.6.5 Application notes**

This family defines requirements for mechanisms that enforce defined quality metrics on provided secrets and generate secrets to satisfy the defined metric.

**EXAMPLE 1** Examples of such mechanisms can include automated checking of user supplied passwords, or automated password generation.

A secret can be generated outside the TOE.

**EXAMPLE 2** An example of a secret generated outside of the TOE can be one that is selected by the user and introduced in the TOE.

In such cases, the FIA\_SOS.1 (Verification of secrets) ([12.6.6](#)) component can be used to ensure that the external generated secret adheres to certain standards, for example a minimum size, not present in a dictionary, and/or not previously used.

Secrets can also be generated by the TOE. In those cases, the FIA\_SOS.2 (TSF Generation of secrets) ([12.6.7](#)) component can be used to require the TOE to ensure that the secrets that will adhere to some specified metrics.

**ISO/IEC 15408-2:2026(en)**

Secrets contain the authentication data provided by the user for an authentication mechanism that is based on knowledge the user possesses. When cryptographic keys are employed, FCS (Cryptographic support) should be used instead of this family.

**12.6.6 FIA\_SOS.1 Verification of secrets****12.6.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**12.6.6.2 Component rationale**

Secrets can be generated by the user. This component ensures that those user generated secrets can be verified to meet a certain quality metric.

**12.6.6.3 Element FIA\_SOS.1.1**

**The TSF shall provide a mechanism to verify that secrets meet [assignment (a1): *a defined quality metric*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST provides a defined quality metric. The quality metric specification may be as simple as a description of the quality checks to be performed, or as formal as a reference to a government standard that defines the quality metrics that secrets must meet.

**12.6.7 FIA\_SOS.2 TSF Generation of secrets****12.6.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**12.6.7.2 Component rationale**

This component allows the TSF to generate secrets for specific functions such as authentication by means of passwords.

When a pseudo-random number generator is used in a secret generation algorithm, it should accept as input random data that would provide output that has a high degree of unpredictability. This random data (seed) can be derived from a number of available parameters such as a system clock, system registers, date, time, etc. The parameters should be selected to ensure that the number of unique seeds that can be generated from these inputs should be at least equal to the minimum number of secrets that are required to be generated.

**12.6.7.3 Element FIA\_SOS.2.1**

**The TSF shall provide a mechanism to generate secrets that meet [assignment (a1): *a defined quality metric*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST provides a defined quality metric. The quality metric specification can be as simple as a description of the quality checks to be performed or as formal as a reference to a government standard that defines the quality metrics that secrets must meet.

**12.6.7.4 Element FIA\_SOS.2.2**

**The TSF shall be able to enforce the use of TSF generated secrets for [assignment (a1): *list of TSF functions*].**

## ISO/IEC 15408-2:2026(en)

In assignment (a1), the author of a PP, PP-Module, functional package or ST provides a list of TSF functions for which the TSF generated secrets shall be used.

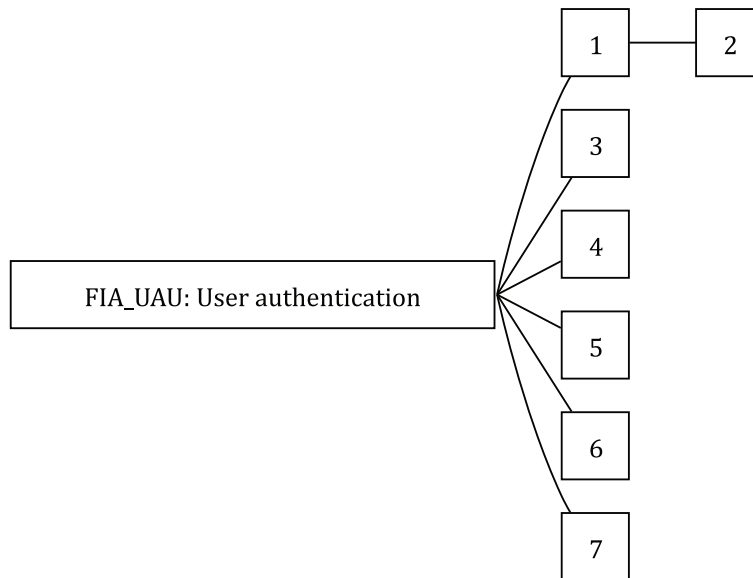
### 12.7 User authentication (FIA\_UAU)

#### 12.7.1 Family Behaviour

This family defines the types of user authentication mechanisms supported by the TSF. This family also defines the required attributes on which the user authentication mechanisms be based.

#### 12.7.2 Component levelling and description

[Figure 44](#) illustrates the component levelling of family FIA\_UAU.



#### Key

1,2,3,4,5,6,7 components

**Figure 44 — FIA\_UAU: Component levelling**

FIA\_UAU.1 (Timing of authentication) ([12.7.6](#)), allows a user to perform certain actions prior to the authentication of the user's identity.

FIA\_UAU.2 (User authentication before any action) ([12.7.7](#)) requires that users are authenticated before any other action will be allowed by the TSF.

FIA\_UAU.3 (Unforgeable authentication) ([12.7.8](#)) 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) ([12.7.9](#)) requires an authentication mechanism that operates with single-use authentication data.

FIA\_UAU.5 (Multiple authentication mechanisms) ([12.7.10](#)) requires that different authentication mechanisms be provided and used to authenticate user identities for specific events.

FIA\_UAU.6 (Re-authenticating) ([12.7.11](#)) requires the ability to specify events for which the user is re-authenticated.

FIA\_UAU.7 (Protected authentication feedback) ([12.7.12](#)) requires that only limited feedback information is provided to the user during the authentication.

## ISO/IEC 15408-2:2026(en)

### 12.7.3 Component management

The following actions related to FIA\_UAU.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- management of the authentication data by an administrator;
- management of the authentication data by the associated user;
- managing the list of actions that can be taken before the user is authenticated.

The following actions related to FIA\_UAU.2 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- management of the authentication data by an administrator;
- management of the authentication data by the user associated with this data.

The following action related to FIA\_UAU.5 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the management of authentication mechanisms.

The following action related to FIA\_UAU.6 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- if an authorized administrator can request re-authentication, the management includes a re-authentication request.

The following action related to FIA\_UAU.7 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the management of the limited feedback information provided to the user during the authentication.

There are no management activities foreseen related to FIA\_UAU.3 or FIA\_UAU.4.

### 12.7.4 Component audit

The following actions related to FIA\_UAU.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: unsuccessful use of the authentication mechanism;
- basic: all use of the authentication mechanism;
- detailed: all TSF mediated actions performed before authentication of the user.

The following actions related to FIA\_UAU.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: unsuccessful use of the authentication mechanism;
- basic: all use of the authentication mechanism.

The following actions related to FIA\_UAU.3 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: detection of fraudulent authentication data;
- basic: all immediate measures taken and results of checks on the fraudulent data.

**ISO/IEC 15408-2:2026(en)**

The following actions related to FIA\_UAU.4 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: attempts to reuse authentication data.

The following actions related to FIA\_UAU.5 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the final decision on authentication;
- basic: the result of each activated mechanism together with the final decision.

The following actions related to FIA\_UAU.6 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: failure of re-authentication;
- basic: all re-authentication attempts.

There are no auditable events foreseen related to FIA\_UAU.7.

### 12.7.5 Application notes

This family defines the types of user authentication mechanisms supported by the TSF. This family defines the required attributes on which the user authentication mechanisms are based.

### 12.7.6 FIA\_UAU.1 Timing of authentication

#### 12.7.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

#### 12.7.6.2 Component rationale

This component requires that the author of a PP, PP-Module, functional package or ST define the TSF-mediated actions that can be performed by the TSF on behalf of the user before the claimed identity of the user is authenticated. The TSF-mediated actions should have no security concerns with users incorrectly identifying themselves prior to being authenticated. For all other TSF-mediated actions not in the list, the user shall be authenticated before the action can be performed by the TSF on behalf of the user.

This component cannot control whether the actions can also be performed before the identification took place. This requires the use of either FIA\_UID.1 (Timing of identification) ([12.8.6](#)) or FIA\_UID.2 (User identification before any action) ([12.8.7](#)) with the appropriate assignments.

#### 12.7.6.3 Element FIA\_UAU.1.1

**The TSF shall allow [assignment (a1): *list of TSF mediated actions*] on behalf of the user to be performed before the user is authenticated.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies a list of TSF-mediated actions that can be performed by the TSF on behalf of a user before the claimed identity of the user is authenticated. This list cannot be empty. If no actions are appropriate, component FIA\_UAU.2 (User authentication before any action) ([12.7.7](#)) should be used instead.

#### 12.7.6.4 Element FIA\_UAU.1.2

**The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.**

**ISO/IEC 15408-2:2026(en)****12.7.7 FIA\_UAU.2 User authentication before any action****12.7.7.1 Component relationships**

Hierarchical to: FIA\_UAU.1 (Timing of authentication) ([12.7.6](#))

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

**12.7.7.2 Component rationale**

This component requires that a user is authenticated before any other TSF-mediated action can take place on behalf of that user.

**12.7.7.3 Element FIA\_UAU.2.1**

The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

**12.7.8 FIA\_UAU.3 Unforgeable authentication****12.7.8.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**12.7.8.2 Component rationale**

This component addresses requirements for mechanisms that provide protection of authentication data. Authentication data that is copied from another user or is in some way constructed should be detected and/or rejected. These mechanisms provide confidence that users authenticated by the TSF are actually who they claim to be.

This component may be useful only with authentication mechanisms that are based on authentication data that cannot be shared. It is impossible for a TSF to detect or prevent the sharing of passwords outside the control of the TSF.

**12.7.8.3 Element FIA\_UAU.3.1**

**The TSF shall [selection <sup>(s1)</sup>: *detect, prevent*] use of authentication data that has been forged by any user of the TSF.**

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify whether the TSF will detect, prevent, or detect and prevent forging of authentication data.

**12.7.8.4 Element FIA\_UAU.3.2**

**The TSF shall [selection <sup>(s1)</sup>: *detect, prevent*] use of authentication data that has been copied from any other user of the TSF.**

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify whether the TSF will detect, prevent, or detect and prevent copying of authentication data.

## ISO/IEC 15408-2:2026(en)

### 12.7.9 FIA\_UAU.4 Single-use authentication mechanisms

#### 12.7.9.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 12.7.9.2 Component rationale

This component addresses requirements for authentication mechanisms based on single-use authentication data. Single-use authentication data can be something the user has or knows, but not something the user is.

EXAMPLE Single-use authentication data include single-use passwords, encrypted timestamps, and/or random numbers from a secret lookup table.

The author of a PP, PP-Module, functional package or ST can specify to which authentication mechanism(s) this requirement applies.

#### 12.7.9.3 Element FIA\_UAU.4.1

**The TSF shall prevent reuse of authentication data related to [assignment <sup>(a1)</sup>: *identified authentication mechanism(s)*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of authentication mechanisms to which this requirement applies. This assignment can be "all authentication mechanisms".

### 12.7.10 FIA\_UAU.5 Multiple authentication mechanisms

#### 12.7.10.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 12.7.10.2 Component rationale

The use of this component allows specification of requirements for more than one authentication mechanism to be used within a TOE. For each distinct mechanism, applicable requirements are chosen from the FIA (Identification and authentication) class to be applied to each mechanism. It is possible that the same component is selected multiple times in order to reflect different requirements for the different use of the authentication mechanism.

The management functions in class FMT (Security management) ([Clause 13](#)) provide maintenance capabilities for the set of authentication mechanisms, as well as the rules that determine whether the authentication was successful.

To allow anonymous users to interact with the TOE, a "none" authentication mechanism may be incorporated. The use of such access is clearly explained in the rules of FIA\_UAU.5.2.

#### 12.7.10.3 Element FIA\_UAU.5.1

**The TSF shall provide [assignment <sup>(a1)</sup>: *list of multiple authentication mechanisms*] to support user authentication.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST defines the available authentication mechanisms.

**ISO/IEC 15408-2:2026(en)****12.7.10.4 Element FIA\_UAU.5.2**

**The TSF shall authenticate any user's claimed identity according to the [assignment (a1): *rules describing how the multiple authentication mechanisms provide authentication*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the rules that describe how the authentication mechanisms provide authentication and when each is intended to be used. This means that for each situation the set of mechanisms used for authenticating the user shall be described.

**12.7.11 FIA\_UAU.6 Re-authenticating****12.7.11.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**12.7.11.2 Component rationale**

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.

**12.7.11.3 Element FIA\_UAU.6.1**

**The TSF shall re-authenticate the user under the conditions [assignment (a1): *list of conditions under which re-authentication is required*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the list of conditions requiring re-authentication. This list may 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.

**12.7.12 FIA\_UAU.7 Protected authentication feedback****12.7.12.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FIA\_UAU.1 (Timing of authentication) ([12.7.6](#))

**12.7.12.2 Component rationale**

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 can not be appropriate.

This component requires that the authentication data is not provided as-is back to the user. In a workstation environment, it can display a substitute character for each password character provided, and not the original character.

EXAMPLE A star "\*" character.

**12.7.12.3 Element FIA\_UAU.7.1**

**The TSF shall provide only [assignment (a1): *list of feedback*] to the user while the authentication is in progress.**

## ISO/IEC 15408-2:2026(en)

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the feedback related to the authentication process that will be provided to the user.

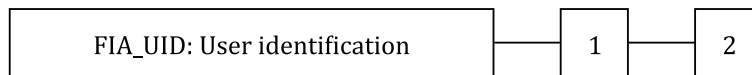
### 12.8 User identification (FIA\_UID)

#### 12.8.1 Family Behaviour

This family defines the conditions under which users shall be required to identify themselves before performing any other actions that are mediated by the TSF and which require user identification.

#### 12.8.2 Component levelling and description

[Figure 45](#) illustrates the component levelling of family FIA\_UID.



#### Key

1,2 components

**Figure 45 — FIA\_UID: Component levelling**

FIA\_UID.1 (Timing of identification) ([12.8.6](#)), allows users to perform certain actions before being identified by the TSF.

FIA\_UID.2 (User identification before any action) ([12.8.7](#)) requires that users identify themselves before any other action will be allowed by the TSF.

#### 12.8.3 Component management

The following actions related to FIA\_UID.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the management of the user identities;
- if an authorized administrator can change the actions allowed before identification, the managing of the action lists.

The following actions related to FIA\_UID.2 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- the management of the user identities.

#### 12.8.4 Component audit

The following actions related to FIA\_UID.1 and FIA\_UID.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: unsuccessful use of the user identification mechanism, including the user identity provided;
- basic: all use of the user identification mechanism, including the user identity provided.

#### 12.8.5 Application notes

This family defines the conditions under which users are required to identify themselves before performing any other actions that shall be mediated by the TSF and that require user identification.

**ISO/IEC 15408-2:2026(en)****12.8.6 FIA\_UID.1 Timing of identification****12.8.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**12.8.6.2 Component rationale**

This component poses requirements for the user to be identified. The author of a PP, PP-Module, functional package or ST may indicate specific actions that are performed before the identification takes place.

If FIA\_UID.1 (Timing of identification) is used, the TSF-mediated actions mentioned in FIA\_UID.1 (Timing of identification) should also appear in this FIA\_UAU.1 (Timing of authentication) ([12.7.6](#)).

**12.8.6.3 Element FIA\_UID.1.1**

**The TSF shall allow [assignment <sup>(a1)</sup>: *list of TSF-mediated actions*] on behalf of the user to be performed before the user is identified.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies a list of TSF-mediated actions that can be performed by the TSF on behalf of a user before the user is identified. If no actions are appropriate, component FIA\_UID.2 (User identification before any action) ([12.8.7](#)) should be used instead.

**12.8.6.4 Element FIA\_UID.1.2**

**The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.**

**12.8.7 FIA\_UID.2 User identification before any action****12.8.7.1 Component relationships**

Hierarchical to: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

Dependencies: No dependencies.

**12.8.7.2 Component rationale**

In this component users will be identified. A user is not allowed by the TSF to perform any other action before being identified.

**12.8.7.3 Element FIA\_UID.2.1**

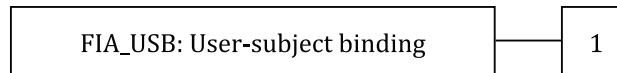
The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

**12.9 User-subject binding (FIA\_USB)****12.9.1 Family Behaviour**

An authenticated user, in order to use the TOE, typically activates a subject. The user's security attributes are associated (totally or partially) with this subject. This family defines requirements to create and maintain the association of the user's security attributes to a subject acting on the user's behalf.

## 12.9.2 Component levelling and description

[Figure 46](#) illustrates the component levelling of family FIA\_USB.



### Key

1 component

**Figure 46 — FIA\_USB: Component levelling**

FIA\_USB.1 (User-subject binding) ([12.9.6](#)) requires the specification of any rules governing the association between user attributes and the subject attributes into which they are mapped.

## 12.9.3 Component management

The following actions related to FIA\_USB.1 can be considered for the management functions in class FMT (Security management) ([Clause 13](#)):

- an authorized administrator can define default subject security attributes;
- an authorized administrator can change subject security attributes.

## 12.9.4 Component audit

The following actions related to FIA\_USB.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: unsuccessful binding of user security attributes to a subject;
- basic: success and failure of binding of user security attributes to a subject.

## 12.9.5 Application notes

An authenticated user, in order to use the TOE, typically activates a subject. The user's security attributes are associated (totally or partially) with this subject. This family defines requirements to create and maintain the association of the user's security attributes to a subject acting on the user's behalf.

## 12.9.6 FIA\_USB.1 User-subject binding

### 12.9.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FIA\_ATD.1 (User attribute definition) ([12.5.6](#))

### 12.9.6.2 Component rationale

It is intended that a subject is acting on behalf of the user who caused the subject to come into being or to be activated to perform a certain task.

Therefore, when a subject is created, that subject is acting on behalf of the user who initiated the creation. In cases where anonymity is used, the subject is still acting on behalf of a user, but the identity of that user is unknown. A special category of subjects is those subjects that serve 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.

### 12.9.6.3 Element FIA\_USB.1.1

**The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: [assignment <sup>(a1)</sup>: *list of user security attributes*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify a list of the user security attributes that shall be bound to subjects.

### 12.9.6.4 Element FIA\_USB.1.2

**The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: [assignment <sup>(a1)</sup>: *rules for the initial association of attributes*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify any rules that are to apply upon initial association of attributes with subjects, or "none".

### 12.9.6.5 Element FIA\_USB.1.3

**The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: [assignment <sup>(a1)</sup>: *rules for the changing of attributes*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify any rules that are to apply when changes are made to the user security attributes associated with subjects acting on behalf of users, or "none".

## 13 Class FMT Security management

### 13.1 Introduction

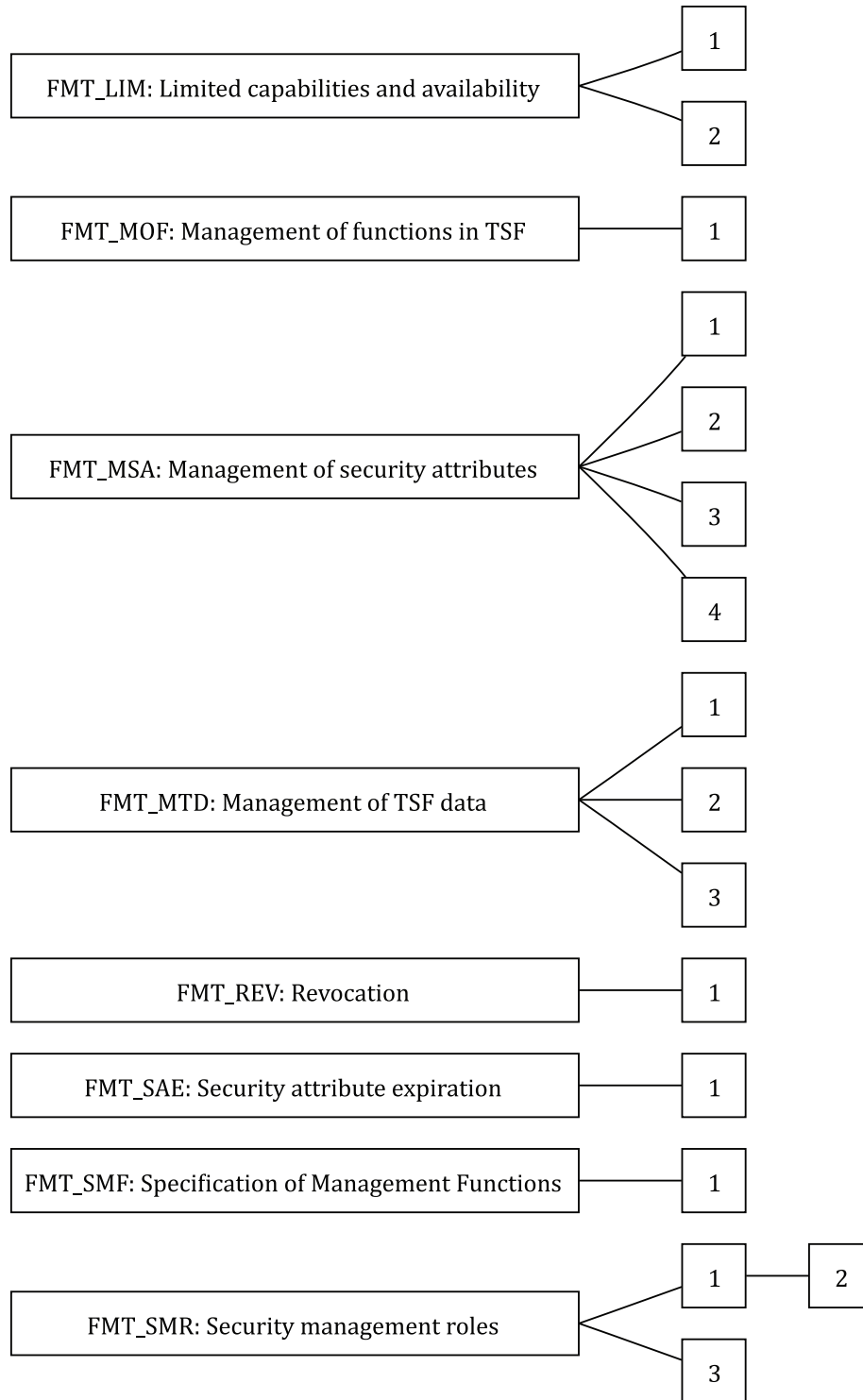
This class is intended to specify the management of several aspects of the TSF: security attributes, TSF data and functions. The different management roles and their interaction, such as separation of capability, can be specified.

This class has the following objectives:

- management of TSF data;
- management of security attributes;
- management of functions of the TSF;
- definition of security roles.

[Figure 47](#) describes the family hierarchy of class FMT.

**ISO/IEC 15408-2:2026(en)**



**Key**  
 1,2,3,4 components

**Figure 47 — FMT: Security management, class decomposition**

Direct, indirect and optional dependencies for the components of class FMT are shown in [Table 6](#).

**ISO/IEC 15408-2:2026(en)**

**Table 6 — Dependency table for class FMT: Security management**

	FDP_ACC.1 (11.3.6)	FDP_IFC.1 (11.7.6)	FIA_UID.1 (12.8.6)	FMT_LIM.1 (13.3.6)	FMT_LIM.2 (13.3.7)	FMT_MSA.1 (13.5.6)	FMT_MTD.1 (13.6.6)	FMT_SMF.1 (13.9.6)	FMT_SMR.1 (13.10.6)	FPT_STM.1 (15.14.6)
FMT_LIM.1 (13.3.6)				-	X					
FMT_LIM.2 (13.3.7)				X	-					
FMT_MOF.1 (13.4.6)			-					X	X	
FMT_MSA.1 (13.5.6)	O <sup>1</sup>	O <sup>1</sup>	-					X	X	
FMT_MSA.2 (13.5.7)	O <sup>1</sup>	O <sup>1</sup>	-			X		-	X	
FMT_MSA.3 (13.5.8)	-	-	-			X		-	X	
FMT_MSA.4 (13.5.9)	O <sup>1</sup>	O <sup>1</sup>								
FMT_MTD.1 (13.6.6)			-					X	X	
FMT_MTD.2 (13.6.7)			-				X	-	X	
FMT_MTD.3 (13.6.8)			-				X	-	-	
FMT_REV.1 (13.7.6)			-						X	
FMT_SAE.1 (13.8.6)			-						X	X
FMT_SMF.1 (13.9.6)										
FMT_SMR.1 (13.10.6)			X							
FMT_SMR.2 (13.10.7)			X							
FMT_SMR.3 (13.10.8)			-						X	

## 13.2 Notes on class FMT

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) (13.7.6), or FMT\_SAE.1 (Time-limited authorization) (13.8.6), where the behaviour can be impaired. In such situations, use of components from FPT\_TRC (Internal TOE TSF data replication consistency) (15.17) is advisable.

The FMT\_LIM (Limited capabilities and availability) (13.3) family provides requirements that allow the specification of a policy that limits the capabilities and the availability of TSF functions. This is useful when a PP, PP-Module, functional package or ST author needs to enforce design principles such as least privilege and attack surface minimization.

NOTE These, and other architectural and design principles along with appropriate evaluation considerations are discussed in ISO/IEC/TS 19249.[5]

## 13.3 Limited capabilities and availability (FMT\_LIM)

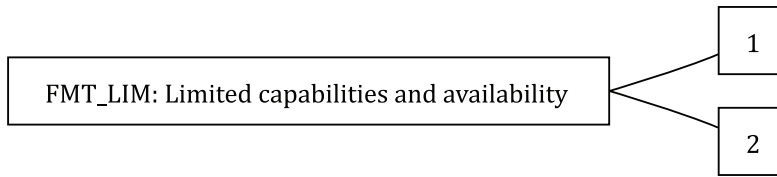
### 13.3.1 Family Behaviour

This family defines requirements that limit the capabilities and availability of functions in a combined manner.

NOTE FDP\_ACF (Access control functions) restricts the access to functions whereas the component Limited Capability of this family requires the functions themselves to be designed in a specific manner.

### 13.3.2 Component levelling and description

[Figure 48](#) illustrates the component levelling of family FMT\_LIM.



#### Key

1,2 components

**Figure 48 — FMT\_LIM: Component levelling**

FMT\_LIM.1 (Limited capabilities) ([13.3.6](#)) requires that the TSF is built to provide only the capabilities (perform action, gather information) necessary for its genuine purpose.

FMT\_LIM.2 (Limited availability) ([13.3.7](#)) requires that the TSF restrict the use of functions (refer to FMT\_LIM.1 (Limited capabilities) ([13.3.6](#))). This can be achieved, for instance, by removing or by disabling functions in a specific phase of the TOE's life-cycle.

### 13.3.3 Component management

There are no management activities foreseen related to FMT\_LIM.1 or FMT\_LIM.2.

### 13.3.4 Component audit

There are no auditable events foreseen related to FMT\_LIM.1 or FMT\_LIM.2.

### 13.3.5 Application notes

The functional requirements FMT\_LIM.1 (Limited capabilities) ([13.3.6](#)) and FMT\_LIM.2 (Limited availability) ([13.3.7](#)) assume that there are two types of mechanisms (limitation of capabilities and limitation of availability) which together provide protection in order to enforce the policy. This also allows that:

- 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
- 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.

### 13.3.6 FMT\_LIM.1 Limited capabilities

#### 13.3.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FMT\_LIM.2 (Limited availability) ([13.3.7](#))

#### 13.3.6.2 Component rationale

This component ensures that the TSF only provides the minimum set of capabilities necessary for its intended operation.

**EXAMPLE** An example of a limited capability is JTAG interface enablement, which can be either enabled or disabled.

### 13.3.6.3 Element FMT\_LIM.1.1

**The TSF shall limit its capabilities so that in conjunction with FMT\_LIM.2 (Limited availability) (13.3.7) the following policy is enforced: [assignment (a1): *Limited capability and availability policy*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the limited capability policy.

### 13.3.7 FMT\_LIM.2 Limited availability

#### 13.3.7.1 Component relationships

Hierarchical to: No other components.

Dependencies: FMT\_LIM.1 (Limited capabilities) (13.3.6)

#### 13.3.7.2 Component rationale

This component ensures that certain TSF capabilities are only available during specific phases of the TOE's life-cycle.

**EXAMPLE** An example of a limited availability is JTAG interface enablement, which can be either enabled or disabled before operational use of the TOE.

#### 13.3.7.3 Element FMT\_LIM.2.1

**The TSF shall be designed in a manner that limits its availability so that in conjunction with FMT\_LIM.1 (Limited capabilities) (13.3.6) the following policy is enforced: [assignment (a1): *Limited capability and availability policy*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the limited availability policy.

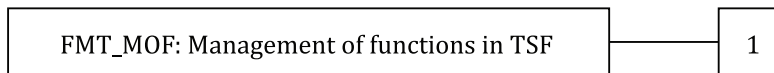
## 13.4 Management of functions in TSF (FMT\_MOF)

### 13.4.1 Family Behaviour

This family allows authorized users to control over the management of functions in the TSF.

### 13.4.2 Component levelling and description

[Figure 49](#) illustrates the component levelling of family FMT\_MOF.



#### Key

1 component

**Figure 49 — FMT\_MOF: Component levelling**

FMT\_MOF.1 (Management of security functions behaviour) (13.4.6) allows the authorized users (roles) to manage the behaviour of functions in the TSF that use rules or have specified conditions that may be manageable.

**ISO/IEC 15408-2:2026(en)****13.4.3 Component management**

The following action related to FMT\_MOF.1 can be considered for the management functions in FMT (Security management):

- managing the group of roles that can interact with the functions in the TSF.

**13.4.4 Component audit**

The following action related to FMT\_MOF.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- basic: all modifications in the behaviour of the functions in the TSF.

**13.4.5 Application 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:

- 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 1** 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.

- management functions that relate to controls over availability;

**EXAMPLE 2** Definition and update of availability parameters or resource quotas.

- management functions that relate to general installation and configuration;

**EXAMPLE 3** TOE configuration, manual recovery, installation of TOE security fixes (if any), repair and reinstallation of hardware.

- management functions that relate to routine control and maintenance of TOE resources.

**EXAMPLE 4** Enabling and disabling peripheral devices, mounting of removable storage media, backup, and recovery.

These functions are required to be present in a TOE based on the families included in the PP or ST. It is the responsibility of the author of a PP, PP-Module, functional package or ST to ensure that adequate functions will be provided to manage the TOE in a secure fashion.

The TSF can contain functions that can be controlled by an administrator.

**EXAMPLE 5** The auditing functions can be switched off, the time synchronization can be switchable, and/or the authentication mechanism can be modifiable.

## ISO/IEC 15408-2:2026(en)

### 13.4.6 FMT\_MOF.1 Management of security functions behaviour

#### 13.4.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FMT\_SMR.1 (Security roles) ([13.10.6](#))  
FMT\_SMF.1 (Specification of Management Functions) ([13.9.6](#))

#### 13.4.6.2 Component rationale

This component allows identified roles to manage the security functions of the TSF. This can 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 involves changing authentication mechanisms.

#### 13.4.6.3 Element FMT\_MOF.1.1

**The TSF shall restrict the ability to [selection <sup>(s1)</sup>: *determine the behaviour of, disable, enable, modify the behaviour of*] the functions [assignment <sup>(a1)</sup>: *list of functions*] to [assignment <sup>(a2)</sup>: *the authorized identified roles*].**

In selection (s1), the author of a PP, PP-Module, functional package or ST should select whether the role can determine the behaviour of, disable, enable, and/or modify the behaviour of the security functions.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the functions that can be modified by the identified roles. Examples include auditing and time determination.

In assignment (a2), the author of a PP, PP-Module, functional package or ST 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) ([13.10.6](#)).

### 13.5 Management of security attributes (FMT\_MSA)

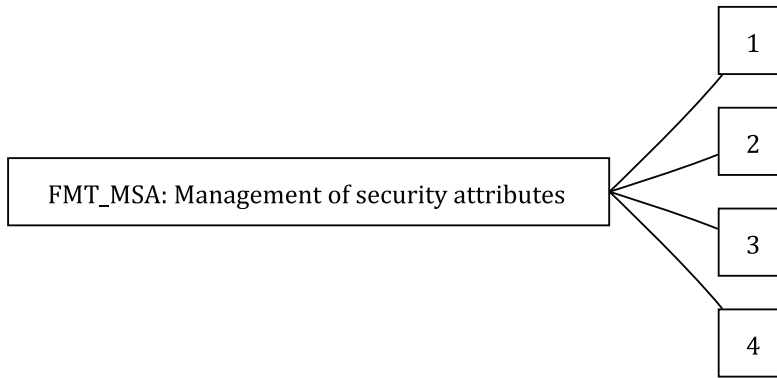
#### 13.5.1 Family Behaviour

This family allows authorized users control over the management of security attributes. This management can include capabilities for viewing and modifying of security attributes.

#### 13.5.2 Component levelling and description

[Figure 50](#) illustrates the component levelling of family FMT\_MSA.

## ISO/IEC 15408-2:2026(en)



### Key

1,2,3,4 components

**Figure 50 — FMT\_MSA: Component levelling**

FMT\_MSA.1 (Management of security attributes) ([13.5.6](#)) allows authorized users (roles) to manage the specified security attributes.

FMT\_MSA.2 (Secure security attributes) ([13.5.7](#)) ensures that values assigned to security attributes are valid with respect to the secure state.

FMT\_MSA.3 (Static attribute initialization) ([13.5.8](#)) ensures that the default values of security attributes are appropriately either permissive or restrictive in nature.

FMT\_MSA.4 (Security attribute value inheritance) ([13.5.9](#)) allows the rules/policies to be specified that will dictate the value to be inherited by a security attribute.

### 13.5.3 Component management

The following actions related to FMT\_MSA.1 can be considered for the management functions in FMT (Security management):

- managing the group of roles that can interact with the security attributes;
- managing the rules by which security attributes inherit specified values.

The following action related to FMT\_MSA.2 can be considered for the management functions in FMT (Security management):

- managing the rules by which security attributes inherit specified values.

The following actions related to FMT\_MSA.3 can be considered for the management functions in FMT (Security management):

- managing the group of roles that can specify initial values;
- managing the permissive or restrictive setting of default values for a given access control SFP;
- managing the rules by which security attributes inherit specified values.

The following action related to FMT\_MSA.4 can be considered for the management functions in FMT (Security management):

- specification of the role permitted to establish or modify security attributes.

## ISO/IEC 15408-2:2026(en)

### 13.5.4 Component audit

The following action related to FMT\_MSA.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- basic: all modifications of the values of security attributes.

The following actions related to FMT\_MSA.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: all offered and rejected values for a security attribute;
- detailed: all offered and accepted secure values for a security attribute.

The following actions related to FMT\_MSA.3 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- basic: modifications of the default setting of permissive or restrictive rules;
- basic: all modifications of the initial values of security attributes.

The following action related to FMT\_MSA.4 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- basic: modifications of security attributes, possibly with the old and/or values of security attributes that were modified.

### 13.5.5 Application notes

This family defines the requirements on the management of security attributes.

Security attributes affect the behaviour of the TSF.

**EXAMPLE** Examples of security attributes are the groups to which a user belongs, the roles they can assume, the priority of a process (subject), and the rights belonging to a role or a user.

These security attributes can need to be managed by the user, a subject, a specific authorized user (a user with explicitly given rights for this management) or inherit values according to a given policy/set of rules.

It is noted that the right to assign rights to users is itself a security attribute and/or potentially subject to management by FMT\_MSA.1 (Management of security attributes) ([13.5.6](#)).

FMT\_MSA.2 (Secure security attributes) ([13.5.7](#)) can be used to ensure that any accepted combination of security attributes is within a secure state. The definition of what "secure" means is left to the TOE guidance.

In some instances, subjects, objects, or user accounts are created. If no explicit values for the related security attributes are given, default values are required to be used. FMT\_MSA.1 (Management of security attributes) ([13.5.6](#)) can be used to specify that these default values can be managed.

### 13.5.6 FMT\_MSA.1 Management of security attributes

#### 13.5.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]  
FMT\_SMR.1 (Security roles) ([13.10.6](#))  
FMT\_SMF.1 (Specification of Management Functions) ([13.9.6](#))

**ISO/IEC 15408-2:2026(en)****13.5.6.2 Component rationale**

This component allows users acting in certain roles to manage identified security attributes. The users are assigned to a role within the component FMT\_SMR.1 (Security roles) ([13.10.6](#)).

The default value of a parameter is the value the parameter takes when it is instantiated without specifically assigned values. An initial value is provided during the instantiation (creation) of a parameter and overrides the default value.

**13.5.6.3 Element FMT\_MSA.1.1**

**The TSF shall enforce the [assignment <sup>(a1)</sup>: *access control SFP(s), information flow control SFP(s)*] to restrict the ability to [selection <sup>(s1)</sup>: *change\_default, query, modify, delete, [assignment <sup>(a2)</sup>: other operations]*] the security attributes [assignment <sup>(a3)</sup>: *list of security attributes*] to [assignment <sup>(a4)</sup>: *the authorized identified roles*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should list the access control SFP(s) or the information flow control SFP(s) for which the security attributes are applicable.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the security attributes that can be operated on by the identified roles. It is possible for the author of a PP, PP-Module, functional package or ST to specify that the default value such as default access-rights can be managed.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the operations that can be applied to the identified security attributes. The author of a PP, PP-Module, functional package or ST can specify that the role can modify the default value (change\_default), query, modify the security attribute, delete the security attributes entirely or define their own operation.

EXAMPLE 1 Examples of these security attributes are user-clearance, priority of service level, access control list, default access rights.

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify the roles that are allowed to operate on the security attributes. The possible roles are specified in FMT\_SMR.1 (Security roles) ([13.10.6](#)) Security roles.

In assignment (a4), if selected, the author of a PP, PP-Module, functional package or ST should specify which other operations the role can perform.

EXAMPLE 2 An example of such an operation is "create".

**13.5.7 FMT\_MSA.2 Secure security attributes****13.5.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]  
FMT\_MSA.1 (Management of security attributes) ([13.5.6](#))  
FMT\_SMR.1 (Security roles) ([13.10.6](#))

**13.5.7.2 Component rationale**

This component contains requirements on the values that can be assigned to security attributes. The assigned values should be such that the TOE will remain in a secure state.

The definition of what "secure" means is not answered in this component but is left to the development of the TOE and the resulting information in the guidance. An example can be that if a user account is created, it should have a non-trivial password.

**ISO/IEC 15408-2:2026(en)****13.5.7.3 Element FMT\_MSA.2.1**

**The TSF shall ensure that only secure values are accepted for [assignment (a1): *list of security attributes*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of security attributes that require only secure values to be provided.

**13.5.8 FMT\_MSA.3 Static attribute initialization****13.5.8.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FMT\_MSA.1 (Management of security attributes) ([13.5.6](#))  
FMT\_SMR.1 (Security roles) ([13.10.6](#))

**13.5.8.2 Component rationale**

This component requires that the TSF provide default values for relevant object security attributes, which can be overridden by an initial value. It may still be possible for a new object to have different security attributes at creation if a mechanism exists to specify the permissions at time of creation.

**13.5.8.3 Element FMT\_MSA.3.1**

**The TSF shall enforce the [assignment (a1): *access control SFP, information flow control SFP*] to provide [selection, choose one of (s1): *restrictive, permissive, [assignment (a2): other property]*] default values for security attributes that are used to enforce the SFP.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should list the access control SFP or the information flow control SFP for which the security attributes are applicable.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should select whether the default property of the access control attribute will be restrictive, permissive, or another property. Only one of these options may be chosen.

In selection (s1), if the author of a PP, PP-Module, functional package or ST selects another property, the author of a PP, PP-Module, functional package or ST should specify the desired characteristics of the default values.

**13.5.8.4 Element FMT\_MSA.3.2**

**The TSF shall allow the [assignment (a1): *the authorized identified roles*] to specify alternative initial values to override the default values when an object or information is created.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the roles that are allowed to modify the values of the security attributes. The possible roles are specified in FMT\_SMR.1 (Security roles) ([13.10.6](#)).

**13.5.9 FMT\_MSA.4 Security attribute value inheritance****13.5.9.1 Component relationships**

Hierarchical to: No other components.

Dependencies: [FDP\_ACC.1 (Subset access control) ([11.3.6](#)), or  
FDP\_IFC.1 (Subset information flow control) ([11.7.6](#))]

### 13.5.9.2 Component rationale

This component requires specification of the set of rules through which the security attribute inherits values and the conditions to be met for these rules to be applied.

### 13.5.9.3 Element FMT\_MSA.4.1

**The TSF shall use the following rules to set the value of security attributes: [assignment <sup>(a1)</sup>: *rules for setting the values of security attributes*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the rules governing the value that will be inherited by the specified security attribute, including the conditions that shall be met for the rules to be applied.

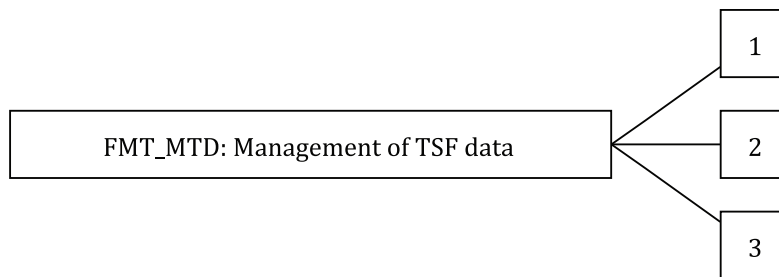
## 13.6 Management of TSF data (FMT\_MTD)

### 13.6.1 Family Behaviour

This family allows authorized users (roles) control over the management of TSF data.

### 13.6.2 Component levelling and description

[Figure 51](#) illustrates the component levelling of family FMT\_MTD.



**Key**

1,2,3 components

**Figure 51 — FMT\_MTD: Component levelling**

FMT\_MTD.1 (Management of TSF data) ([13.6.6](#)) allows authorized users to manage TSF data.

FMT\_MTD.2 (Management of limits on TSF data) ([13.6.7](#)) specifies the action to be taken if limits on TSF data are reached or exceeded.

FMT\_MTD.3 (Secure TSF data) ([13.6.8](#)) ensures that values assigned to TSF data are valid with respect to the secure state.

### 13.6.3 Component management

The following action related to FMT\_MTD.1 can be considered for the management functions in FMT (Security management):

- managing the group of roles that can interact with the TSF data.

The following action related to FMT\_MTD.2 can be considered for the management functions in FMT (Security management):

- managing the group of roles that can interact with the limits on the TSF data.

There are no management activities foreseen related to FMT\_MTD.3.

## ISO/IEC 15408-2:2026(en)

### 13.6.4 Component audit

The following action related to FMT\_MTD.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

— basic: all modifications to the values of TSF data.

The following actions related to FMT\_MTD.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

— basic: all modifications to the limits on TSF data;

— basic: all modifications in the actions to be taken in case of violation of the limits.

The following action related to FMT\_MTD.3 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

— minimal: all rejected values of TSF data.

### 13.6.5 Application notes

This component imposes requirements on the management of TSF data.

EXAMPLE Examples of TSF data are the current time and the audit trail.

This family allows the specification of whom can read, delete, or create the audit trail.

### 13.6.6 FMT\_MTD.1 Management of TSF data

#### 13.6.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FMT\_SMR.1 (Security roles) ([13.10.6](#))  
FMT\_SMF.1 (Specification of Management Functions) ([13.9.6](#))

#### 13.6.6.2 Component rationale

This component allows users with a certain role to manage values of TSF data. The users are assigned to a role within the component FMT\_SMR.1 (Security roles) ([13.10.6](#)).

The default value of a parameter is the values the parameter takes when it is instantiated without specifically assigned values. An initial value is provided during the instantiation (creation) of a parameter and overrides the default value.

#### 13.6.6.3 Element FMT\_MTD.1.1

**The TSF shall restrict the ability to [selection <sup>(s1)</sup>: *change\_default, query, modify, delete, clear, [assignment <sup>(a1)</sup>: other operations]*] the [assignment <sup>(a2)</sup>: *list of TSF data*] to [assignment <sup>(a3)</sup>: *the authorized identified roles*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the operations that can be applied to the identified TSF data. The author of a PP, PP-Module, functional package or ST can specify that the role can modify the default value (change\_default), clear, query or modify the TSF data, or delete the TSF data entirely. If demanded, the author of a PP, PP-Module, functional package or ST can specify any type of operation. To clarify "clear TSF data" means that the content of the TSF data is removed, but that the entity that stores the TSF data remains in the TOE.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the TSF data that can be operated on by the identified roles. It is possible for the author of a PP, PP-Module, functional package or ST to specify that the default value can be managed.

**ISO/IEC 15408-2:2026(en)**

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the roles that are allowed to operate on the TSF data. The possible roles are specified in FMT\_SMR.1 (Security roles) ([13.10.6](#)).

In assignment (a3), if selected, the author of a PP, PP-Module, functional package or ST should specify which other operations the role can perform.

**13.6.7 FMT\_MTD.2 Management of limits on TSF data****13.6.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FMT\_MTD.1 (Management of TSF data) ([13.6.6](#))  
FMT\_SMR.1 (Security roles) ([13.10.6](#))

**13.6.7.2 Component rationale**

This component specifies limits on TSF data, and actions to be taken if these limits are exceeded. This component will allow limits on the size of the audit trail to be defined, and specification of the actions to be taken when these limits are exceeded.

**13.6.7.3 Element FMT\_MTD.2.1**

**The TSF shall restrict the specification of the limits for [assignment <sup>(a1)</sup>: *list of TSF data*] to [assignment <sup>(a2)</sup>: *the authorized identified roles*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the TSF data that can have limits, and the value of those limits. An example of such TSF data is the number of users logged-in.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the roles that are allowed to modify the limits on the TSF data. The possible roles are specified in FMT\_SMR.1 (Security roles) ([13.10.6](#)).

**13.6.7.4 Element FMT\_MTD.2.2**

**The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: [assignment <sup>(a1)</sup>: *actions to be taken*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the actions to be taken if the specified limit on the specified TSF data is exceeded.

**13.6.8 FMT\_MTD.3 Secure TSF data****13.6.8.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FMT\_MTD.1 (Management of TSF data) ([13.6.6](#))

**13.6.8.2 Component rationale**

This component covers requirements on the values that can be assigned to TSF data. The assigned values should be such that the TOE will remain in a secure state.

The definition of what "secure" means is not answered in this component but is left to the development of the TOE and the resulting information in the guidance.

### 13.6.8.3 Element FMT\_MTD.3.1

The TSF shall ensure that only secure values are accepted for [assignment <sup>(a1)</sup>: *list of TSF data*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify what TSF data require only secure values to be accepted.

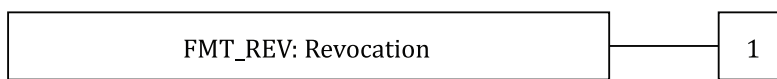
## 13.7 Revocation (FMT\_REV)

### 13.7.1 Family Behaviour

This family addresses revocation of security attributes for a variety of entities within a TOE.

### 13.7.2 Component levelling and description

[Figure 52](#) illustrates the component levelling of family FMT\_REV.



#### Key

1 component

**Figure 52 — FMT\_REV: Component levelling**

FMT\_REV.1 (Revocation) ([13.7.6](#)) provides for revocation of security attributes to be enforced at some point in time.

### 13.7.3 Component management

The following actions related to FMT\_REV.1 can be considered for the management functions in FMT (Security management):

- managing the group of roles that can invoke revocation of security attributes;
- managing the lists of users, subjects, objects, and other resources for which revocation is possible;
- managing the revocation rules.

### 13.7.4 Component audit

The following actions related to FMT\_REV.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: unsuccessful revocation of security attributes;
- basic: all attempts to revoke security attributes.

### 13.7.5 Application notes

This family addresses revocation of security attributes for a variety of entities within a TOE.

### 13.7.6 FMT\_REV.1 Revocation

#### 13.7.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FMT\_SMR.1 (Security roles) ([13.10.6](#))

## ISO/IEC 15408-2:2026(en)

### 13.7.6.2 Component rationale

This component specifies requirements on the revocation of rights. It requires the specification of the revocation rules. Examples are:

- revocation will take place on the next login of the user;
- revocation will take place on the next attempt to open the file;
- revocation will take place within a fixed time. This can mean that all open connections are re-evaluated every x minutes.

### 13.7.6.3 Element FMT\_REV.1.1

**The TSF shall restrict the ability to revoke [assignment <sup>(a1)</sup>: *list of security attributes*] associated with the [selection <sup>(s1)</sup>: *users, subjects, objects, [assignment <sup>(a2)</sup>: other additional resources*] under the control of the TSF to [assignment <sup>(a3)</sup>: *the authorized identified roles*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify which security attributes shall be revoked when a change is made to the associated object/subject/user/other resource.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify whether the ability to revoke security attributes from users, subjects, objects, or any additional resources shall be provided by the TSF.

In selection (s1), the author of a PP, PP-Module, functional package or ST 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) ([13.10.6](#)).

In assignment (a3), the author of a PP, PP-Module, functional package or ST should, if additional resources is selected, specify whether the ability to revoke their security attributes shall be provided by the TSF.

### 13.7.6.4 Element FMT\_REV.1.2

**The TSF shall enforce the rules [assignment <sup>(a1)</sup>: *specification of revocation rules*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the revocation rules. Examples of these rules can include: "prior to the next operation on the associated resource", or "for all new subject creations".

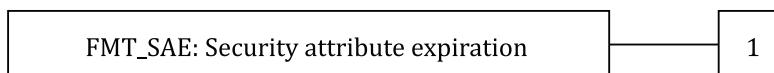
## 13.8 Security attribute expiration (FMT\_SAE)

### 13.8.1 Family Behaviour

This family addresses the capability to enforce time limits for the validity of security attributes.

### 13.8.2 Component levelling and description

[Figure 53](#) illustrates the component levelling of family FMT\_SAE.



#### Key

1 component

**Figure 53 — FMT\_SAE: Component levelling**

FMT\_SAE.1 (Time-limited authorization) ([13.8.6](#)) provides the capability for an authorized user to specify an expiration time on specified security attributes.

**ISO/IEC 15408-2:2026(en)****13.8.3 Component management**

The following actions related to FMT\_SAE.1 can be considered for the management functions in FMT (Security management):

- managing the list of security attributes for which expiration is intended to be supported;
- the actions to be taken if the expiration time has passed.

**13.8.4 Component audit**

The following actions related to FMT\_SAE.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- basic: specification of the expiration time for an attribute;
- basic: action taken due to attribute expiration.

**13.8.5 Application notes**

This family addresses the capability to enforce time limits for the validity of security attributes. This family can be applied to specify expiration requirements for access control attributes, identification and authentication attributes, certificates, audit attributes, etc.

EXAMPLE An example of a certificate is key certificates such as ITU-T Recommendation X.509,<sup>[8]</sup>ISO/IEC 9594-8<sup>[2]</sup>.

**13.8.6 FMT\_SAE.1 Time-limited authorization****13.8.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FMT\_SMR.1 (Security roles) ([13.10.6](#))  
FPT\_STM.1 (Reliable time stamps) ([15.14.6](#))

**13.8.6.2 Element FMT\_SAE.1.1**

**The TSF shall restrict the capability to specify an expiration time for [assignment (a<sup>1</sup>): *list of security attributes for which expiration is intended to be supported*] to [assignment (a<sup>2</sup>): *the authorized identified roles*].**

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should provide the list of security attributes for which expiration is intended to be supported.

In assignment (a<sup>2</sup>), an example of such an attribute can be a user's security clearance.

**13.8.6.3 Element FMT\_SAE.1.2**

**For each of these security attributes, the TSF shall be able to [assignment (a<sup>1</sup>): *list of actions to be taken for each security attribute*] after the expiration time for the indicated security attribute has passed.**

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should provide a list of actions to be taken for each security attribute when it expires. An example can be that the user's security clearance, when it expires, is set to the lowest allowable clearance on the TOE. If immediate revocation is desired by the PP, PP-Module, functional package or ST, the action "immediate revocation" should be specified.

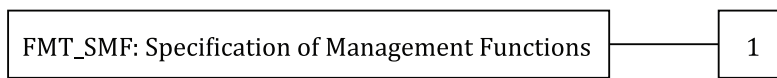
## 13.9 Specification of Management Functions (FMT\_SMF)

### 13.9.1 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) restricts the ability to use these management functions.

### 13.9.2 Component levelling and description

[Figure 54](#) illustrates the component levelling of family FMT\_SMF.



#### Key

1 component

**Figure 54 — FMT\_SMF: Component levelling**

FMT\_SMF.1 (Specification of Management Functions) ([13.9.6](#)) requires that the TSF provide specific management functions.

### 13.9.3 Component management

There are no management activities foreseen related to FMT\_SMF.1.

### 13.9.4 Component audit

The following action related to FMT\_SMF.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

— minimal: use of the management functions.

### 13.9.5 Application notes

This family allows the specification of the management functions to be provided by the TOE. Each security management function that is listed in fulfilling the assignment is either security attribute management, TSF data management, or security function management.

### 13.9.6 FMT\_SMF.1 Specification of Management Functions

#### 13.9.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 13.9.6.2 Component rationale

This component specifies the management functions to be provided.

## ISO/IEC 15408-2:2026(en)

PP, PP-Module, functional package or ST authors should consult the "Management" sections for components included in their PP, PP-Module, functional package or ST to provide a basis for the management functions to be listed via this component.

### 13.9.6.3 Element FMT\_SMF.1.1

**The TSF shall be capable of performing the following management functions: [assignment (a1): *list of management functions to be provided by the TSF*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the management functions to be provided by the TSF, either security attribute management, TSF data management, or security function management.

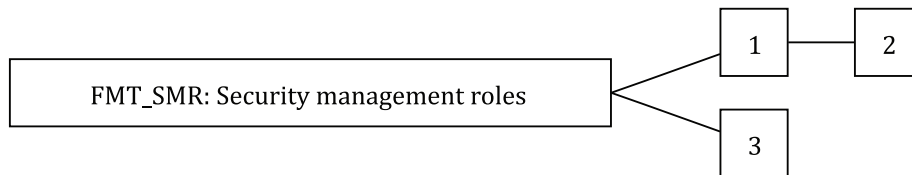
## 13.10 Security management roles (FMT\_SMR)

### 13.10.1 Family Behaviour

This family is intended to control the assignment of different roles to users. The capabilities of these roles with respect to security management are described in the other families in this class.

### 13.10.2 Component levelling and description

[Figure 55](#) illustrates the component levelling of family FMT\_SMR.



#### Key

1,2,3 components

**Figure 55 — FMT\_SMR: Component levelling**

FMT\_SMR.1 (Security roles) ([13.10.6](#)) specifies the roles with respect to security that the TSF recognizes.

FMT\_SMR.2 (Restrictions on security roles) ([13.10.7](#)) specifies that in addition to the specification of the roles, there are rules that control the relationship between the roles.

FMT\_SMR.3 (Assuming roles) ([13.10.8](#)) requires that an explicit request is given to the TSF to assume a role.

### 13.10.3 Component management

The following action related to FMT\_SMR.1 can be considered for the management functions in FMT (Security management):

- managing the group of users that are part of a role.

The following actions related to FMT\_SMR.2 can be considered for the management functions in FMT (Security management):

- managing the group of users that are part of a role;
- managing the conditions that the roles must satisfy.

There are no management activities foreseen related to FMT\_SMR.3.

**ISO/IEC 15408-2:2026(en)****13.10.4 Component audit**

The following actions related to FMT\_SMR.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: modifications to the group of users that are part of a role;
- detailed: every use of the rights of a role.

The following actions related to FMT\_SMR.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: modifications to the group of users that are part of a role;
- minimal: unsuccessful attempts to use a role due to the given conditions on the roles;
- detailed: every use of the rights of a role.

The following action related to FMT\_SMR.3 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: explicit request to assume a role.

**13.10.5 Application notes**

This family reduces the likelihood of damage resulting from users abusing their authority by taking actions outside their assigned functional responsibilities. It also addresses the threat that inadequate mechanisms have been provided to securely administer the TSF.

This family requires that information be maintained to identify whether a user is authorized to use a particular security-relevant administrative function.

Some management actions can be performed by users, while other management actions can only be performed by designated roles within the organization. This family allows the definition of different roles, such as owner, auditor, administrator, daily-management.

The roles as used in this family are security related roles. Each role can encompass an extensive set of capabilities or can be a single right. This family defines the roles. The capabilities of the role are defined in FMT\_LIM (Limited capabilities and availability), FMT\_MSA (Management of security attributes) and FMT\_MTD (Management of TSF data).

EXAMPLE 1 Set of capabilities: root in Unix-like operating systems.

Single right: right to read a single object such as the helpfile.

Some type of roles can be mutually exclusive.

EXAMPLE 2 The daily-management can define and activate users but is not able to remove users, which is reserved for the administrator (role).

This class will allow policies such as two-person control to be specified.

**13.10.6 FMT\_SMR.1 Security roles****13.10.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

## ISO/IEC 15408-2:2026(en)

### 13.10.6.2 Component rationale

This component specifies the different roles that the TSF should recognize. Often the system distinguishes between the owner of an entity, an administrator, and other users.

### 13.10.6.3 Element FMT\_SMR.1.1

**The TSF shall maintain the roles [assignment (a1): *the authorized identified roles*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the roles that are recognized by the system. These are the roles that users can occupy with respect to security.

### 13.10.6.4 Element FMT\_SMR.1.2

**The TSF shall be able to associate users with roles.**

### 13.10.7 FMT\_SMR.2 Restrictions on security roles

#### 13.10.7.1 Component relationships

Hierarchical to: FMT\_SMR.1 (Security roles) ([13.10.6](#))

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

#### 13.10.7.2 Component rationale

This component specifies the different roles that the TSF should recognize, and conditions on how those roles can be managed. Often the system distinguishes between the owner of an entity, an administrator, and other users.

The conditions on those roles specify the interrelationship between the different roles, as well as restrictions on when the role can be assumed by a user.

#### 13.10.7.3 Element FMT\_SMR.2.1

**The TSF shall maintain the roles: [assignment (a1): *the authorized identified roles*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the roles that are recognized by the system. These are the roles that users can occupy with respect to security.

#### 13.10.7.4 Element FMT\_SMR.2.2

**The TSF shall be able to associate users with roles.**

#### 13.10.7.5 Element FMT\_SMR.2.3

**The TSF shall ensure that the conditions [assignment (a1): *conditions for the different roles*] are satisfied.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the conditions that govern role assignment.

## ISO/IEC 15408-2:2026(en)

### 13.10.8 FMT\_SMR.3 Assuming roles

#### 13.10.8.1 Component relationships

Hierarchical to: No other components.

Dependencies: FMT\_SMR.1 (Security roles) ([13.10.6](#))

#### 13.10.8.2 Component rationale

This component specifies that an explicit request shall be given to assume the specific role.

#### 13.10.8.3 Element FMT\_SMR.3.1

**The TSF shall require an explicit request to assume the following roles: [assignment <sup>(a1)</sup>: *the roles*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the roles that require an explicit request to be assumed.

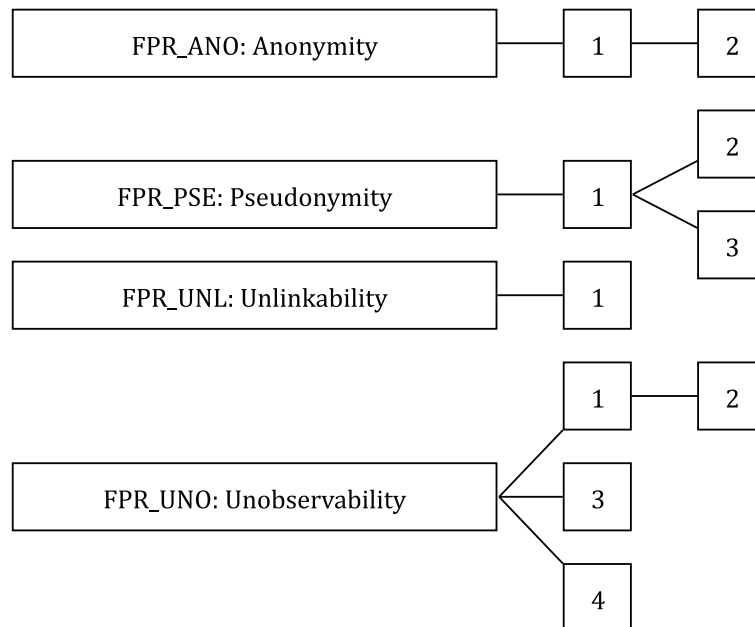
EXAMPLE Examples of roles are: owner, auditor, and administrator.

## 14 Class FPR Privacy

### 14.1 Introduction

This class contains privacy requirements. These requirements provide a user protection against discovery and misuse of identity by other users.

[Figure 56](#) illustrates the family hierarchy of class FPR.



#### Key

1,2,3,4 components

**Figure 56 — FPR: Privacy, class decomposition**

Direct, indirect and optional dependencies for the components of class FPR are shown in [Table 7](#).

**ISO/IEC 15408-2:2026(en)**

**Table 7 — Dependency table for class FPR: Privacy**

	FIA_UID.1 (12.8.6)	FPR_UNO.1 (14.6.6)
FPR_ANO.1 (14.3.6)		
FPR_ANO.2 (14.3.7)		
FPR_PSE.1 (14.4.6)		
FPR_PSE.2 (14.4.7)	X	
FPR_PSE.3 (14.4.8)		
FPR_UNL.1 (14.5.6)		
FPR_UNO.1 (14.6.6)		
FPR_UNO.2 (14.6.7)		
FPR_UNO.3 (14.6.8)		X
FPR_UNO.4 (14.6.9)		

**14.2 Notes on class FPR**

This class describes the requirements that can 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 1** A PP, PP-Module, functional package or ST author can 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 can impose limitations on the use of the components of other classes, such as FIA (Identification and authentication) or FAU (Security audit).

**EXAMPLE 2** 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 can still be possible to include audit requirements in a PP, PP-Module, functional package or 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 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 can 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. (e.g. 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 3** The TSF is not expected to screen the user name in electronic messages or databases.

**ISO/IEC 15408-2:2026(en)**

All families in this class have components that are specified through operations. These operations allow the author of a PP, PP-Module, functional package or ST to state the cooperating users/subjects to which the TSF must be resistant.

**EXAMPLE 4** An instantiation of anonymity can be, "The TSF 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.

**NOTE** See ISO/IEC/TS 19608,<sup>[6]</sup> which is based on the ISO/IEC 15408 series, for:

- selecting and specifying SFRs from this document to protect Personally Identifiable Information (PII);
- the procedure to define both privacy and SFRs in a coordinated manner;
- developing privacy functional requirements as extended components based on the privacy principles defined in ISO/IEC 29100<sup>[7]</sup> through the paradigm described in this document.

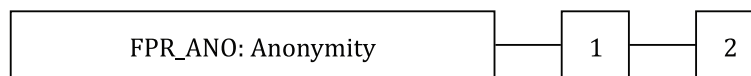
### 14.3 Anonymity (FPR\_ANO)

#### 14.3.1 Family Behaviour

This family ensures that a user can 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.

#### 14.3.2 Component levelling and description

[Figure 57](#) illustrates the component levelling of family FPR\_ANO.



**Key**

1,2 components

**Figure 57 — FPR\_ANO: Component levelling**

FPR\_ANO.1 (Anonymity) ([14.3.6](#)) 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) ([14.3.7](#)) enhances the requirements of FPR\_ANO.1 (Anonymity) ([14.3.6](#)) by ensuring that the TSF does not ask for the user identity.

#### 14.3.3 Component management

There are no management activities foreseen related to FPR\_ANO.1 or FPR\_ANO.2.

#### 14.3.4 Component audit

The following actions related to FPR\_ANO.1 and FPR\_ANO.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the invocation of the anonymity mechanism.

#### 14.3.5 Application notes

Anonymity ensures that a subject may use a resource or service without disclosing its user identity.

**ISO/IEC 15408-2:2026(en)**

The intention of this family is to specify that a user or subject can take action without releasing its user identity to others such as users, subjects, or objects. The family provides the author of a PP, PP-Module, functional package or ST with a means to identify the set of users that cannot see the identity of someone performing certain actions.

Therefore, if a subject, using anonymity, performs an action, another subject will not be able to determine either the identity or even a reference to the identity of the user employing the subject. The focus of the anonymity is on the protection of the user's identity, not on the protection of the subject identity; hence, the identity of the subject is not protected from disclosure.

Although the identity of the subject is not released to other subjects or users, the TSF is not explicitly prohibited from obtaining the users identity. In case the TSF is not allowed to know the identity of the user, FPR\_ANO.2 (Anonymity without soliciting information) (14.3.7) can be invoked. In that case, the TSF should not request the user information.

The interpretation of "determine" should be taken in the broadest sense of the word.

The component levelling and description distinguishes between the users and an authorized user. An authorized user is often excluded from the component, and therefore allowed to retrieve a user's identity. However, there is no specific requirement that an authorized user shall be able to determine the user's identity. For ultimate privacy, the components would be used to say that no user or authorized user can see the identity of anyone performing any action.

Although some systems will provide anonymity for all services that are provided, other systems provide anonymity for certain subjects/operations. To provide this flexibility, an operation is included where the scope of the requirement is defined. If the author of a PP, PP-Module, functional package or ST wants to address all subjects/operations, the words "all subjects and all operations" can be provided.

Possible applications include the ability to make enquiries of a confidential nature to public 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.

### 14.3.6 FPR\_ANO.1 Anonymity

#### 14.3.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 14.3.6.2 Component rationale

This component ensures that the identity of a user is protected from disclosure. There may be instances, however, that a given authorized user can determine who performed certain actions. This component gives the flexibility to capture either a limited or total privacy policy.

#### 14.3.6.3 Element FPR\_ANO.1.1

**The TSF shall ensure that [assignment (a1): *set of users and/or subjects*] are unable to determine the real user name bound to [assignment (a2): *list of subjects and/or operations and/or objects*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the set of users and/or subjects against which the TSF provides protection. For example, even if the author of a PP, PP-Module, functional package or ST specifies a single user or subject role, the TSF must not only provide protection against each individual user or subject but also protect with respect to cooperating users and/or subjects.

### **14.3.7 FPR\_ANO.2 Anonymity without soliciting information**

#### **14.3.7.1 Component relationships**

Hierarchical to: FPR\_ANO.1 (Anonymity) ([14.3.6](#))

Dependencies: No dependencies.

#### **14.3.7.2 Component rationale**

This component is used to ensure that the TSF is not allowed to know the identity of the user.

#### **14.3.7.3 Element FPR\_ANO.2.1**

The TSF shall ensure that [assignment <sup>(a1)</sup>: *set of users and/or subjects*] are unable to determine the real user name bound to [assignment <sup>(a2)</sup>: *list of subjects and/or operations and/or objects*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the set of users and/or subjects against which the TSF provides protection. For example, even if the author of a PP, PP-Module, functional package or ST specifies a single user or subject role, the TSF not only provides protection against each individual user or subject but protects with respect to cooperating users and/or subjects. A set of users, for example, can be a group of users which can operate under the same role or can all use the same process(es).

In assignment (a2), the author of a PP, PP-Module, functional package or ST 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 voting application.

#### **14.3.7.4 Element FPR\_ANO.2.2**

**The TSF shall provide [assignment <sup>(a1)</sup>: *list of services*] to [assignment <sup>(a2)</sup>: *list of subjects*] without soliciting any reference to the real user name.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify the list of services which are subject to the anonymity requirement, for example, "the accessing of job descriptions".

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of subjects from which the real user name of the subject should be protected when the specified services are provided.

## **14.4 Pseudonymity (FPR\_PSE)**

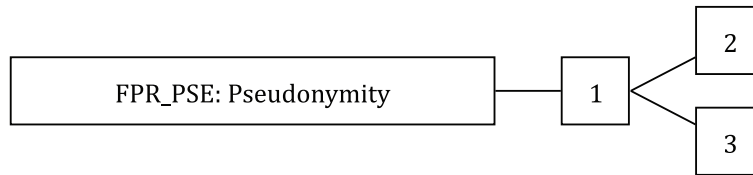
### **14.4.1 Family Behaviour**

This family ensures that a user may use a resource or service without disclosing its user identity but can still be accountable for that use.

### **14.4.2 Component levelling and description**

[Figure 58](#) illustrates the component levelling of family FPR\_PSE.

## ISO/IEC 15408-2:2026(en)



### Key

1,2,3 components

**Figure 58 — FPR\_PSE: Component levelling**

FPR\_PSE.1 (Pseudonymity) (14.4.6) requires that a set of users and/or subjects are unable to determine the identity of a user bound to a subject or operation, but that these users are still accountable for their actions.

FPR\_PSE.2 (Reversible pseudonymity) (14.4.7) requires the TSF to provide a capability to determine the original user identity based on a provided alias.

FPR\_PSE.3 (Alias pseudonymity) (14.4.8) requires the TSF to follow certain construction rules for the alias to the user identity.

### 14.4.3 Component management

There are no management activities foreseen related to FPR\_PSE.1, FPR\_PSE.2 or FPR\_PSE.3.

### 14.4.4 Component audit

The following action related to FPR\_PSE.1, FPR\_PSE.2 and FPR\_PSE.3 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

— minimal: the subject/user that requested resolution of the user identity should be audited.

### 14.4.5 Application notes

Pseudonymity ensures that a user may use a resource or service without disclosing its identity but can still be accountable for that use. The user can be accountable by directly being related to a reference (alias) held by the TSF, or by providing an alias that will be used for processing purposes, such as an account number.

In several respects, pseudonymity resembles anonymity. Both pseudonymity and anonymity protect the identity of the user, but in pseudonymity a reference to the user's identity is maintained for accountability or other purposes.

The component FPR\_PSE.1 (Pseudonymity) (14.4.6) does not specify the requirements on the reference to the user's identity. For the purpose of specifying requirements on this reference two sets of requirements are presented: FPR\_PSE.2 (Reversible pseudonymity) (14.4.7) and FPR\_PSE.3 (Alias pseudonymity) (14.4.8).

A way to use the reference is by being able to obtain the original user identity.

**EXAMPLE 1** 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).

In general, the user's identity can be retrieved under specific conditions. The author of a PP, PP-Module, functional package or ST can want to incorporate FPR\_PSE.2 (Reversible pseudonymity) (14.4.7) to describe those services.

Another usage of the reference is as an alias for a user.

**EXAMPLE 2** 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 (e.g. statistical operations on use of the system). In this case, the author of a PP, PP-Module, functional package or ST can wish to incorporate FPR\_PSE.3 (Alias pseudonymity) (14.4.8) to specify the rules to which the reference must conform.

**ISO/IEC 15408-2:2026(en)**

Using these constructs, digital money can be created using FPR\_PSE.2 (Reversible pseudonymity) (14.4.7) specifying that the user identity will be protected and, if specified in the condition, that there be a requirement to trace the user identity if the digital money is spent twice. When the user is honest, the user identity is protected; if the user tries to cheat, the user identity can be traced.

A different kind of system can be a digital credit card, where the user will provide a pseudonym that indicates an account from which the cash can be subtracted. In such cases, for example, FPR\_PSE.3 (Alias pseudonymity) (14.4.8) can be used. This component would specify that the user identity will be protected and, furthermore, that the same user will only get assigned values for which he/she has provided money (if so specified in the conditions).

It should be considered that the more stringent components potentially cannot be combined with other requirements, such as identification and authentication or audit. The interpretation of "determine the identity" should be taken in the broadest sense of the word. The information is not provided by the TSF during the operation, nor can the entity determine the subject or the owner of the subject that invoked the operation, nor will the TSF record information, available to the users or subjects, which can release the user identity in the future.

The intent is that the TSF not reveal any information that would compromise the identity of the user.

EXAMPLE 3 The identity of subjects acting on the user's behalf.

The information that is considered to be sensitive depends on the effort an attacker is capable of spending.

EXAMPLE 4 Possible applications include the ability to charge a caller for premium rate telephone services without disclosing their identity, or to be charged for the anonymous use of an electronic payment system. 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 users and the specific services that they use, and can misuse this information. Additionally, to anonymity services, pseudonymity services contain 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.

## 14.4.6 FPR\_PSE.1 Pseudonymity

### 14.4.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

### 14.4.6.2 Component rationale

This component provides the user protection against disclosure of identity to other users. Users remain accountable for their actions.

### 14.4.6.3 Element FPR\_PSE.1.1

**The TSF shall ensure that [assignment <sup>(a1)</sup>: *set of users and/or subjects*] are unable to determine the real user name bound to [assignment <sup>(a2)</sup>: *list of subjects and/or operations and/or objects*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the set of users and/or subjects against which the TSF provides protection. For example, even if the author of a PP, PP-Module, functional package or ST specifies a single user or subject role, the TSF not only provide protection against each individual user or subject but protect with respect to cooperating users and/or subjects. A set of users, for example, can be a group of users which can operate under the same role or can all use the same process(es).

**ISO/IEC 15408-2:2026(en)**

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of subjects and/or operations and/or objects where the real user name of the subject should be protected.

**EXAMPLE** An example of a scenario where the real user name of the subject is protected is when a user accesses job offers.

**14.4.6.4 Element FPR\_PSE.1.2**

**The TSF shall be able to provide [assignment <sup>(a1)</sup>: *number of aliases*] aliases of the real user name to [assignment <sup>(a2)</sup>: *list of subjects*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify the (one or more) number of aliases the TSF is able to provide.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of subjects to whom the TSF is able to provide an alias.

**NOTE** "Objects" includes any other attributes that can enable another user or subject to derive the actual identity of the user.

**14.4.6.5 Element FPR\_PSE.1.3**

**The TSF shall [selection, choose one of <sup>(s1)</sup>: *determine an alias for a user, accept the alias from the user*] and verify that it conforms to the [assignment <sup>(a1)</sup>: *alias metric*].**

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify whether the user alias is generated by the TSF or supplied by the user. Only one of these options may be chosen.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify the metric to which the TSF-generated or user-generated alias should conform.

**14.4.7 FPR\_PSE.2 Reversible pseudonymity****14.4.7.1 Component relationships**

Hierarchical to: FPR\_PSE.1 (Pseudonymity) ([14.4.6](#))

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

**14.4.7.2 Component rationale**

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) ([14.4.6](#)) 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".

**14.4.7.3 Element FPR\_PSE.2.1**

**The TSF shall ensure that [assignment <sup>(a1)</sup>: *set of users and/or subjects*] are unable to determine the real user name bound to [assignment <sup>(a2)</sup>: *list of subjects and/or operations and/or objects*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the set of users and/or subjects against which the TSF provides protection. For example, even if the author of a PP, PP-Module, functional package or ST 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/

**ISO/IEC 15408-2:2026(en)**

or subjects. A set of users, for example, can be a group of users which can operate under the same role or can all use the same process(es).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of subjects and/or operations and/or objects where the real user name of the subject should be protected.

**EXAMPLE** An example of a scenario where the real user name of the subject is protected is when a user accesses job offers.

**NOTE** "Objects" includes any other attributes that can enable another user or subject to derive the actual identity of the user.

**14.4.7.4 Element FPR\_PSE.2.2**

The TSF shall be able to provide [assignment (a1): *number of aliases*] aliases of the real user name to [assignment (a2): *list of subjects*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify the (one or more) number of aliases the TSF, is able to provide.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of subjects to whom the TSF is able to provide an alias.

**14.4.7.5 Element FPR\_PSE.2.3**

The TSF shall [selection, choose one of (s1): *determine an alias for a user, accept the alias from the user*] and verify that it conforms to the [assignment (a1): *alias metric*].

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify whether the user alias is generated by the TSF or supplied by the user. Only one of these options may be chosen.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify the metric to which the TSF-generated or user-generated alias should conform.

**14.4.7.6 Element FPR\_PSE.2.4**

**The TSF shall provide [selection (s1): *an authorized user, [assignment (a1): list of trusted subjects]] a capability to determine the user identity based on the provided alias only under the following [assignment (a2): list of conditions].***

In assignment (a1), the author of a PP, PP-Module, functional package or ST should select whether the authorized user and/or trusted subjects can determine the real user name.

In selection (s1), the author of a PP, PP-Module, functional package or ST should identify the list of conditions under which the trusted subjects and authorized user can determine the real user name based on the provided reference. These conditions can be conditions such as time of day, or they can be administrative such as on a court order.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of trusted subjects that can obtain the real user name under a specified condition.

**14.4.8 FPR\_PSE.3 Alias pseudonymity****14.4.8.1 Component relationships**

Hierarchical to: FPR\_PSE.1 (Pseudonymity) ([14.4.6](#))

Dependencies: No dependencies.

**ISO/IEC 15408-2:2026(en)****14.4.8.2 Component rationale**

In this component, the TSF ensures that the provided reference meets certain construction rules, and thereby can be used in a secure way by potentially insecure subjects.

If a user wants to use disk resources without disclosing its identity, pseudonymity can be used. However, every time the user accesses the system, the same alias shall be used. Such conditions may be specified in this component.

**14.4.8.3 Element FPR\_PSE.3.1**

The TSF shall ensure that [assignment <sup>(a1)</sup>: *set of users and/or subjects*] are unable to determine the real user name bound to [assignment <sup>(a2)</sup>: *list of subjects and/or operations and/or objects*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the set of users and/or subjects against which the TSF provides protection. For example, even if the author of a PP, PP-Module, functional package or ST 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, can be a group of users which can operate under the same role or can all use the same process(es).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of subjects and/or operations and/or objects where the real user name of the subject should be protected.

**EXAMPLE** An example of a scenario where the real user name of the subject is protected is when a user accesses job offers.

**NOTE** "Objects" includes any other attributes which can enable another user or subject to derive the actual identity of the user.

**14.4.8.4 Element FPR\_PSE.3.2**

The TSF shall be able to provide [assignment <sup>(a1)</sup>: *number of aliases*] aliases of the real user name to [assignment <sup>(a2)</sup>: *list of subjects*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify the (one or more) number of aliases the TSF is able to provide.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of subjects to whom the TSF is able to provide an alias.

**14.4.8.5 Element FPR\_PSE.3.3**

The TSF shall [selection, choose one of <sup>(s1)</sup>: *determine an alias for a user, accept the alias from the user*] and verify that it conforms to the [assignment <sup>(a1)</sup>: *alias metric*].

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify whether the user alias is generated by the TSF, or supplied by the user. Only one of these options may be chosen.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify the metric to which the TSF-generated or user-generated alias should conform.

**14.4.8.6 Element FPR\_PSE.3.4**

**The TSF shall provide an alias to the real user name which shall be identical to an alias provided previously under the following [assignment <sup>(a1)</sup>: *list of conditions*] otherwise the alias provided shall be unrelated to previously provided aliases.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify the list of conditions that indicate when the used reference for the real user name shall be identical and when it shall be different, for example, "when the user logs on to the same host" it will use a unique alias.

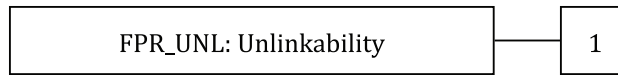
## 14.5 Unlinkability (FPR\_UNL)

### 14.5.1 Family Behaviour

This family ensures that selected entities can be linked together without external entities being able to back trace these links.

### 14.5.2 Component levelling and description

[Figure 59](#) illustrates the component levelling of family FPR\_UNL.



#### Key

1 component

**Figure 59 — FPR\_UNL: Component levelling**

FPR\_UNL.1 (Unlinkability of operations) ([14.5.6](#)) requires that users and/or subjects are unable to determine whether the same user caused certain specific operations in the system, or whether operations are related in some other manner. This component ensures that users cannot link different operations in the system and thereby obtain information.

### 14.5.3 Component management

The following action related to FPR\_UNL.1 can be considered for the management functions in FMT (Security management):

- the management of the unlinkability function.

### 14.5.4 Component audit

The following action related to FPR\_UNL.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the invocation of the unlinkability mechanism.

### 14.5.5 Application notes

Unlinkability ensures that a user may make multiple uses of resources or services without others being able to link these uses together. Unlinkability differs from pseudonymity that, although in pseudonymity the user is also not known, relations between different actions can be provided.

The requirements for unlinkability are intended to protect the user identity against the use of profiling of the operations.

**EXAMPLE 1** 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.

Hiding the relationship between different invocations of a service or access of a resource will prevent this kind of information gathering.

As a result, a requirement for unlinkability can imply that the subject and user identity of an operation must be protected. Otherwise, this information can be used to link operations together.

Unlinkability requires that different operations cannot be related. This relationship can take several forms.

**EXAMPLE 2** The user associated with the operation, or the terminal which initiated the action, or the time the action was executed.

**ISO/IEC 15408-2:2026(en)**

The author of a PP, PP-Module, functional package or ST can specify what kind of relationships are present that must be countered.

Possible applications include the ability to make multiple use of a pseudonym without creating a usage pattern that can disclose the user's identity.

**EXAMPLE 3** 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.

Unlinkability protects users from linkages, which can be drawn between several actions of a customer.

**EXAMPLE 4** A series of phone calls made by an anonymous customer to different partners, where the combination of the partner's identities can disclose the identity of the customer.

## 14.5.6 FPR\_UNL.1 Unlinkability of operations

### 14.5.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

### 14.5.6.2 Component rationale

This component ensures that users cannot link different operations in the system and thereby obtain information.

### 14.5.6.3 Element FPR\_UNL.1.1

**The TSF shall ensure that [assignment <sup>(a1)</sup>: *set of entities and/or operations*] are unable to determine whether [assignment <sup>(a2)</sup>: *list of entities and/or operations*][selection <sup>(s1)</sup>: *were caused by the same user, are related as follows [assignment <sup>(a3)</sup>: *list of relations*]].***

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the set of users and/or subjects against which the TSF provides protection. For example, even if the author of a PP, PP-Module, functional package or ST 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, can be a group of users which can operate under the same role or can all use the same process(es).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of operations which should be subjected to the unlinkability requirement.

**EXAMPLE 1** "Sending email".

In assignment (a3), the author of a PP, PP-Module, functional package or ST should identify the list of relations which should be protected against.

**EXAMPLE 2** "Originate from the same IP address".

**NOTE** This SFR does not only stipulate at the individual set of operations performed by one entity. This SFR intends to look at a chain of interlinked operations by multiple entities. This chain can be subsumed as a transaction.

In selection (s1), the author of a PP, PP-Module, functional package or ST should select the relationships that should be obscured. The selection allows either the user identity or an assignment of relations to be specified.

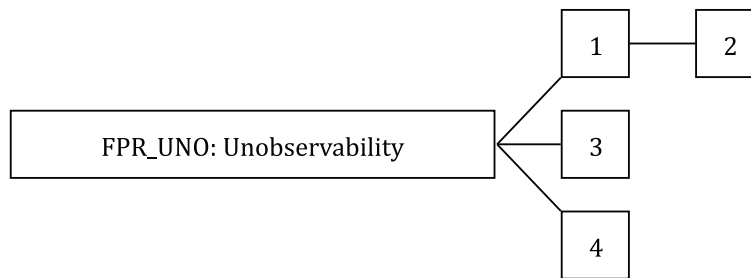
## 14.6 Unobservability (FPR\_UNO)

### 14.6.1 Family Behaviour

This family ensures that a user can use a resource or service without others, especially third parties, being able to observe that the resource or service is being used.

### 14.6.2 Component levelling and description

[Figure 60](#) illustrates the component levelling of family FPR\_UNO.



#### Key

1,2,3,4 components

**Figure 60 — FPR\_UNO: Component levelling**

FPR\_UNO.1 (Unobservability) ([14.6.6](#)) requires that users and/or subjects cannot determine whether an operation is being performed.

FPR\_UNO.2 (Allocation of information impacting unobservability) ([14.6.7](#)) requires that the TSF provide specific mechanisms to avoid the concentration of privacy related information within the TOE. Such concentrations can impact unobservability if a security compromise occurs.

FPR\_UNO.3 (Unobservability without soliciting information) ([14.6.8](#)) requires that the TSF does not try to obtain privacy related information that can be used to compromise unobservability.

FPR\_UNO.4 (Authorized user observability) ([14.6.9](#)) requires the TSF to provide one or more authorized users with a capability to observe the usage of resources and/or services.

### 14.6.3 Component management

The following action related to FPR\_UNO.1 and FPR\_UNO.2 can be considered for the management functions in FMT (Security management):

- the management of the behaviour of the unobservability function.

The following action related to FPR\_UNO.4 can be considered for the management functions in FMT (Security management):

- the list of authorized users that are capable of determining the occurrence of operations.

There are no management activities foreseen related to FPR\_UNO.3.

### 14.6.4 Component audit

The following action related to FPR\_UNO.1 and FPR\_UNO.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the invocation of the unobservability mechanism.

**ISO/IEC 15408-2:2026(en)**

The following action related to FPR\_UNO.4 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the observation of the use of a resource or service by a user or subject.

There are no auditable events foreseen related to FPR\_UNO.3.

**14.6.5 Application notes**

Unobservability ensures that a user may use a resource or service without others, especially third parties, being able to observe that the resource or service is being used.

Unobservability approaches the user identity from a different direction than the previous families Anonymity, Pseudonymity and Unlinkability. In this case, the intent is to hide the use of a resource or service, rather than to hide the user's identity.

A number of techniques can be applied to implement unobservability.

EXAMPLE      Examples of techniques to provide unobservability are:

- 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 can 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 can 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) ([14.6.7](#));
- 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 is intended to reach receivers who fear a stigma for being interested in that information (such as sensitive medical information);
- cryptographic protection and message padding: People observing a message stream can 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.

Sometimes, users should not see the use of a resource, but an authorized user must be allowed to see the use of the resource in order to perform their duties. In such cases, the FPR\_UNO.4 (Authorized user observability) ([14.6.9](#)) may be used, which provides the capability for one or more authorized users to see the usage.

This family makes use of the concept "parts of the TOE". This is considered any part of the TOE that is either physically or logically separated from other parts of the TOE.

Unobservability of communications may be an important factor in many areas, such as the enforcement of constitutional rights, organizational policies, or in defence related applications.

**14.6.6 FPR\_UNO.1 Unobservability****14.6.6.1 Component relationships**

Hierarchical to:      No other components.

Dependencies:      No dependencies.

**14.6.6.2 Component rationale**

This component requires that the use of a function or resource cannot be observed by unauthorized users.

**ISO/IEC 15408-2:2026(en)****14.6.6.3 Element FPR\_UNO.1.1**

The TSF shall ensure that [assignment <sup>(a1)</sup>: *list of users and/or subjects*] are unable to observe the operation [assignment <sup>(a2)</sup>: *list of operations*] on [assignment <sup>(a3)</sup>: *list of objects*] by [assignment <sup>(a4)</sup>: *list of protected users and/or subjects*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of users and/or subjects against which the TSF provides protection. For example, even if the author of a PP, PP-Module, functional package or ST 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, can be a group of users which can operate under the same role or can all use the same process(es).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of operations that are subjected to the unobservability requirement. Other users/subjects will then not be able to observe the operations on a covered object in the specified list (e.g. reading and writing to the object).

In assignment (a3), the author of a PP, PP-Module, functional package or ST should identify the list of objects which are covered by the unobservability requirement.

EXAMPLE 1 A specific mail server or ftp site.

In assignment (a4), the author of a PP, PP-Module, functional package or ST should specify the set of protected users and/or subjects whose unobservability information will be protected.

EXAMPLE 2 "Users accessing the system through the internet".

**14.6.7 FPR\_UNO.2 Allocation of information impacting unobservability****14.6.7.1 Component relationships**

Hierarchical to: FPR\_UNO.1 (Unobservability) ([14.6.6](#))

Dependencies: No dependencies.

**14.6.7.2 Component rationale**

This component requires that the use of a function or resource cannot be observed by specified users or subjects. Furthermore, this component specifies that information related to the privacy of the user is distributed within the TOE such that attackers can not know which part of the TOE to target, or they need to attack multiple parts of the TOE.

An example of the use of this component is the use of a randomly allocated node to provide a function. In such a case the component can require that the privacy related information shall only be available to one identified part of the TOE and will not be communicated outside this part of the TOE.

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).

**14.6.7.3 Element FPR\_UNO.2.1**

The TSF shall ensure that [assignment <sup>(a1)</sup>: *list of users and/or subjects*] are unable to observe the operation [assignment <sup>(a2)</sup>: *list of operations*] on [assignment <sup>(a3)</sup>: *list of objects*] by [assignment <sup>(a4)</sup>: *list of protected users and/or subjects*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of users and/or subjects against which the TSF provides protection. For example, even if the author of a PP, PP-Module,

## ISO/IEC 15408-2:2026(en)

functional package or ST 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 1 A set of users can be a group of users which can operate under the same role or can all use the same process(es).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of operations that are subjected to the unobservability requirement. Other users/subjects will then not be able to observe the operations on a covered object in the specified list (e.g. reading and writing to the object).

In assignment (a3), the author of a PP, PP-Module, functional package or ST should identify the list of objects which are covered by the unobservability requirement.

EXAMPLE 2 An example can be a specific mail server or ftp site.

In assignment (a4), the author of a PP, PP-Module, functional package or ST should specify the set of protected users and/or subjects whose unobservability information will be protected.

EXAMPLE 3 "Users accessing the system through the internet".

### 14.6.7.4 Element FPR\_UNO.2.2

**The TSF shall allocate the [assignment (a1): *unobservability related information*] among different parts of the TOE such that the following conditions hold during the lifetime of the information: [assignment (a2): *list of conditions*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify which privacy related information should be distributed in a controlled manner.

EXAMPLE 1 This information can include: IP address of subject, IP address of object, time, used encryption keys.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the conditions to which the dissemination of the information should adhere. These conditions should be maintained throughout the lifetime of the privacy related information of each instance.

EXAMPLE 2 Examples of these conditions can 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".

### 14.6.8 FPR\_UNO.3 Unobservability without soliciting information

#### 14.6.8.1 Component relationships

Hierarchical to: No other components.

Dependencies: FPR\_UNO.1 (Unobservability) ([14.6.6](#))

#### 14.6.8.2 Component rationale

This component ensures that the TSF does not try to obtain information that can compromise unobservability when provided specific services. Therefore, the TSF will not solicit (i.e. try to obtain from other entities) any information that can be used to compromise unobservability.

**ISO/IEC 15408-2:2026(en)****14.6.8.3 Element FPR\_UNO.3.1**

**The TSF shall provide [assignment (a1): *list of services*] to [assignment (a2): *list of subjects*] without soliciting any reference to [assignment (a3): *privacy related information*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should identify the list of services which are subject to the unobservability requirement.

EXAMPLE 1 "The accessing of job descriptions".

In assignment (a2), the author of a PP, PP-Module, functional package or ST should identify the list of subjects from which privacy related information should be protected when the specified services are provided.

In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify the privacy related information that will be protected from the specified subjects.

EXAMPLE 2 Examples of privacy related information 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.

**14.6.9 FPR\_UNO.4 Authorized user observability****14.6.9.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**14.6.9.2 Component rationale**

This component ensures that there will be one or more authorized users with the rights to view the resource utilization. Without this component, this review is allowed, but not mandated.

**14.6.9.3 Element FPR\_UNO.4.1**

**The TSF shall provide [assignment (a1): *set of authorized users*] with the capability to observe the usage of [assignment (a2): *list of resources and/or services*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the set of authorized users for which the TSF provides the capability to observe the resource utilization.

EXAMPLE A set of authorized users, for example, can be a group of authorized users which can operate under the same role or can all use the same process(es).

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the set of resources and/or services that the authorized user must be able to observe.

**15 Class FPT Protection of the TSF****15.1 Introduction**

This class contains families of functional requirements that relate to:

- the integrity and management of the mechanisms that constitute the TSF;
- the integrity of TSF data.

Families in this class appear to duplicate the components in the FDP (User data protection) class, and they can be implemented using the same mechanisms. While FDP (User data protection) focuses on user data protection, FPT (Protection of the TSF) focuses on TSF data protection. Components from the FPT

## **ISO/IEC 15408-2:2026(en)**

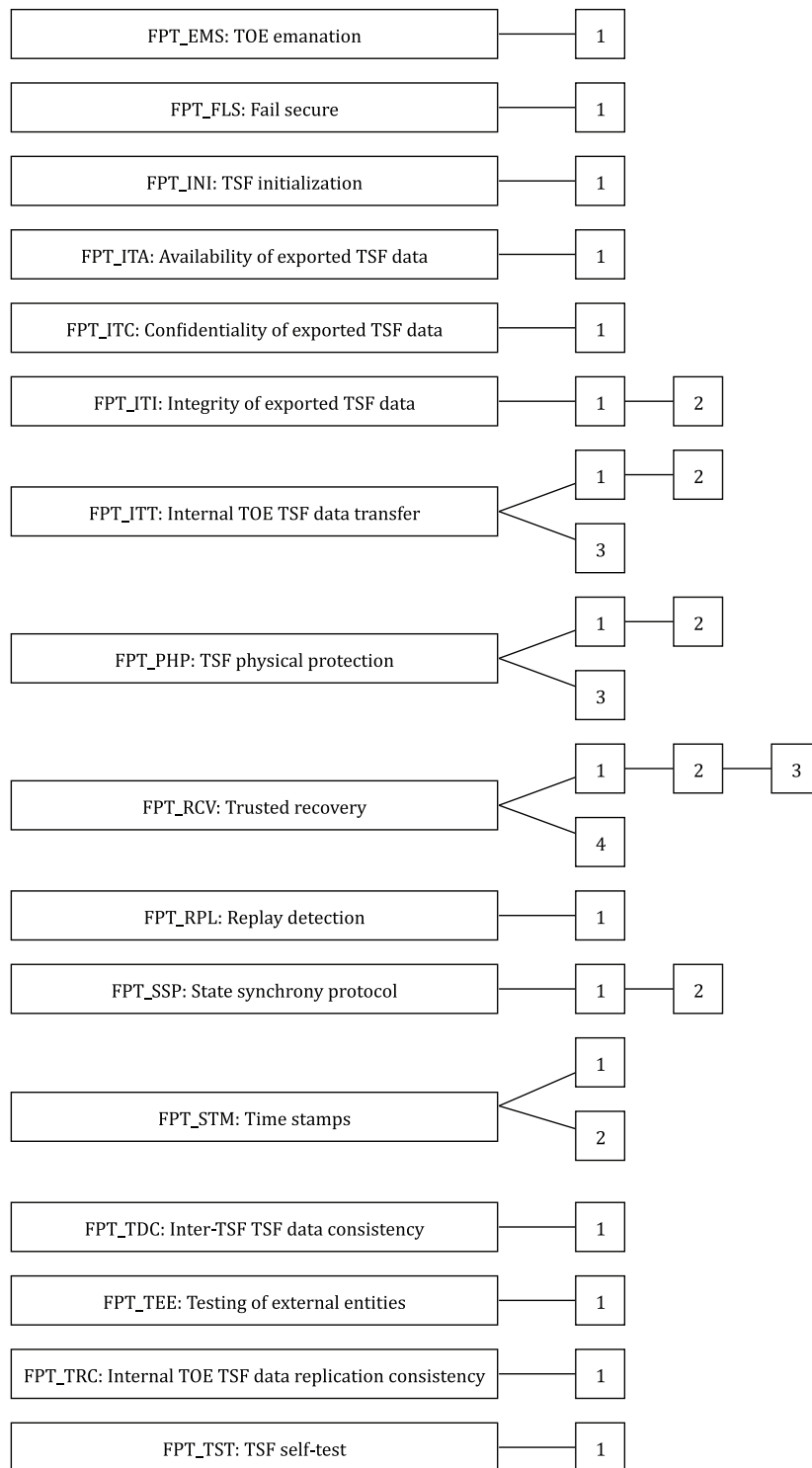
(Protection of the TSF) class are necessary to provide requirements that the SFPs in the TOE cannot be tampered with or bypassed.

This class addresses three significant elements related to the TSF, as follows:

- the TSF's implementation, which executes and implements the mechanisms that enforce the SFRs;
- the TSF's data, which are the administrative databases that guide the enforcement of the SFRs;
- the external entities that the TSF may interact with in order to enforce the SFRs.

[Figure 61](#) illustrates the family hierarchy of class FPT.

**ISO/IEC 15408-2:2026(en)**



**Key**  
 1,2,3,4 components

**Figure 61 — FPT: Protection of the TSF, class decomposition**

Direct, indirect and optional dependencies for the components of class FPT are shown in [Table 8](#).

**ISO/IEC 15408-2:2026(en)**

**Table 8 — Dependency table for class FPT: Protection of the TSF**

	AGD_OPE.1	FMT_MOF.1 (13.4.6)	FMT_SMR.1 (13.10.6)	FPT_ITT.1 (15.9.7)	FPT_STM.1 (15.14.6)
FPT_EMS.1 (15.3.6)					
FPT_FLS.1 (15.4.6)					
FPT_INI.1 (15.5.6)					
FPT_ITA.1 (15.6.6)					
FPT_ITC.1 (15.7.7)					
FPT_ITI.1 (15.8.7)					
FPT_ITI.2 (15.8.8)					
FPT_ITT.1 (15.9.7)					
FPT_ITT.2 (15.9.8)					
FPT_ITT.3 (15.9.9)				X	
FPT_PHP.1 (15.10.6)					
FPT_PHP.2 (15.10.7)		X			
FPT_PHP.3 (15.10.8)					
FPT_RCV.1 (15.11.7)	X				
FPT_RCV.2 (15.11.8)	X				
FPT_RCV.3 (15.11.9)	X				
FPT_RCV.4 (15.11.10)					
FPT_RPL.1 (15.12.6)					
FPT_SSP.1 (15.13.6)				X	
FPT_SSP.2 (15.13.7)				X	
FPT_STM.1 (15.14.6)					
FPT_STM.2 (15.14.7)			X		X
FPT_TDC.1 (15.15.6)					
FPT_TEE.1 (15.16.7)					
FPT_TRC.1 (15.17.6)				X	
FPT_TST.1 (15.18.7)					

**15.2 Notes on class FPT**

All of the families in the FPT (Protection of the TSF) class can be related to these areas, and fall into the following groupings:

- FPT\_EMS (TOE emanation) (15.3), which addresses potential leakage of information from the TOE via emanations;
- FPT\_RCV (Trusted recovery) (15.11), FPT\_FLS (Fail secure) (15.4), and FPT\_TRC (Internal TOE TSF data replication consistency) (15.17), which address the behaviour of the TSF when failure occurs and immediately after;
- FPT\_INI (TSF initialization) (15.5), which addresses the initialization of the TSF into a correct and secure operational state;
- FPT\_ITT (Internal TOE TSF data transfer) (15.9), which addresses protection of TSF data when it is transmitted between physically-separated parts of the TOE;

## ISO/IEC 15408-2:2026(en)

- FPT\_PHP (TSF physical protection) (15.10), which provides an authorized user with the ability to detect external attacks on the parts of the TOE that comprise the TSF;
- FPT\_ITA (Availability of exported TSF data) (15.6), FPT\_ITC (Confidentiality of exported TSF data) (15.7), FPT\_ITI (Integrity of exported TSF data) (15.8), which address the protection and availability of TSF data between the TSF and another trusted IT product;
- FPT\_RPL (Replay detection) (15.12), which addresses the replay of various types of information and/or operations;
- FPT\_SSP (State synchrony protocol) (15.13), which addresses the synchronization of states, based upon TSF data, between different parts of a distributed TSF;
- FPT\_STM (Time stamps) (15.14), which addresses reliable timing;
- FPT\_TDC (Inter-TSF TSF data consistency) (15.15), which addresses the consistency of TSF data shared between the TSF and another trusted IT product;
- FPT\_TEE (Testing of external entities) (15.16) and FPT\_TST (TSF self-test) (15.18), 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.

### 15.3 TOE emanation (FPT\_EMS)

#### 15.3.1 Family Behaviour

This family describes requirements related to the leakage of information based on emanation.

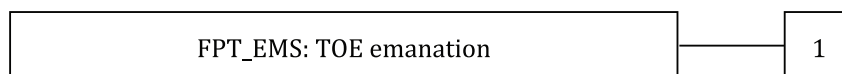
If the TOE must prevent attacks against the TOE and secret data processed by the TOE where the attack is based on external observable phenomena of the TOE during its operation, different types of emissions and interfaces of the TOE as well as different types of TSF data and user data can be addressed.

**EXAMPLE** Examples of such attacks against the TOE and its processed secret data are simple power analysis (SPA), differential power analysis (DPA), simple electromagnetic analysis (SEMA), differential electromagnetic analysis (DEMA), timing attacks, padding oracle attacks, cache miss attacks.

This family describes the functional requirements for the limitation of intelligible emanations which are not directly addressed by any other component of this document.

#### 15.3.2 Component levelling and description

Figure 62 illustrates the component levelling of family FPT\_EMS.



#### Key

1 component

**Figure 62 — FPT\_EMS: Component levelling**

This family consists of one component, FPT\_EMS.1 (Emanation of TSF and User data) (15.3.6), which defines requirements for the TOE to mitigate intelligible emanations.

#### 15.3.3 Component management

There are no management activities foreseen related to FPT\_EMS.1.

### 15.3.4 Component audit

There are no auditable events foreseen related to FPT\_EMS.1.

### 15.3.5 Application notes

This family defines the requirements for the TOE to be able to prevent or mitigate attacks against data stored in and used by the TOE where the attack is based on external observable physical phenomena of the TOE.

EXAMPLE Examples of such attacks are analysis of TOE's electromagnetic radiation, simple power analysis (SPA), differential power analysis (DPA), timing attacks.

FPT\_EMS.1 (Emanation of TSF and User data) (15.3.6) requires the TOE to not emit intelligible emissions enabling access to TSF data or user data.

### 15.3.6 FPT\_EMS.1 Emanation of TSF and User data

#### 15.3.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 15.3.6.2 Component rationale

Specifying this component requires a relational representation of any combination of TSF data and/or user data in relation to any emission combined with the attack surface. Data, emissions and attack surfaces may be typified.

#### 15.3.6.3 Element FPT\_EMS.1.1

**The TSF shall ensure that the TOE does not emit emissions over its attack surface in such amount that these emissions enable access to TSF data and user data as specified in the following table:**

**Table 9 — Limit of Emissions**

ID	Emissions	Attack surface	TSF data	User data
1	[assignment <sup>(a1)</sup> : list of types of emissions]	[assignment <sup>(a2)</sup> : list of types of attack surface]	[assignment <sup>(a3)</sup> : list of types of TSF data]	[assignment <sup>(a4)</sup> : list of types of user data]
...	...	...	...	...

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify for each assignment the list of types of emissions that have been treated, the list of types of attack surfaces under consideration for attacks based on emissions, the list of types of TSF data protected by emissions treatment, and the list of types of user data protected by emissions treatment. This should be done in table form, similar to [Table 9](#), whereby each table row specifies a specific combination of those lists of types of emissions, attack surface, TSF data and user data. Hereby, it is not expected that an author enters all types of emissions and types of attack surface (etc.) in one row.

The table should be completed by the author of a PP, PP-Module, functional package or ST. Each row, which can be identified using the "Identifier", provides a set of assignments for completing the SFR, allowing the author of a PP, PP-Module, functional package or ST to specify the requirements for TOE emanation protection for various different combinations of emissions, interfaces, TSF data and user data.

EXAMPLE Emission and attack surface can be of physical or logical type.

— Types of emissions can include audio frequencies, radio frequencies, information on power consumption, electromagnetic radiation, and timing information.

## ISO/IEC 15408-2:2026(en)

- Types of attack surface can include TOE interfaces, physical ports, IC boundaries, electronic components, and logical access.

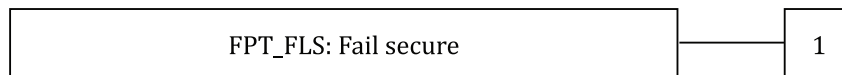
### 15.4 Fail secure (FPT\_FLS)

#### 15.4.1 Family Behaviour

The requirements of this family ensure that the TOE will always enforce its SFRs in the event of identified categories of failures in the TSF.

#### 15.4.2 Component levelling and description

[Figure 63](#) illustrates the component levelling of family FPT\_FLS.



#### Key

1 component

**Figure 63 — FPT\_FLS: Component levelling**

This family consists of only one component, FPT\_FLS.1 (Failure with preservation of secure state) ([15.4.6](#)), which requires that the TSF preserve a secure state in the face of the identified failures.

#### 15.4.3 Component management

There are no management activities foreseen related to FPT\_FLS.1.

#### 15.4.4 Component audit

The following actions related to FPT\_FLS.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- basic: failure of the TSF.

#### 15.4.5 Application notes

The requirements of this family ensure that the TOE will always enforce its SFRs in the event of certain types of failures in the TSF.

#### 15.4.6 FPT\_FLS.1 Failure with preservation of secure state

##### 15.4.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

##### 15.4.6.2 Component rationale

The term "secure state" refers to a state in which the TSF data are consistent and the TSF continues correct enforcement of the SFRs.

Although it is desirable to audit situations in which failure with preservation of secure state occurs, it is not possible in all situations. The author of a PP, PP-Module, functional package or ST should specify those situations in which audit is desired and feasible.

## ISO/IEC 15408-2:2026(en)

Failures in the TSF may include "hard" failures, which indicate an equipment malfunction and which may require maintenance, service, or repair of the TSF. Failures in the TSF may also include recoverable "soft" failures, which may only require initialization or resetting of the TSF.

### 15.4.6.3 Element FPT\_FLS.1.1

**The TSF shall preserve a secure state when the following types of failures occur: [assignment <sup>(a1)</sup>: list of types of failures in the TSF].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should list the types of failures in the TSF for which the TSF should "fail secure," that is, should preserve a secure state and continue to correctly enforce the SFRs.

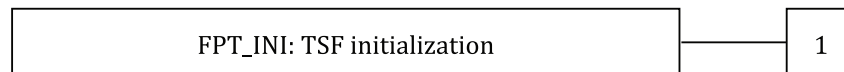
## 15.5 TSF initialization (FPT\_INI)

### 15.5.1 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.

### 15.5.2 Component levelling and description

[Figure 64](#) illustrates the component levelling of family FPT\_INI.



#### Key

1 component

**Figure 64 — FPT\_INI: Component levelling**

This family consists of only one component, FPT\_INI.1 (TSF initialization) ([15.5.6](#)). This component requires the TOE to provide a TSF initialization function that brings the TSF into a secure operational state at power-on. The TOE components related to TSF initialisation are considered themselves part of the TSF, and analysed from that perspective.

### 15.5.3 Component management

There are no management activities foreseen related to FPT\_INI.1.

### 15.5.4 Component audit

There are no auditable events foreseen related to FPT\_INI.1.

### 15.5.5 Application notes

This family defines the functional requirements for the initialization of the TSF. A dedicated function of the TOE ensures that the initialization of the TSF results in a correct and secure operational state. This can cover code/data that are stored and executed from non-modifiable memory at boot time, the immutable root-of-trust, and other one-time programmable (OTP) values such as versions and identifiers.

**ISO/IEC 15408-2:2026(en)****15.5.6 FPT\_INI.1 TSF initialization****15.5.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**15.5.6.2 Component rationale**

This component covers for instance code/data stored and executed from non-modifiable memory at boot time, the immutable root-of-trust, and other OTP values such as versions and identifiers.

**15.5.6.3 Element FPT\_INI.1.1**

The TOE shall provide a TSF initialization function which is self-protected for integrity and authenticity.

**15.5.6.4 Element FPT\_INI.1.2**

The TSF initialization function shall ensure that certain properties hold on certain elements immediately before establishing the TSF in a secure initial state, as specified in the following table:

**Table 10 — Initialization**

ID	Properties	Elements
1	[assignment <sup>(a1)</sup> : <i>property</i> ]	[assignment <sup>(a2)</sup> : <i>list of TSF/user firmware, software or data</i> ]
...	...	...

In assignment (a1), and (a2), the author of a PP, PP-Module, functional package or ST should list the properties and the elements to which they apply, respectively. This should be done in table form, similar to [Table 10](#).

EXAMPLE Properties can include authenticity, integrity, correct version and elements to which the properties apply can include TSF or user firmware, software or data.

It is not expected that an author enters all the properties and elements in one row.

**15.5.6.5 Element FPT\_INI.1.3**

The TSF initialization function shall detect and respond to errors and failures during initialization such that the TOE [selection <sup>(s1)</sup>: *is halted, successfully completes initialization with [selection <sup>(s2)</sup>: reduced functionality, signalling error state, [assignment <sup>(a1)</sup>: *list of actions*]]].*

In selection (s2), (s2), and (a1), the author of a PP, PP-Module, functional package or ST uses the selections and assignments to describe the behaviour of the TSF initialization function in the case that errors or other failures are encountered during the initialization.

**15.5.6.6 Element FPT\_INI.1.4**

The TSF initialization function shall only interact with the TSF in [assignment <sup>(a1)</sup>: *defined methods*] during initialization.

In assignment (a1), the author of a PP, PP-Module, functional package or ST uses the assignment to describe the methods by which the TSF initialization function interacts with the TSF.

## ISO/IEC 15408-2:2026(en)

### 15.6 Availability of exported TSF data (FPT\_ITA)

#### 15.6.1 Family Behaviour

This family defines the rules for the prevention of loss of availability of TSF data moving between the TSF and another trusted IT product.

#### 15.6.2 Component levelling and description

[Figure 65](#) illustrates the component levelling of family FPT\_ITA.



#### Key

1 component

**Figure 65 — FPT\_ITA: Component levelling**

This family consists of only one component, FPT\_ITA.1 (Inter-TSF availability within a defined availability metric) ([15.6.6](#)). This component requires that the TSF ensure, to an identified degree of probability, the availability of TSF data provided to another trusted IT product.

#### 15.6.3 Component management

The following actions related to FPT\_ITA.1 can be considered for the management functions in FMT (Security management):

- management of the list of types of TSF data that be available to another trusted it product.

#### 15.6.4 Component audit

The following actions related to FPT\_ITA.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the absence of TSF data when required by a TOE.

#### 15.6.5 Application notes

This family defines the rules for the prevention of loss of availability of TSF data moving between the TSF and another trusted IT product. This data can be TSF critical data such as passwords, keys, audit data, or TSF executable code.

This family is used in a distributed context where the TSF is providing TSF data to another trusted IT product. The TSF can only take the measures at its site and cannot be held responsible for the TSF at the other trusted IT product.

If there are different availability metrics for different types of TSF data, then this component should be iterated for each unique pairing of metrics and types of TSF data.

#### 15.6.6 FPT\_ITA.1 Inter-TSF availability within a defined availability metric

##### 15.6.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

**ISO/IEC 15408-2:2026(en)****15.6.6.2 Element FPT\_ITA.1.1**

The TSF shall ensure the availability of [assignment <sup>(a1)</sup>: *list of types of TSF data*] provided to another trusted IT product within [assignment <sup>(a2)</sup>: *a defined availability metric*] given the following conditions [assignment <sup>(a3)</sup>: *conditions to ensure availability*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the types of TSF data that are subject to the availability metric.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the availability metric for the applicable TSF data.

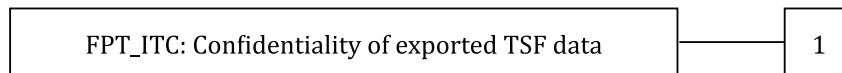
In assignment (a3), the author of a PP, PP-Module, functional package or ST should specify the conditions under which availability must be ensured.

**15.7 Confidentiality of exported TSF data (FPT\_ITC)****15.7.1 Family Behaviour**

This family defines the rules for the protection from unauthorized disclosure of TSF data during transmission between the TSF and another trusted IT product.

**15.7.2 Component levelling and description**

[Figure 66](#) illustrates the component levelling of family FPT\_ITC.

**Key**

1 component

**Figure 66 — FPT\_ITC: Component levelling**

This family consists of only one component, FPT\_ITC.1 (Inter-TSF confidentiality during transmission) ([15.7.7](#)), which requires that the TSF ensure that data transmitted between the TSF and another trusted IT product is protected from disclosure while in transit.

**15.7.3 Component management**

There are no management activities foreseen related to FPT\_ITC.1.

**15.7.4 Component audit**

There are no auditable events foreseen related to FPT\_ITC.1.

**15.7.5 Application notes**

This family defines the rules for the protection from unauthorized disclosure of TSF data 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.

This family is used in a distributed context where the TSF is providing TSF data to another trusted IT product. The TSF can only take the measures at its site and cannot be held responsible for the behaviour of the other trusted IT product.

## 15.7.6 Evaluator notes

Confidentiality of TSF Data during transmission is necessary to protect such information from disclosure.

Some possible implementations that can provide confidentiality include the use of cryptographic algorithms as well as spread spectrum techniques.

## 15.7.7 FPT\_ITC.1 Inter-TSF confidentiality during transmission

### 15.7.7.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

### 15.7.7.2 Component rationale

This component is used when it is necessary to make the requirement for confidentiality of TSF data when being transmitted from the TSF to another trusted IT product.

### 15.7.7.3 Element FPT\_ITC.1.1

**The TSF shall protect all TSF data transmitted from the TSF to another trusted IT product from unauthorized disclosure during transmission.**

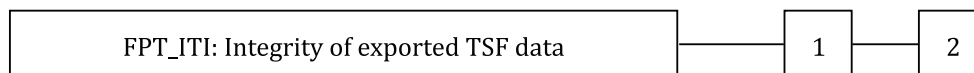
## 15.8 Integrity of exported TSF data (FPT\_ITI)

### 15.8.1 Family Behaviour

This family defines the rules for the protection, from unauthorized modification, of TSF data during transmission between the TSF and another trusted IT product.

### 15.8.2 Component levelling and description

[Figure 67](#) illustrates the component levelling of family FPT\_ITI.



#### Key

1,2 components

**Figure 67 — FPT\_ITI: Component levelling**

FPT\_ITI.1 (Inter-TSF detection of modification) ([15.8.7](#)), provides the ability to detect modification of TSF data during transmission between the TSF and another trusted IT product, under the assumption that another trusted IT product is cognizant of the mechanism used.

FPT\_ITI.2 (Inter-TSF detection and correction of modification) ([15.8.8](#)) provides the ability for another trusted IT product not only to detect modification, but to correct modified TSF data under the assumption that another trusted IT product is cognizant of the mechanism used.

### 15.8.3 Component management

The following actions related to FPT\_ITI.2 can be considered for the management functions in FMT (Security management):

- management of the types of TSF data that the TSF tries to correct if modified in transit;

## ISO/IEC 15408-2:2026(en)

- management of the types of action that the TSF takes if TSF data is modified in transit.

There are no management activities foreseen related to FPT\_ITI.1.

### 15.8.4 Component audit

The following actions related to FPT\_ITI.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the detection of modification of transmitted TSF data;
- basic: the action taken upon detection of modification of transmitted TSF data.

The following actions related to FPT\_ITI.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the detection of modification of transmitted TSF data;
- basic: the action taken upon detection of modification of transmitted TSF data;
- basic: the use of the correction mechanism.

### 15.8.5 Application notes

This family defines the rules for the protection, from unauthorized modification, of TSF data 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.

This family is used in a distributed context where the TSF is exchanging TSF data with another trusted IT product. Note that a requirement that addresses modification, detection, or recovery at another trusted IT product cannot be specified, as the mechanisms that another trusted IT product will use to protect its data cannot be determined in advance. For this reason, these requirements are expressed in terms of the "TSF providing a capability" which another trusted IT product can use.

### 15.8.6 Evaluator notes

In the FPT\_ITI.2 (Inter-TSF detection and correction of modification) ([15.8.8](#)) component some possible means of satisfying this requirement involves the use of cryptographic functions or some form of checksum.

### 15.8.7 FPT\_ITI.1 Inter-TSF detection of modification

#### 15.8.7.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 15.8.7.2 Component rationale

This component should be used in situations where it is sufficient to detect when data have been modified. An example of such a situation is one in which another trusted IT product can request the TOE's TSF to retransmit data when modification has been detected or respond to such types of request.

The desired strength of modification detection is based upon a specified modification metric that is a function of the algorithm used, which may range from a weak checksum and parity mechanisms that may fail to detect multiple bit changes, to more complicated cryptographic checksum approaches.

**ISO/IEC 15408-2:2026(en)****15.8.7.3 Element FPT\_ITI.1.1**

**The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and another trusted IT product within the following metric: [assignment <sup>(a1)</sup>: *a defined modification metric*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the modification metric that the detection mechanism satisfies. This modification metric shall specify the desired strength of the modification detection.

**15.8.7.4 Element FPT\_ITI.1.2**

**The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and another trusted IT product and perform [assignment <sup>(a1)</sup>: *action to be taken*] if modifications are detected.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the actions to be taken if a modification of TSF data has been detected. An example of an action is: "ignore the TSF data and request the originating trusted product to send the TSF data again".

**15.8.8 FPT\_ITI.2 Inter-TSF detection and correction of modification****15.8.8.1 Component relationships**

Hierarchical to: FPT\_ITI.1 (Inter-TSF detection of modification) ([15.8.7](#))

Dependencies: No dependencies.

**15.8.8.2 Component rationale**

This component should be used in situations where it is necessary to detect or correct modifications of TSF critical data.

The desired strength of modification detection is based upon a specified modification metric that is a function of the algorithm used, which may range from a checksum and parity mechanisms that may fail to detect multiple bit changes, to more complicated cryptographic checksum approaches. The metric that is defined can either refer to the attacks it will resist or to mechanisms that are well known in the public literature.

**EXAMPLE** An example of an attack reference is: "Only 1 in 1000 random messages will be accepted." An example of a well-known mechanism is: "The strength must be conformant to the strength offered by the Secure Hash Algorithm."

The approach taken to correct modification can be done through some form of error correcting checksum.

**15.8.8.3 Element FPT\_ITI.2.1**

**The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and another trusted IT product within the following metric: [assignment <sup>(a1)</sup>: *a defined modification metric*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the modification metric that the detection mechanism satisfies. This modification metric shall specify the desired strength of the modification detection.

**15.8.8.4 Element FPT\_ITI.2.2**

**The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and another trusted IT product and perform [assignment <sup>(a1)</sup>: *action to be taken*] if modifications are detected.**

## ISO/IEC 15408-2:2026(en)

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the actions to be taken if a modification of TSF data has been detected.

### 15.8.8.5 Element FPT\_ITI.2.3

**The TSF shall provide the capability to correct [assignment (a1): *type of modification*] of all TSF data transmitted between the TSF and another trusted IT product.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should define the types of modification from which the TSF should be capable of recovering.

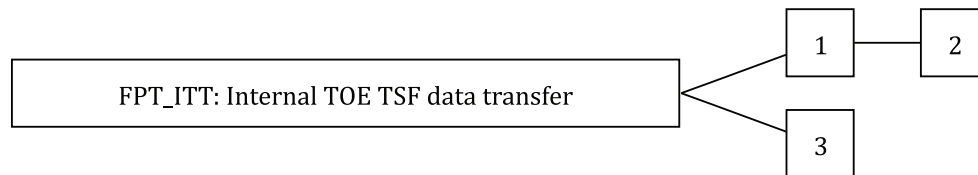
## 15.9 Internal TOE TSF data transfer (FPT\_ITT)

### 15.9.1 Family Behaviour

This family provides requirements that address protection of TSF data when it is transferred between separate parts of a TOE across an internal channel.

### 15.9.2 Component levelling and description

[Figure 68](#) illustrates the component levelling of family FPT\_ITT.



#### Key

1,2,3 components

**Figure 68 — FPT\_ITT: Component levelling**

FPT\_ITT.1 (Basic internal TSF data transfer protection) ([15.9.7](#)) requires that TSF data be protected when transmitted between separate parts of the TOE.

FPT\_ITT.2 (TSF data transfer separation) ([15.9.8](#)) requires that the TSF separate user data from TSF data during transmission.

FPT\_ITT.3 (TSF data integrity monitoring) ([15.9.9](#)) requires that the TSF data transmitted between separate parts of the TOE is monitored for identified integrity errors.

### 15.9.3 Component management

The following actions related to FPT\_ITT.1 can be considered for the management functions in FMT (Security management):

- management of the types of modification against which the TSF should protect;
- management of the mechanism used to provide the protection of the data in transit between different parts of the TSF.

The following actions related to FPT\_ITT.2 can be considered for the management functions in FMT (Security management):

- management of the types of modification against which the TSF should protect;
- management of the mechanism used to provide the protection of the data in transit between different parts of the TSF;

## ISO/IEC 15408-2:2026(en)

- management of the separation mechanism.

The following actions related to FPT\_ITT.3 can be considered for the management functions in FMT (Security management):

- management of the types of modification against which the TSF should protect;
- management of the mechanism used to provide the protection of the data in transit between different parts of the TSF;
- management of the types of modification of TSF data the TSF should try to detect;
- management of the actions that will be taken.

### 15.9.4 Component audit

The following actions related to FPT\_ITT.3 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the detection of modification of TSF data;
- basic: the action taken following detection of an integrity error.

There are no auditable events foreseen related to FPT\_ITT.1 or FPT\_ITT.2.

### 15.9.5 Application 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.

### 15.9.6 Evaluator notes

One practical mechanism available to a TSF to provide this protection is a cryptographically-based mechanism.

### 15.9.7 FPT\_ITT.1 Basic internal TSF data transfer protection

#### 15.9.7.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 15.9.7.2 Element FPT\_ITT.1.1

**The TSF shall protect TSF data from [selection (s1): *disclosure, modification*] when it is transmitted between separate parts of the TOE.**

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify the desired type of protection to be provided from the choices: disclosure, modification.

**ISO/IEC 15408-2:2026(en)****15.9.8 FPT\_ITT.2 TSF data transfer separation****15.9.8.1 Component relationships**

Hierarchical to: FPT\_ITT.1 (Basic internal TSF data transfer protection) ([15.9.7](#))

Dependencies: No dependencies.

**15.9.8.2 Component rationale**

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.

**15.9.8.3 Element FPT\_ITT.2.1**

The TSF shall protect TSF data from [selection (s<sup>1</sup>): *disclosure, modification*] when it is transmitted between separate parts of the TOE.

In selection (s<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should specify the desired type of protection to be provided from the choices: disclosure, modification.

**15.9.8.4 Element FPT\_ITT.2.2**

**The TSF shall separate user data from TSF data when such data is transmitted between separate parts of the TOE.**

**15.9.9 FPT\_ITT.3 TSF data integrity monitoring****15.9.9.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FPT\_ITT.1 (Basic internal TSF data transfer protection) ([15.9.7](#))

**15.9.9.2 Element FPT\_ITT.3.1**

**The TSF shall be able to detect [selection (s<sup>1</sup>): *modification of data, substitution of data, re-ordering of data, deletion of data, [assignment (a<sup>1</sup>): other integrity errors]*] for TSF data transmitted between separate parts of the TOE.**

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should specify the desired type of modification that the TSF shall be able to detect. The author of a PP, PP-Module, functional package or ST should select from: modification of data, substitution of data, re-ordering of data, deletion of data, or any other integrity errors.

In selection (s<sup>1</sup>), if the author of a PP, PP-Module, functional package or ST chooses the latter selection noted in the preceding paragraph, then the author should also specify what those other integrity errors are that the TSF should be capable of detecting.

**15.9.9.3 Element FPT\_ITT.3.2**

**Upon detection of a data integrity error, the TSF shall take the following actions: [assignment (a<sup>1</sup>): *specify the action to be taken*].**

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should specify the action to be taken when an integrity error is identified.

## 15.10 TSF physical protection (FPT\_PHP)

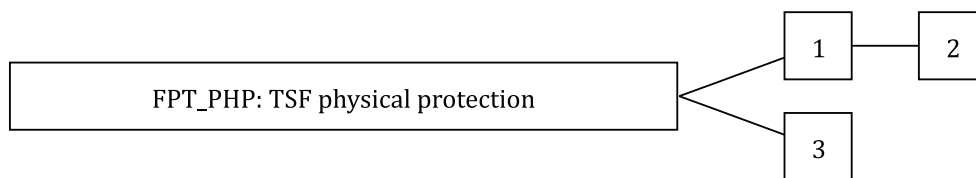
### 15.10.1 Family Behaviour

TSF physical protection components refer to restrictions on unauthorized physical access to the TSF, and to the deterrence of, and resistance to, unauthorized physical modification, or substitution of the TSF.

The requirements of components in this family ensure that the TSF is protected from physical tampering and interference. Satisfying the requirements of these components results in the TSF being packaged and used in such a manner that physical tampering is detectable, or resistance to physical tampering is enforced. Without these components, the protection functions of a TSF lose their effectiveness in environments where physical damage cannot be prevented. This family also provides requirements regarding how the TSF shall respond to physical tampering attempts.

### 15.10.2 Component levelling and description

[Figure 69](#) illustrates the component levelling of family FPT\_PHP.



#### Key

1,2,3 components

**Figure 69 — FPT\_PHP: Component levelling**

FPT\_PHP.1 (Passive detection of physical attack) ([15.10.6](#)), 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 invokes a security administrative function or perform manual inspection to determine if tampering has occurred.

FPT\_PHP.2 (Notification of physical attack) ([15.10.7](#)), provides for automatic notification of tampering for an identified subset of physical penetrations.

FPT\_PHP.3 (Resistance to physical attack) ([15.10.8](#)), provides for features that prevent or resist physical tampering with TSF devices and TSF elements.

### 15.10.3 Component management

The following action related to FPT\_PHP.1 can be considered for the management functions in FMT (Security management):

- management of the user or role that determines whether physical tampering has occurred.

The following actions related to FPT\_PHP.2 can be considered for the management functions in FMT (Security management):

- management of the user or role that gets informed about intrusions;
- management of the list of devices that should inform the indicated user or role about the intrusion.

The following action related to FPT\_PHP.3 can be considered for the management functions in FMT (Security management):

- management of the automatic responses to physical tampering.

#### **15.10.4 Component audit**

The following action related to FPT\_PHP.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: if detection by it means, detection of intrusion.

The following action related to FPT\_PHP.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: detection of intrusion.

There are no auditable events foreseen related to FPT\_PHP.3.

#### **15.10.5 Application notes**

TSF physical protection components refer to restrictions on unauthorized physical access to the TSF, and to the deterrence of, and resistance to, unauthorized physical modification, or substitution of the TSF.

The requirements in this family ensure that the TSF is protected from physical tampering and interference. Satisfying the requirements of these components results in the TSF being packaged and used in such a manner that physical tampering is detectable, or resistance to physical tampering is measurable based on defined work factors. Without these components, the protection functions of a TSF lose their effectiveness in environments where physical damage cannot be prevented. This component also provides requirements regarding how the TSF respond to physical tampering attempts.

**EXAMPLE 1** Examples of physical tampering scenarios include mechanical attack, radiation, changing the temperature.

It is acceptable for the functions that are available to an authorized user for detecting physical tampering to be available only in an off-line or maintenance mode. Controls should be in place to limit access during such modes to authorized users. It is possible that the TSF is not "operational" during those modes, and hence unable to provide normal enforcement for authorized user access. The physical implementation of a TOE can consist of several structures. This set of "elements" as a whole protect (protect, notify and resist) the TSF from physical tampering. This does not mean that all devices provide these features, but the complete physical construct as a whole should.

**EXAMPLE 2** Examples of structures include an outer shielding, cards, and chips.

Although there is only minimal auditing associating with these components, this is solely because there is the potential that the detection and alarm mechanisms may be implemented completely in hardware, below the level of interaction with an audit subsystem. Nevertheless, a PP, PP-Module, functional package or ST author may determine that for a particular anticipated threat environment, there is a need to audit physical tampering. If this is the case, the author of a PP, PP-Module, functional package or ST should include appropriate requirements in the list of audit events.

Inclusion of these requirements can have implications on the hardware design and its interface to the software.

**EXAMPLE 3** 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.

#### **15.10.6 FPT\_PHP.1 Passive detection of physical attack**

##### **15.10.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

## ISO/IEC 15408-2:2026(en)

### 15.10.6.2 Component rationale

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\_MOF.1 (Management of security functions behaviour) (13.4.6) 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.

### 15.10.6.3 Element FPT\_PHP.1.1

**The TSF shall provide unambiguous detection of physical tampering that can compromise the TSF.**

### 15.10.6.4 Element FPT\_PHP.1.2

**The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.**

### 15.10.7 FPT\_PHP.2 Notification of physical attack

#### 15.10.7.1 Component relationships

Hierarchical to: FPT\_PHP.1 (Passive detection of physical attack) (15.10.6)

Dependencies: FMT\_MOF.1 (Management of security functions behaviour) (13.4.6)

#### 15.10.7.2 Component rationale

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, is not noticed. Specification of management functions in FMT\_MOF.1 (Management of security functions behaviour) (13.4.6) should be considered to specify who can make use of that capability, and how they can make use of that capability.

#### 15.10.7.3 Element FPT\_PHP.2.1

**The TSF shall provide unambiguous detection of physical tampering that can compromise the TSF.**

#### 15.10.7.4 Element FPT\_PHP.2.2

**The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.**

#### 15.10.7.5 Element FPT\_PHP.2.3

**For [assignment (a<sup>1</sup>): *list of TSF devices/elements for which active detection is required*], the TSF shall monitor the devices and elements and notify [assignment (a<sup>2</sup>): *a designated user or role*] when physical tampering with the TSF's devices or TSF's elements has occurred.**

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should provide a list of TSF devices/elements for which active detection of physical tampering is required.

In assignment (a<sup>2</sup>), the author of a PP, PP-Module, functional package or ST should designate a user or role that is intended to be notified when tampering is detected. The type of user or role may vary depending

## ISO/IEC 15408-2:2026(en)

on the particular security administration component (from FMT\_MOF.1 (Management of security functions behaviour) (13.4.6)) included in the PP, PP-Module, functional package or ST.

### 15.10.8 FPT\_PHP.3 Resistance to physical attack

#### 15.10.8.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 15.10.8.2 Component rationale

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.

#### 15.10.8.3 Element FPT\_PHP.3.1

**The TSF shall resist [assignment (a1): *physical tampering scenarios*] to the [assignment (a2): *list of TSF devices/elements*] by responding automatically such that the SFRs are always enforced.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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 cannot be retrieved.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the list of TSF devices/elements for which the TSF should resist physical tampering in the scenarios that have been identified.

### 15.11 Trusted recovery (FPT\_RCV)

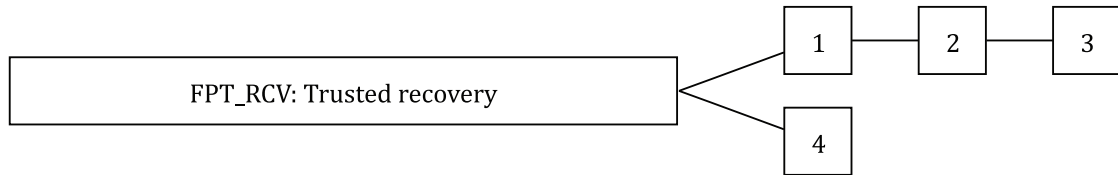
#### 15.11.1 Family Behaviour

The requirements of this family ensure that the TSF can determine that the TOE is started up without protection compromise and can recover without protection compromise after discontinuity of operations. This family is important because the start-up state of the TSF determines the protection of subsequent states.

#### 15.11.2 Component levelling and description

[Figure 70](#) illustrates the component levelling of family FPT\_RCV.

## ISO/IEC 15408-2:2026(en)



### Key

1,2,3,4 components

**Figure 70 — FPT\_RCV: Component levelling**

FPT\_RCV.1 (Manual recovery) (15.11.7), allows a TOE to only provide mechanisms that involve human intervention to return to a secure state.

FPT\_RCV.2 (Automated recovery) (15.11.8), provides, for at least one type of service discontinuity, recovery to a secure state without human intervention; recovery for other discontinuities that can require human intervention.

FPT\_RCV.3 (Automated recovery without undue loss) (15.11.9), also provides for automated recovery, but strengthens the requirements by disallowing undue loss of protected objects.

FPT\_RCV.4 (Function recovery) (15.11.10), provides for recovery at the level of particular functions, ensuring either successful completion or rollback of TSF data to a secure state.

### 15.11.3 Component management

The following action related to FPT\_RCV.1 can be considered for the management functions in FMT (Security management):

- management of who can access the restore capability within the maintenance mode.

The following actions related to FPT\_RCV.2 and FPT\_RCV.3 can be considered for the management functions in FMT (Security management):

- management of who can access the restore capability within the maintenance mode;
- management of the list of failures/service discontinuities that will be handled through the automatic procedures.

There are no management activities foreseen related to FPT\_RCV.4.

### 15.11.4 Component audit

The following actions related to FPT\_RCV.1, FPT\_RCV.2 and FPT\_RCV.3 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: the fact that a failure or service discontinuity occurred;
- minimal: resumption of the regular operation;
- basic: type of failure or service discontinuity.

The following actions related to FPT\_RCV.4 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: if possible, the impossibility to return to a secure state after a failure of the TSF;
- basic: if possible, the detection of a failure of a function.

## ISO/IEC 15408-2:2026(en)

### 15.11.5 Application notes

The requirements of this family ensure that the TSF can determine that the TOE is started-up without protection compromise and can recover without protection compromise after discontinuity of operations. This family is important because the start-up state of the TSF determines the protection of subsequent states.

Recovery components reconstruct the TSF secure states, or prevent transitions to insecure states, as a direct response to occurrences of expected failures, discontinuity of operation or start-up. Failures that must be generally anticipated include the following:

- 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);
- 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);
- 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).

Recovery can be from either a complete or partial failure scenario. Although a complete failure can occur in a monolithic operating system, it is less likely to occur in a distributed environment. In such environments, subsystems may fail, but other portions remain operational. Further, critical components may be redundant (disk mirroring, alternative routes), and checkpoints may be available. Thus, recovery is expressed in terms of recovery to a secure state.

There are different interactions between FPT\_RCV (Trusted recovery) and FPT\_TST (TSF self-test) ([15.18](#)) components to be considered when selecting FPT\_RCV (Trusted recovery):

- a) the need for trusted recovery may be indicated through the results of TSF self-testing, where the results of the self-tests indicate that the TSF is in an insecure state and return to a secure state or entrance in maintenance mode is required;
- b) a failure, as discussed in a), may be identified by an administrator. Either the administrator may perform the actions to return the TOE to a secure state and then invoke TSF self-tests to confirm that the secure state has been achieved. Or, the TSF self-tests may be invoked to complete the recovery process;
- c) a combination of a) and b) above, where the need for trusted recovery is indicated through the results of TSF self-testing, the administrator performs the actions to return the TOE to a secure state and then invokes TSF self-tests to confirm that the secure state has been achieved;
- d) self-tests detect a failure/service discontinuity, then either automated recovery or entrance to a maintenance mode.

This family identifies a maintenance mode. In this maintenance mode, normal operation can be impossible or severely restricted, as otherwise insecure situations can occur. Typically, only authorized users should be allowed access to this mode but the real details of who can access this mode is a function of FMT (Security management). If FMT (Security management) does not put any controls on who can access this mode, then it may be acceptable to allow any user to restore the system if the TOE enters such a state. However, in practice, this is probably not desirable as the user restoring the system has an opportunity to configure the TOE in such a way as to violate the SFRs.

Mechanisms designed to detect exceptional conditions during operation fall under FPT\_TST (TSF self-test) ([15.18](#)), FPT\_FLS (Fail secure), and other areas that address the concept of "Software Safety". It is likely that the use of one of these families will be required to support the adoption of FPT\_RCV (Trusted recovery). This is to ensure that the TOE will be able to detect when recovery is required.

**ISO/IEC 15408-2:2026(en)**

Throughout this family, the phrase "secure state" is used. This refers to some state in which the TOE has consistent TSF data and a TSF that can correctly enforce the policy. This state may be the initial "boot" of a clean system, or it can be some checkpointed state.

Following recovery, it may be necessary to confirm that the secure state has been achieved through self-testing of the TSF. However, if the recovery is performed in a manner such that only a secure state can be achieved, else recovery fails, then the dependency to the FPT\_TST.1 (TSF self-testing) (15.18.7) component may be argued away.

**15.11.6 Evaluator notes**

In FPT\_RCV.1 (Manual recovery) (15.11.7), it is acceptable for the functions that are available to an authorized user for trusted recovery to be available only in a maintenance mode. Controls should be in place to limit access during maintenance to authorized users.

In FPT\_RCV.2 (Automated recovery) (15.11.8), it is acceptable for the functions that are available to an authorized user for trusted recovery to be available only in a maintenance mode. Controls should be in place to limit access during maintenance to authorized users.

For FPT\_RCV.2.1, it is the responsibility of the developer of the TSF to determine the set of recoverable failures and service discontinuities.

In FPT\_RCV.3 (Automated recovery without undue loss) (15.11.9), it is acceptable for the functions that are available to an authorized user for trusted recovery to be available only in a maintenance mode. Controls should be in place to limit access during maintenance to authorized users.

It is assumed that the evaluators will verify the robustness of the automated recovery mechanisms.

**15.11.7 FPT\_RCV.1 Manual recovery****15.11.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: AGD\_OPE.1 (Operational user guidance) (see ISO/IEC 15408-3:2026, 11.2.4)

**15.11.7.2 Component rationale**

In the hierarchy of the trusted recovery family, recovery that requires only manual intervention is the least desirable, for it precludes the use of the system in an unattended fashion.

This component is intended for use in TOEs that do not require unattended recovery to a secure state. The requirements of this component reduce the threat of protection compromise resulting from an attended TOE returning to an insecure state after recovery from a failure or other discontinuity.

**15.11.7.3 Element FPT\_RCV.1.1**

**After [assignment <sup>(a1)</sup>: *list of failures/service discontinuities*] the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of failures or service discontinuities following which the TOE will enter a maintenance mode.

## ISO/IEC 15408-2:2026(en)

### 15.11.8 FPT\_RCV.2 Automated recovery

#### 15.11.8.1 Component relationships

Hierarchical to: FPT\_RCV.1 (Manual recovery) ([15.11.7](#))

Dependencies: AGD\_OPE.1 (Operational user guidance) (see ISO/IEC 15408-3:2026, 11.2.4)

#### 15.11.8.2 Component rationale

Automated recovery is considered to be more useful than manual recovery, as it allows the machine to operate in an unattended fashion.

The component FPT\_RCV.2 (Automated recovery) extends the feature coverage of FPT\_RCV.1 (Manual recovery) ([15.11.7](#)) by requiring that there be at least one automated method of recovery from failure or service discontinuity. It addresses the threat of protection compromise resulting from an unattended TOE returning to an insecure state after recovery from a failure or other discontinuity.

#### 15.11.8.3 Element FPT\_RCV.2.1

**When automated recovery from [assignment (a1): *list of failures/service discontinuities*] is not possible**, the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of failures or service discontinuities following which the TOE will need to enter a maintenance mode.

#### 15.11.8.4 Element FPT\_RCV.2.2

**For [assignment (a1): *list of failures/service discontinuities*], the TSF shall ensure the return of the TOE to a secure state using automated procedures.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of failures or other discontinuities for which automated recovery shall be possible.

### 15.11.9 FPT\_RCV.3 Automated recovery without undue loss

#### 15.11.9.1 Component relationships

Hierarchical to: FPT\_RCV.2 (Automated recovery) ([15.11.8](#))

Dependencies: AGD\_OPE.1 (Operational user guidance) (see ISO/IEC 15408-3:2026, 11.2.4)

#### 15.11.9.2 Component rationale

Automated recovery is considered to be more useful than manual recovery, but it runs the risk of losing a substantial number of objects. Preventing undue loss of objects provides additional utility to the recovery effort.

The component FPT\_RCV.3 (Automated recovery without undue loss) extends the feature coverage of FPT\_RCV.2 (Automated recovery) ([15.11.8](#)) by requiring that there not be undue loss of TSF data or objects under the control of the TSF. At FPT\_RCV.2 (Automated recovery) ([15.11.8](#)), the automated recovery mechanisms can conceivably recover by deleting all objects and returning the TSF to a known secure state. This type of drastic automated recovery is precluded in FPT\_RCV.3 (Automated recovery without undue loss).

This component addresses the threat of protection compromise resulting from an unattended TOE returning to an insecure state after recovery from a failure or other discontinuity with a large loss of TSF data or objects under the control of the TSF.

**ISO/IEC 15408-2:2026(en)****15.11.9.3 Element FPT\_RCV.3.1**

When automated recovery from [assignment <sup>(a1)</sup>: *list of failures/service discontinuities*] is not possible, the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of failures or service discontinuities following which the TOE will need to enter a maintenance mode.

**15.11.9.4 Element FPT\_RCV.3.2**

For [assignment <sup>(a1)</sup>: *list of failures/service discontinuities*], the TSF shall ensure the return of the TOE to a secure state using automated procedures.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of failures or other discontinuities for which automated recovery is possible.

**15.11.9.5 Element FPT\_RCV.3.3**

**The functions provided by the TSF to recover from failure or service discontinuity shall ensure that the secure initial state is restored without exceeding [assignment <sup>(a1)</sup>: *quantification*] for loss of TSF data or objects under the control of the TSF.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should provide a quantification for the amount of loss of TSF data or objects that is acceptable.

**15.11.9.6 Element FPT\_RCV.3.4**

**The TSF shall provide the capability to determine the objects that were or were not capable of being recovered.**

**15.11.10 FPT\_RCV.4 Function recovery****15.11.10.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**15.11.10.2 Component rationale**

Function recovery requires that if there should be some failure in the TSF, that certain functions in the TSF should either complete successfully or recover to a secure state.

**15.11.10.3 Element FPT\_RCV.4.1**

**The TSF shall ensure that [assignment <sup>(a1)</sup>: *list of functions and failure scenarios*] have the property that the function either completes successfully, or for the indicated failure scenarios, recovers to a consistent and secure state.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify a list of the functions and failure scenarios. In the event that any of the identified failure scenarios happen, the functions that have been specified shall either complete successfully or recover to a consistent and secure state.

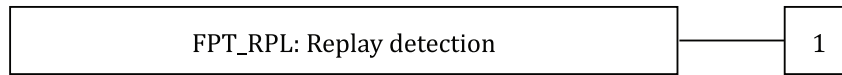
## 15.12 Replay detection (FPT\_RPL)

### 15.12.1 Family Behaviour

This family addresses detection of replay for various types of entities and subsequent actions to correct. In the case where replay may be detected, this effectively prevents it.

### 15.12.2 Component levelling and description

[Figure 71](#) illustrates the component levelling of family FPT\_RPL.



#### Key

1 component

**Figure 71 — FPT\_RPL: Component levelling**

The family consists of only one component, FPT\_RPL.1 (Replay detection) ([15.12.6](#)), which requires that the TSF shall be able to detect the replay of identified entities.

### 15.12.3 Component management

The following actions related to FPT\_RPL.1 can be considered for the management functions in FMT (Security management):

- management of the list of identified entities for which replay is detected;
- management of the list of actions that are required to be taken in case of replay.

### 15.12.4 Component audit

The following actions related to FPT\_RPL.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- basic: detected replay attacks;
- detailed: action to be taken based on the specific actions.

### 15.12.5 Application notes

This family addresses detection of replay for various types of entities and subsequent actions to correct.

### 15.12.6 FPT\_RPL.1 Replay detection

#### 15.12.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 15.12.6.2 Component rationale

The entities included here are those that can be involved in replay detection.

EXAMPLE Messages, service requests, service responses, or sessions.

### 15.12.6.3 Element FPT\_RPL.1.1

The TSF shall detect replay for the following entities: [assignment <sup>(a1)</sup>: *list of identified entities*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should provide a list of identified entities for which detection of replay should be possible.

### 15.12.6.4 Element FPT\_RPL.1.2

The TSF shall perform [assignment <sup>(a1)</sup>: *list of specific actions*] when replay is detected.

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of actions to be taken by the TSF when replay is detected. The potential set of actions that can be taken includes: ignoring the replayed entity, requesting confirmation of the entity from the identified source, and terminating the subject from which the re-played entity originated.

## 15.13 State synchrony protocol (FPT\_SSP)

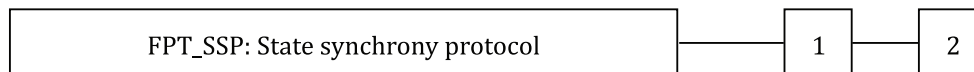
### 15.13.1 Family Behaviour

Distributed TOEs 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.

FPT\_SSP (State synchrony protocol) establishes the requirement for certain critical functions of the TSF to use this trusted protocol. FPT\_SSP (State synchrony protocol) ensures that two distributed parts of the TOE have synchronized their states after a security-relevant action.

### 15.13.2 Component levelling and description

[Figure 72](#) illustrates the component levelling of family FPT\_SSP.



#### Key

1,2 components

**Figure 72 — FPT\_SSP: Component levelling**

FPT\_SSP.1 (Simple trusted acknowledgement) ([15.13.6](#)) requires only a simple acknowledgment by the data recipient.

FPT\_SSP.2 (Mutual trusted acknowledgement) ([15.13.7](#)) requires mutual acknowledgment of the data exchange.

### 15.13.3 Component management

There are no management activities foreseen related to FPT\_SSP.1 or FPT\_SSP.2.

### 15.13.4 Component audit

The following action related to FPT\_SSP.1 and FPT\_SSP.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: failure to receive an acknowledgement when expected.

**ISO/IEC 15408-2:2026(en)****15.13.5 Application notes**

Distributed TOEs may 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.

FPT\_SSP (State synchrony protocol) establishes the requirement for certain critical functions of the TSF to use a trusted protocol. FPT\_SSP (State synchrony protocol) ensures that two distributed parts of the TOE, such as hosts, have synchronized their states after a security-relevant action.

It is possible that some states will never be synchronized, or the transaction cost can be too high for practical use.

**EXAMPLE 1** 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.

Indeterminacy is unique to distributed TOEs. Indeterminacy and state synchrony are related, and the same solution may apply. It is futile to design for indeterminate states; the author of a PP, PP-Module, functional package or ST should express other requirements in such cases.

**EXAMPLE 2** In cases where an indeterminate state occurs, the TSF can raise an alarm or log the event for auditing purposes.

**15.13.6 FPT\_SSP.1 Simple trusted acknowledgement****15.13.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FPT\_ITT.1 (Basic internal TSF data transfer protection) ([15.9.7](#))

**15.13.6.2 Component rationale**

In this component, the TSF supplies an acknowledgement to another part of the TSF when requested. This acknowledgement should indicate that one part of a distributed TOE successfully received an unmodified transmission from a different part of the distributed TOE.

**15.13.6.3 Element FPT\_SSP.1.1**

**The TSF shall acknowledge, when requested by another part of the TSF, the receipt of an unmodified TSF data transmission.**

**15.13.7 FPT\_SSP.2 Mutual trusted acknowledgement****15.13.7.1 Component relationships**

Hierarchical to: FPT\_SSP.1 (Simple trusted acknowledgement) ([15.13.6](#))

Dependencies: FPT\_ITT.1 (Basic internal TSF data transfer protection) ([15.9.7](#))

## ISO/IEC 15408-2:2026(en)

### 15.13.7.2 Component rationale

In this component, in addition to the TSF being able to provide an acknowledgement for the receipt of a data transmission, the TSF complies with a request from another part of the TSF 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.

### 15.13.7.3 Element FPT\_SSP.2.1

The TSF shall acknowledge, when requested by another part of the TSF, the receipt of an unmodified TSF data transmission.

### 15.13.7.4 Element FPT\_SSP.2.2

**The TSF shall ensure that the relevant parts of the TSF know the correct status of transmitted data among its different parts, using acknowledgements.**

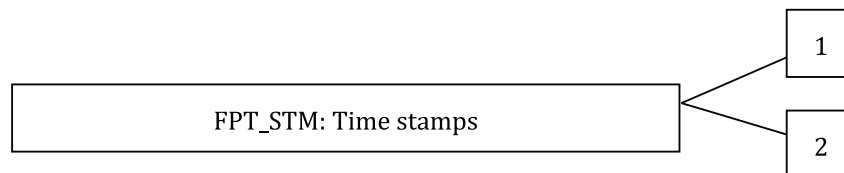
## 15.14 Time stamps (FPT\_STM)

### 15.14.1 Family Behaviour

This family addresses requirements for a reliable time stamp function within a TOE.

### 15.14.2 Component levelling and description

[Figure 73](#) illustrates the component levelling of family FPT\_STM.



#### Key

1,2 components

**Figure 73 — FPT\_STM: Component levelling**

FPT\_STM.1 (Reliable time stamps) ([15.14.6](#)) requires that the TSF provide reliable time stamps for TSF functions.

FPT\_STM.2 (Time source) ([15.14.7](#)), requires the description of the time source used in timestamps.

### 15.14.3 Component management

The following action related to FPT\_STM.1 can be considered for the management functions in FMT (Security management):

- management of the time.

The following action related to FPT\_STM.2 can be considered for the management functions in FMT (Security management):

- setting of time by user authorized according to security policy.

**ISO/IEC 15408-2:2026(en)****15.14.4 Component audit**

The following actions related to FPT\_STM.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: changes to the time;
- detailed: providing a timestamp.

The following actions related to FPT\_STM.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: discontinuous changes to the time;
- detailed: changes to the time source.

**15.14.5 Application notes**

This family addresses requirements for a reliable time stamp function within a TOE.

It is the responsibility of the author of a PP, PP-Module, functional package or ST to clarify the meaning of the phrase "reliable time stamp", and to indicate where the responsibility lies in determining the acceptance of trust.

**15.14.6 FPT\_STM.1 Reliable time stamps****15.14.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**15.14.6.2 Component rationale**

Some possible uses of this component include providing reliable time stamps for the purposes of audit as well as for security attribute expiration.

**15.14.6.3 Element FPT\_STM.1.1**

**The TSF shall be able to provide reliable time stamps.**

**15.14.7 FPT\_STM.2 Time source****15.14.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FPT\_STM.1 (Reliable time stamps) ([15.14.6](#))  
FMT\_SMR.1 (Security roles) ([13.10.6](#))

**15.14.7.2 Component rationale**

In continuation of FPT\_STM.1 (Reliable time stamps) ([15.14.6](#)) Reliable time stamps, FPT\_STM.2 (Time source) focuses on the time source used in such time stamps. FPT\_STM.2 (Time source) requires the description of the time source used for time stamps whereby setting the time directly or configuring another time source by an authorized user according to the respective security policy can be chosen.

## ISO/IEC 15408-2:2026(en)

### 15.14.7.3 Element FPT\_STM.2.1

The TSF shall allow the [assignment <sup>(a1)</sup>: *user authorized by security policy*] to [selection <sup>(s1)</sup>: *set the time, configure another time source*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the user that is authorized by the security policy to choose the time source used in timestamps.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify whether the time can be set directly by that authorized user or result from the configuration of another time source by that authorized user.

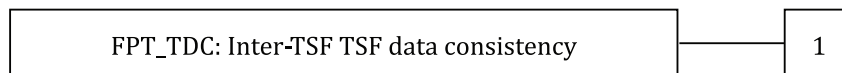
## 15.15 Inter-TSF TSF data consistency (FPT\_TDC)

### 15.15.1 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.

### 15.15.2 Component levelling and description

[Figure 74](#) illustrates the component levelling of family FPT\_TDC.



#### Key

1 component

**Figure 74 — FPT\_TDC: Component levelling**

FPT\_TDC.1 (Inter-TSF basic TSF data consistency) ([15.15.6](#)) requires that the TSF provide the capability to ensure consistency of attributes between TSFs.

### 15.15.3 Component management

There are no management activities foreseen related to FPT\_TDC.1.

### 15.15.4 Component audit

The following actions related to FPT\_TDC.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: successful use of TSF data consistency mechanisms;
- basic: use of the TSF data consistency mechanisms;
- basic: identification of which TSF data have been interpreted;
- basic: detection of modified TSF data.

### 15.15.5 Application notes

In a distributed or composite environment, a TOE may need to exchange TSF data with another trusted IT Product.

EXAMPLE The SFP-attributes associated with data, audit information, identification information.

**ISO/IEC 15408-2:2026(en)**

This family defines the requirements for sharing and consistent interpretation of these 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 can be used to produce security attribute consistency, but they are not provided for here.

This family is different from FDP\_ETC (Export from the TOE) and FDP\_ITC (Import from outside of the TOE), 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 FPT\_ITI (Integrity of exported TSF data) family. These components specify requirements for the TSF to be able to detect or detect and correct modifications to TSF data in transit.

**15.15.6 FPT\_TDC.1 Inter-TSF basic TSF data consistency****15.15.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**15.15.6.2 Component rationale**

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 can 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.

**15.15.6.3 Element FPT\_TDC.1.1**

**The TSF shall provide the capability to consistently interpret [assignment (a1): *list of TSF data types*] when shared between the TSF and another trusted IT product.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST 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.

**15.15.6.4 Element FPT\_TDC.1.2**

**The TSF shall use [assignment (a1): *list of interpretation rules to be applied by the TSF*] when interpreting the TSF data from another trusted IT product.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should assign the list of interpretation rules to be applied by the TSF.

**15.16 Testing of external entities (FPT\_TEE)****15.16.1 Family Behaviour**

This family defines requirements for the TSF to perform tests on one or more external entities.

This component is not intended to be applied to human users.

## ISO/IEC 15408-2:2026(en)

External entities can include applications running on the TOE, hardware or software running "underneath" the TOE (e.g. platforms, operating systems) or applications/boxes connected to the TOE (e.g. intrusion detection systems, firewalls, login servers, time servers).

### 15.16.2 Component levelling and description

[Figure 75](#) illustrates the component levelling of family FPT\_TEE.



#### Key

1 component

**Figure 75 — FPT\_TEE: Component levelling**

FPT\_TEE.1 (Testing of external entities) ([15.16.7](#)), provides for testing of the external entities by the TSF.

### 15.16.3 Component management

The following actions related to FPT\_TEE.1 can be considered for the management functions in FMT (Security management):

- management of the conditions under which the testing of external entities occurs, such as during initial start-up, regular interval, or under specified conditions;
- management of the time interval if appropriate.

### 15.16.4 Component audit

The following action related to FPT\_TEE.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- basic: execution of the tests of the external entities and the results of the tests.

### 15.16.5 Application 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 can be run are:

- tests for the presence of a firewall, and possibly whether it is correctly configured;
- tests of some of the properties of the operating system that an application TOE runs on;
- tests of some of the properties of the IC that a smart card OS TOE runs on (such as the random number generator).

The external entity can "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.

### 15.16.6 Evaluator notes

The tests of external entities should be sufficient to test all of the characteristics of them upon which the TSF relies.

## ISO/IEC 15408-2:2026(en)

For FPT\_TEE.1 (Testing of external entities) (15.16.7), it is acceptable for the functions for periodic testing to be available only in an off-line or maintenance mode. Controls should be in place to limit access, during maintenance, to authorized users.

### 15.16.7 FPT\_TEE.1 Testing of external entities

#### 15.16.7.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 15.16.7.2 Component rationale

This component is not intended to be applied to human users.

This component provides support for the periodic testing of properties related to external entities upon which the TSF's operation depends, by requiring the ability to periodically invoke testing functions.

The author of a PP, PP-Module, functional package or ST may refine the requirement to state whether the function should be available in off-line, on-line or maintenance mode.

#### 15.16.7.3 Element FPT\_TEE.1.1

**The TSF shall run a suite of tests [selection (s<sup>1</sup>): *during initial start-up, periodically during normal operation, at the request of an authorized user, [assignment (a<sup>1</sup>): other conditions]*] to check the fulfilment of [assignment (a<sup>2</sup>): *list of properties of the external entities*].**

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST specifies the properties of the external entities to be checked by the tests.

EXAMPLE 1 Examples of these properties can include configuration or availability properties of a directory server supporting some access control part of the TSF.

In selection (s<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should specify when the TSF will run the testing of external entities, during initial start-up, periodically during normal operation, at the request of an authorized user, or under other conditions. If the tests are run often, then the end users should have more confidence that the TOE is operating correctly than if the tests are run less frequently. However, this need for confidence that the TOE is operating correctly involves a balance with the potential impact on the availability of the TOE, as often times, the testing of external entities may delay the normal operation of a TOE.

In assignment (a<sup>2</sup>), the author of a PP, PP-Module, functional package or ST should, if other conditions are selected, specify the frequency with which the testing of external entities will be run.

EXAMPLE 2 An example of this other frequency or condition can be to run the tests each time a user requests to initiate a session with the TOE. For instance, this can be the case of testing a directory server before its interaction with the TSF during the user authentication process.

#### 15.16.7.4 Element FPT\_TEE.1.2

**If the test fails, the TSF shall [assignment (a<sup>1</sup>): *action(s)*].**

In assignment (a<sup>1</sup>), the author of a PP, PP-Module, functional package or ST should specify what are the action(s) that the TSF shall perform when the testing fails.

EXAMPLE Examples of these action(s), illustrated by a directory server instance, can include connecting to an alternative available server or otherwise looking for a backup server.

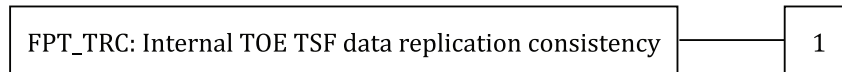
## 15.17 Internal TOE TSF data replication consistency (FPT\_TRC)

### 15.17.1 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 can 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 can occur when parts become disabled.

### 15.17.2 Component levelling and description

[Figure 76](#) illustrates the component levelling of family FPT\_TRC.



#### Key

1 component

**Figure 76 — FPT\_TRC: Component levelling**

This family consists of only one component, FPT\_TRC.1 (Internal TSF consistency) ([15.17.6](#)), which requires that the TSF ensure the consistency of TSF data that is replicated in multiple locations.

### 15.17.3 Component management

There are no management activities foreseen related to FPT\_TRC.1.

### 15.17.4 Component audit

The following actions related to FPT\_TRC.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: restoring consistency upon reconnection;
- basic: detected inconsistency between TSF data.

### 15.17.5 Application notes

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 an internal channel between parts of the TOE becomes inoperative. If the TOE is internally structured as a network of parts of the TOE, this can occur when parts become disabled, network connections are broken, and so on.

The method of ensuring consistency is not specified in this component. It can be attained through a form of transaction logging (where appropriate transactions are "rolled back" to a site upon reconnection); it can also update the replicated data through a synchronization protocol. If a particular protocol is necessary for a PP, PP-Module, functional package or ST, it can be specified through refinement.

It can be impossible to synchronize some states, or the cost of such synchronization can be too high.

**EXAMPLE** Examples of this situation are communication channel and encryption key revocations.

Indeterminate states can also occur; if a specific behaviour is desired, it should be specified via refinement.

## ISO/IEC 15408-2:2026(en)

### 15.17.6 FPT\_TRC.1 Internal TSF consistency

#### 15.17.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FPT\_ITT.1 (Basic internal TSF data transfer protection) ([15.9.7](#))

#### 15.17.6.2 Element FPT\_TRC.1.1

**The TSF shall ensure that TSF data is consistent when replicated between parts of the TOE.**

#### 15.17.6.3 Element FPT\_TRC.1.2

**When parts of the TOE containing replicated TSF data are disconnected, the TSF shall ensure the consistency of the replicated TSF data upon reconnection before processing any requests for [assignment <sup>(a1)</sup>: *list of functions dependent on TSF data replication consistency*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of functions dependent on TSF data replication consistency.

### 15.18 TSF self-test (FPT\_TST)

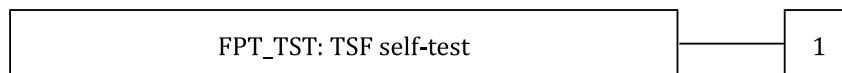
#### 15.18.1 Family Behaviour

The family defines the requirements for the self-testing of the TSF to verify its correct operation. Examples are interfaces to enforcement functions, and sample arithmetical operations on critical parts of the TOE. These tests can be carried out at start-up, periodically, at the request of the authorized user, or when other conditions are met. The actions to be taken by the TOE as the result of self-testing are defined in other families.

The requirements of this family are also needed to detect the corruption of TSF data and TSF itself (i.e. TSF executable code or TSF hardware component) by various failures that do not necessarily stop the TOE's operation (which would be handled by other families). These checks are required to be performed because these failures cannot necessarily be prevented. Such failures can occur either because of unforeseen failure modes or associated oversights in the design of hardware, firmware, or software, or because of malicious corruption of the TSF due to inadequate logical and/or physical protection.

#### 15.18.2 Component levelling and description

[Figure 77](#) illustrates the component levelling of family FPT\_TST.



#### Key

1 component

**Figure 77 — FPT\_TST: Component levelling**

FPT\_TST.1 (TSF self-testing) ([15.18.7](#)), provides the ability to test the TSF's correct operation. These tests can be performed at start-up, periodically, at the request of the authorized user, or when other conditions are met. It also provides the ability to verify the integrity of TSF data and TSF itself.

**ISO/IEC 15408-2:2026(en)****15.18.3 Component management**

The following actions related to FPT\_TST.1 can be considered for the management functions in FMT (Security management):

- management of the conditions under which TSF self-testing occurs, such as during initial start-up, regular interval, or under specified conditions;
- management of the time interval if appropriate.

**15.18.4 Component audit**

The following actions related to FPT\_TST.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: indication that the TSF self-tests were completed and any failures of the tests;
- basic: execution of the TSF self-tests and the results of the tests.

**15.18.5 Application notes**

The family defines the requirements for the self-testing of the TSF to verify its correct operation.

**EXAMPLE** Examples are interfaces to enforcement functions, and sample arithmetical operations on critical parts of the TOE.

These tests can be carried out at start-up, periodically, at the request of an authorized user, or when other conditions are met. The actions to be taken by the TOE as the result of self-testing are defined in other families.

The requirements of this family are also needed to detect the corruption of TSF data and TSF itself (i.e. TSF executable code or TSF hardware component) by various failures that do not necessarily stop the TOE's operation (which would be handled by other families). These checks are performed because these failures can not necessarily be prevented. Such failures can occur either because of unforeseen failure modes or associated oversights in the design of hardware, firmware, or software, or because of malicious corruption of the TSF due to inadequate logical and/or physical protection.

In addition, use of this component may, with appropriate conditions, help to prevent inappropriate or damaging TSF changes being applied to an operational TOE as the result of maintenance activities.

The term "correct operation of the TSF" refers primarily to the operation of the TSF and the integrity of the TSF data.

**15.18.6 Evaluator notes**

For FPT\_TST.1 (TSF self-testing) ([15.18.7](#)) TSF testing, it is acceptable for the functions that are available to the authorized user for periodic testing to be available only in an off-line or maintenance mode. Controls should be in place to limit access during these modes to authorized users.

**15.18.7 FPT\_TST.1 TSF self-testing****15.18.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**ISO/IEC 15408-2:2026(en)****15.18.7.2 Component rationale**

This component provides support for the testing of the critical functions of the TSF's operation by requiring the ability to invoke testing functions and check the integrity of TSF data and executable code.

**15.18.7.3 Element FPT\_TST.1.1**

**The TSF shall run a suite of the following self-tests [selection (s1): *during initial start-up, periodically during normal operation, at the request of the authorized user, at the conditions [assignment (a1): conditions under which self-test should occur]* to demonstrate the correct operation of [selection (s2): [assignment (a2): parts of TSF], the TSF]: [assignment (a3): list of self-tests run by the TSF].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should, if selected, specify the conditions under which the self-test should take place.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify when the TSF will execute the TSF test; during initial start-up, periodically during normal operation, at the request of an authorized user, at other conditions. In the case of the latter option, the author of a PP, PP-Module, functional package or ST should also assign what those conditions are via the following assignment.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should, if selected, specify the list of parts of the TSF that will be subject to TSF self-testing.

In selection (s2), the author of a PP, PP-Module, functional package or ST should specify whether the self-tests shall be carried out to demonstrate the correct operation of the entire TSF, or of only specified parts of TSF.

**15.18.7.4 Element FPT\_TST.1.2**

**The TSF shall provide authorized users with the capability to verify the integrity of [selection (s1): [assignment (a1): parts of TSF data], TSF data].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should, if selected, specify the list of TSF data that will be verified for integrity.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify whether data integrity is intended to be verified for all TSF data, or only for selected data.

**15.18.7.5 Element FPT\_TST.1.3**

**The TSF shall provide authorized users with the capability to verify the integrity of [selection (s1): [assignment (a1): parts of TSF], TSF].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should, if selected, specify the list of TSF that will be verified for integrity.

**NOTE** When FCS\_RBG.1 (Random bit generation (RBG)) ([10.5.6](#)) is selected, the standards selected in FCS\_RBG.1 (Random bit generation (RBG)) ([10.5.6](#)) can require a suite of self-tests run by the TSF. The author of a PP, PP-Module, functional package, or ST will review each standard selected so as to meet the whole part or only the referred certain part of the standard.

In selection (s1), the author of a PP, PP-Module, functional package or ST should specify whether TSF integrity is intended to be verified for all TSF, or only for selected TSF.

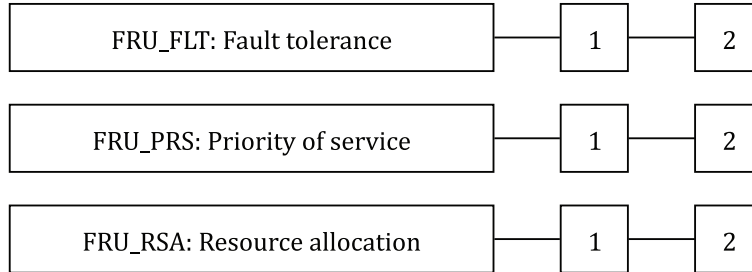
**16 Class FRU Resource utilization****16.1 Introduction**

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

**ISO/IEC 15408-2:2026(en)**

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 78 illustrates the family hierarchy of class FRU.



**Key**  
 1,2 components

**Figure 78 — FRU: Resource utilization, class decomposition**

Direct, indirect and optional dependencies for the components of class FRU are shown in Table 11.

**Table 11 — Dependency table for class FRU: Resource utilization**

	FPT_FLS.1 (15.4.6)
FRU_FLT.1 (16.3.6)	X
FRU_FLT.2 (16.3.7)	X
FRU_PRS.1 (16.4.6)	
FRU_PRS.2 (16.4.7)	
FRU_RSA.1 (16.5.6)	
FRU_RSA.2 (16.5.7)	

**16.2 Notes on class FRU**

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.

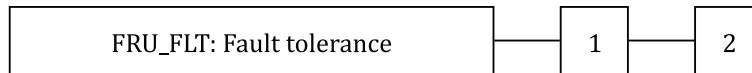
**16.3 Fault tolerance (FRU\_FLT)**

**16.3.1 Family Behaviour**

The requirements of this family ensure that the TOE will maintain correct operation even in the event of failures.

### 16.3.2 Component levelling and description

Figure 79 illustrates the component levelling of family FRU\_FLT.



#### Key

1,2 components

**Figure 79 — FRU\_FLT: Component levelling**

FRU\_FLT.1 (Degraded fault tolerance) (16.3.6) requires the TOE to continue correct operation of identified capabilities in the event of identified failures.

FRU\_FLT.2 (Limited fault tolerance) (16.3.7) requires the TOE to continue correct operation of all capabilities in the event of identified failures.

### 16.3.3 Component management

There are no management activities foreseen related to FRU\_FLT.1 or FRU\_FLT.2.

### 16.3.4 Component audit

The following actions related to FRU\_FLT.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: any failure detected by the TSF;
- basic: all TOE capabilities being discontinued due to a failure.

The following action related to FRU\_FLT.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: any failure detected by the TSF.

### 16.3.5 Application notes

This family provides requirements for the availability of capabilities even in the case of failures.

EXAMPLE 1 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 2 The author of a PP, PP-Module, functional package or ST can 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) (15.4.6).

The mechanisms to provide fault tolerance can 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 can 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).

**ISO/IEC 15408-2:2026(en)****16.3.6 FRU\_FLT.1 Degraded fault tolerance****16.3.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FPT\_FLS.1 (Failure with preservation of secure state) ([15.4.6](#))

**16.3.6.2 Component rationale**

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.

**16.3.6.3 Element FRU\_FLT.1.1**

**The TSF shall ensure the operation of [assignment <sup>(a1)</sup>: *list of TOE capabilities*] when the following failures occur: [assignment <sup>(a2)</sup>: *list of type of failures*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of TOE capabilities the TOE will maintain during and after a specified failure.

In assignment (a2), the author of a PP, PP-Module, functional package or ST should specify the list of types of failures against which the TOE is explicitly protected. If a failure in this list occurs, the TOE will be able to continue its operation.

**16.3.7 FRU\_FLT.2 Limited fault tolerance****16.3.7.1 Component relationships**

Hierarchical to: FRU\_FLT.1 (Degraded fault tolerance) ([16.3.6](#))

Dependencies: FPT\_FLS.1 (Failure with preservation of secure state) ([15.4.6](#))

**16.3.7.2 Component rationale**

This component is intended to specify against what type of failures the TOE be resistant. 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.

**16.3.7.3 Element FRU\_FLT.2.1**

**The TSF shall ensure the operation of all the TOE's capabilities when the following failures occur: [assignment <sup>(a1)</sup>: *list of type of failures*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of types of failures against which the TOE is explicitly protected. If a failure in this list occurs, the TOE will be able to continue its operation.

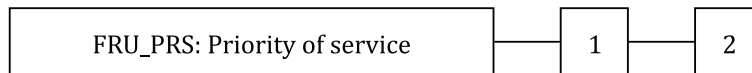
## 16.4 Priority of service (FRU\_PRS)

### 16.4.1 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.

### 16.4.2 Component levelling and description

[Figure 80](#) illustrates the component levelling of family FRU\_PRS.



#### Key

1,2 components

**Figure 80 — FRU\_PRS: Component levelling**

FRU\_PRS.1 (Limited priority of service) ([16.4.6](#)), provides priorities for a subject's use of a subset of the resources under the control of the TSF.

FRU\_PRS.2 (Full priority of service) ([16.4.7](#)), provides priorities for a subject's use of all of the resources under the control of the TSF.

### 16.4.3 Component management

The following action related to FRU\_PRS.1 and FRU\_PRS.2 can be considered for the management functions in FMT (Security management):

- assignment of priorities to each subject in the TSF.

### 16.4.4 Component audit

The following actions related to FRU\_PRS.1 and FRU\_PRS.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: rejection of operation based on the use of priority within an allocation;
- basic: all attempted uses of the allocation function which involves the priority of the service functions.

### 16.4.5 Application notes

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 interference or delay due to low priority activities, i.e. time critical tasks will not be delayed by tasks that are less time critical.

This family can be applicable to several types of resources.

EXAMPLE Processing capacity, and communication channel capacity.

The Priority of Service mechanism can be passive or active. In a passive Priority of Service system, the system will select the task with the highest priority when given a choice between two waiting applications. While using passive Priority of Service mechanisms, when a low priority task is running, it cannot be interrupted by a high priority task. While using an active Priority of Service mechanisms, lower priority tasks can be interrupted by new high priority tasks.

**ISO/IEC 15408-2:2026(en)**

The audit requirement states that all reasons for rejection should be audited. It is left to the developer to argue that an operation is not rejected but delayed.

**16.4.6 FRU\_PRS.1 Limited priority of service****16.4.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**16.4.6.2 Component rationale**

This component defines priorities for a subject, and the resources for which this priority will be used. If some subject attempts to act on a resource controlled by the Priority of Service requirements, the access and/or time of access will be dependent on the subject's priority, the priority of the currently acting subject, and the priority of the subjects still in the queue.

**16.4.6.3 Element FRU\_PRS.1.1**

**The TSF shall assign a priority to each subject in the TSF.**

**16.4.6.4 Element FRU\_PRS.1.2**

**The TSF shall ensure that each access to [assignment <sup>(a1)</sup>: *controlled resources*] shall be mediated on the basis of the assigned priority of the subjects.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of controlled resources for which the TSF enforces priority of service.

**16.4.7 FRU\_PRS.2 Full priority of service****16.4.7.1 Component relationships**

Hierarchical to: FRU\_PRS.1 (Limited priority of service) ([16.4.6](#))

Dependencies: No dependencies.

**16.4.7.2 Component rationale**

This component defines priorities for a subject. All shareable resources under the control of the TSF will be subjected to the Priority of Service mechanism. If some subject attempts to take action on a shareable TSF resource, the access and/or time of access will be dependent on the subject's priority, the priority of the currently acting subject, and the priority of the subjects still in the queue.

**16.4.7.3 Element FRU\_PRS.2.1**

The TSF shall assign a priority to each subject in the TSF.

**16.4.7.4 Element FRU\_PRS.2.2**

The TSF shall ensure that each access to **all shareable resources** shall be mediated on the basis of the assigned priority of the subjects.

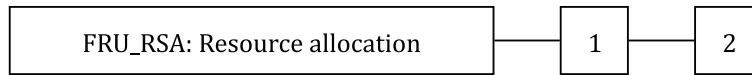
## 16.5 Resource allocation (FRU\_RSA)

### 16.5.1 Family Behaviour

The requirements of this family allow the TSF to control the use of resources by users and subjects such that denial of service will not occur because of unauthorized monopolization of resources.

### 16.5.2 Component levelling and description

[Figure 81](#) illustrates the component levelling of family FRU\_RSA.



#### Key

1,2 components

**Figure 81 — FRU\_RSA: Component levelling**

FRU\_RSA.1 (Maximum quotas) ([16.5.6](#)), provides requirements for quota mechanisms that ensure that users and subjects will not monopolize a controlled resource.

FRU\_RSA.2 (Minimum and maximum quotas) ([16.5.7](#)), 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.

### 16.5.3 Component management

The following action related to FRU\_RSA.1 can be considered for the management functions in FMT (Security management):

- specifying maximum limits for a resource for groups and/or individual users and/or subjects by an administrator.

The following action related to FRU\_RSA.2 can be considered for the management functions in FMT (Security management):

- specifying minimum and maximum limits for a resource for groups and/or individual users and/or subjects by an administrator.

### 16.5.4 Component audit

The following actions related to FRU\_RSA.1 and FRU\_RSA.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: rejection of allocation operation due to resource limits;
- basic: all attempted uses of the resource allocation functions for resources that are under control of the TSF.

### 16.5.5 Application notes

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 unauthorized denial of service will not take place by means of monopolization of resources by other users or subjects.

Resource allocation rules allow the creation of quotas or other means of defining limits on the amount of resource space or time that may be allocated on behalf of a specific user or subjects.

EXAMPLE 1 These rules can be implemented in various ways, such as:

## ISO/IEC 15408-2:2026(en)

- defining object quotas that constrain the number and/or size of objects a specific user can allocate;
- controlling the allocation and deallocation of preassigned resource units to ensure that these units remain under the control of the TSF.

In general, these functions will be implemented through the use of attributes assigned to users and resources.

The objective of these components is to ensure a certain amount of fairness among the users and subjects.

**EXAMPLE 2** One example of ensuring fairness among users is preventing a single user from allocating all the available space.

Since resource allocation often goes beyond the lifespan of a subject (i.e. files often exist longer than the applications that generated them), and multiple instantiations of subjects by the same user should not negatively affect other users too much, the components allow that the allocation limits are related to the users. In some situations, the resources are allocated by a subject.

**EXAMPLE 3** An example of a resource that can be allocated by a subject includes main memory or CPU cycles.

In those instances, the components allow that the resource allocation be on the level of subjects.

This family imposes requirements on resource allocation, not on the use of the resource itself. The audit requirements therefore, as stated, also apply to the allocation of the resource, not to the use of the resource.

### 16.5.6 FRU\_RSA.1 Maximum quotas

#### 16.5.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 16.5.6.2 Component rationale

This component provides requirements for quota mechanisms that apply to only a specified set of the shareable resources in the TOE. The requirements allow the quotas to be associated with a user, possibly assigned to groups of users or subjects as applicable to the TOE.

#### 16.5.6.3 Element FRU\_RSA.1.1

**The TSF shall enforce maximum quotas of the following resources: [assignment <sup>(a1)</sup>: *controlled resources*] that [selection <sup>(s1)</sup>: *individual user, defined group of users, subjects*] can use [selection <sup>(s2)</sup>: *simultaneously, over a specified period of time*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the list of controlled resources for which maximum resource allocation limits are required. If all resources under the control of the TSF are required to be included, the words "all TSF resources" may be specified.

**EXAMPLE** Examples of controlled resources include processes, disk space, memory, and bandwidth.

In selection (s1), the PP/ST author should select whether the maximum quotas apply to individual users, to a defined group of users, or subjects or any combination of these.

In selection (s2), the PP/ST author should select whether the maximum quotas are applicable to any given time (simultaneously), or over a specific time interval.

## 16.5.7 FRU\_RSA.2 Minimum and maximum quotas

### 16.5.7.1 Component relationships

Hierarchical to: FRU\_RSA.1 (Maximum quotas) ([16.5.6](#))

Dependencies: No dependencies.

### 16.5.7.2 Component rationale

This component provides requirements for quota mechanisms that apply to a specified set of the shareable resources in the TOE. The requirements allow the quotas to be associated with a user, or possibly assigned to groups of users as applicable to the TOE.

### 16.5.7.3 Element FRU\_RSA.2.1

The TSF shall enforce maximum quotas of the following resources [assignment <sup>(a1)</sup>: *controlled resources*] that [selection <sup>(s1)</sup>: *individual user, defined group of users, subjects*] can use [selection <sup>(s2)</sup>: *simultaneously, over a specified period of time*].

In assignment (a1), the author of a PP, PP-Module, functional package or ST should specify the controlled resources for which maximum and minimum resource allocation limits are required. If all resources under the control of the TSF are required to be included, the words "all TSF resources" can be specified.

In selection (s2), the author of a PP, PP-Module, functional package or ST should select whether the maximum quotas apply to individual users, to a defined group of users, or subjects or any combination of these.

### 16.5.7.4 Element FRU\_RSA.2.2

**The TSF shall ensure the provision of minimum quantity of each [assignment <sup>(a1)</sup>: *controlled resource*] that is available for [selection <sup>(s1)</sup>: *an individual user, defined group of users, subjects*] to use [selection <sup>(s2)</sup>: *simultaneously, over a specified period of time*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the controlled resources for which a minimum allocation limit is set. If all resources under the control of the TSF are included the words "all TSF resources" can be specified.

**EXAMPLE** Examples of controlled resources include processes, disk space, memory and bandwidth.

In selection (s1), the author of a PP, PP-Module, functional package or ST selects whether the minimum quotas apply to individual users, to a defined group of users, or subjects or any combination of these.

In selection (s2), the author of a PP, PP-Module, functional package or ST selects whether the minimum quotas are applicable to any given time (simultaneously), or over a specific time interval.

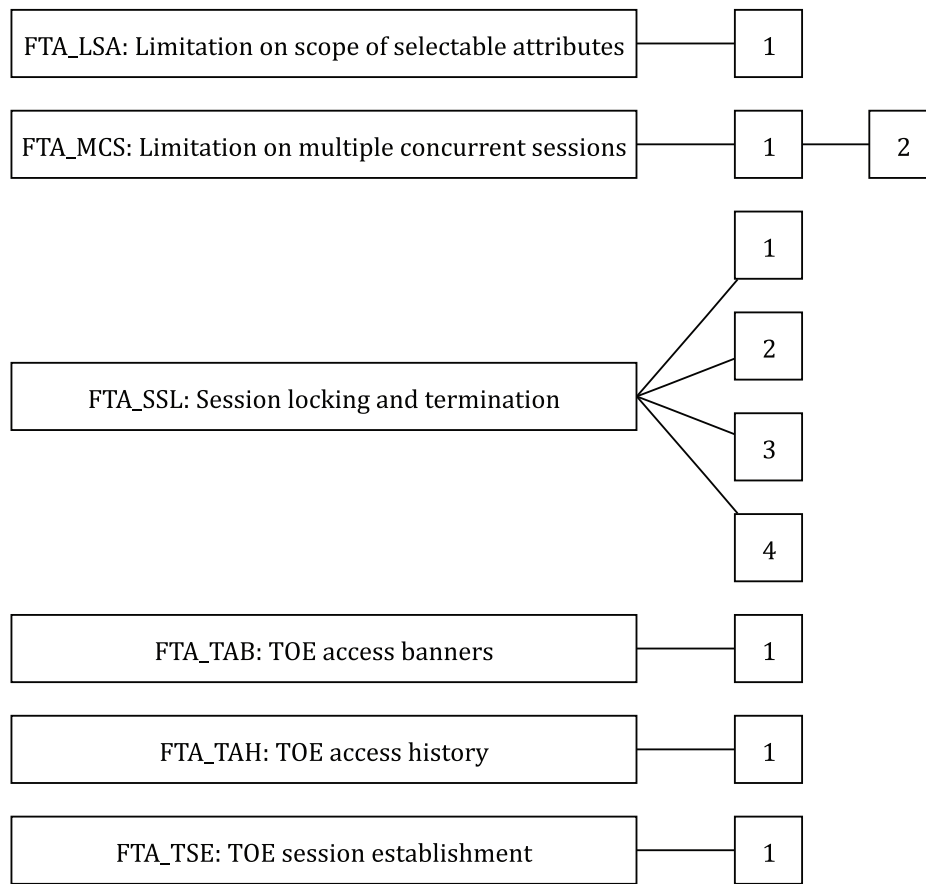
## 17 Class FTA TOE access

### 17.1 Introduction

This family specifies functional requirements for controlling the establishment of a user's session.

[Figure 82](#) illustrates the family hierarchy of class FTA.

**ISO/IEC 15408-2:2026(en)**



**Key**  
 1,2,3,4 components

**Figure 82 — FTA: TOE access, class decomposition**

Direct, indirect and optional dependencies for the components of class FTA are shown in [Table 12](#).

**Table 12 — Dependency table for class FTA: TOE access**

	FIA_UAU.1 (12.7.6)	FIA_UID.1 (12.8.6)	FMT_SMR.1 (13.10.6)
FTA_LSA.1 (17.3.6)			
FTA_MCS.1 (17.4.6)		X	
FTA_MCS.2 (17.4.7)		X	
FTA_SSL.1 (17.5.6)	X		
FTA_SSL.2 (17.5.7)		X	
FTA_SSL.3 (17.5.8)			X
FTA_SSL.4 (17.5.9)			
FTA_TAB.1 (17.6.6)			
FTA_TAH.1 (17.7.6)			
FTA_TSE.1 (17.8.6)			

## 17.2 Notes on class FTA

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.

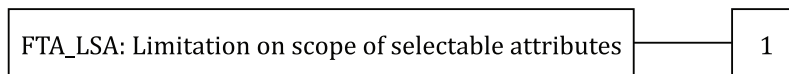
## 17.3 Limitation on scope of selectable attributes (FTA\_LSA)

### 17.3.1 Family Behaviour

This family defines requirements to limit the scope of session security attributes that a user can select for a session.

### 17.3.2 Component levelling and description

[Figure 83](#) illustrates the component levelling of family FTA\_LSA.



#### Key

1 component

**Figure 83 — FTA\_LSA: Component levelling**

FTA\_LSA.1 (Limitation on scope of selectable attributes) ([17.3.6](#)), provides the requirement for a TOE to limit the scope of the session security attributes during session establishment.

### 17.3.3 Component management

The following action related to FTA\_LSA.1 can be considered for the management functions in FMT (Security management):

- management of the scope of the session security attributes by an administrator.

### 17.3.4 Component audit

The following actions related to FTA\_LSA.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: all failed attempts at selecting session security attributes;
- basic: all attempts at selecting session security attributes;
- detailed: capture of the values of each of the session security attributes.

### 17.3.5 Application notes

This family defines requirements that will limit the session security attributes a user can select, and the subjects to which a user can be bound, based on: The method of access, the location or port of access, and/or the time.

**EXAMPLE 1** An example of an environmental condition that can be used to limit session security attributes is a restriction based on the time of day or the day of the week.

**ISO/IEC 15408-2:2026(en)**

This family provides the capability for a PP, PP-Module, functional package or 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 2** A user can be allowed to establish a "secret session" during normal business hours but outside those hours the same user can be constrained to only establishing "unclassified sessions".

The identification of relevant constraints on the domain of selectable attributes can be achieved through the use of the selection operation. These constraints can be applied on an attribute-by-attribute basis. When there exists a need to specify constraints on multiple attributes, this component is required to be replicated for each attribute.

**EXAMPLE 3** Examples of attributes that can 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 can be based upon time-of-day, day-of-week, or calendar dates. This constraint provides some operational protection against user actions that can occur at a time where proper monitoring or where proper procedural measures are not in place.

### **17.3.6 FTA\_LSA.1 Limitation on scope of selectable attributes**

#### **17.3.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

#### **17.3.6.2 Component rationale**

No component notes or rationale have been provided.

#### **17.3.6.3 Element FTA\_LSA.1.1**

**The TSF shall restrict the scope of the session security attributes [assignment <sup>(a1)</sup>: *session security attributes*], based on [assignment <sup>(a2)</sup>: *attributes*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the set of session security attributes that are constrained.

In assignment (a2), examples of these session security attributes are user clearance level, integrity level and roles.

### **17.4 Limitation on multiple concurrent sessions (FTA\_MCS)**

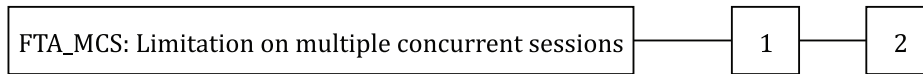
#### **17.4.1 Family Behaviour**

This family defines requirements to place limits on the number of concurrent sessions that belong to the same user.

#### **17.4.2 Component levelling and description**

[Figure 84](#) illustrates the component levelling of family FTA\_MCS.

## ISO/IEC 15408-2:2026(en)



### Key

1,2 components

**Figure 84 — FTA\_MCS: Component levelling**

FTA\_MCS.1 (Basic limitation on multiple concurrent sessions) ([17.4.6](#)), provides limitations that apply to all users of the TSF.

FTA\_MCS.2 (Per user attribute limitation on multiple concurrent sessions) ([17.4.7](#)) extends FTA\_MCS.1 (Basic limitation on multiple concurrent sessions) ([17.4.6](#)) by requiring the ability to specify limitations on the number of concurrent sessions based on the related security attributes.

### 17.4.3 Component management

The following action related to FTA\_MCS.1 can be considered for the management functions in FMT (Security management):

- management of the maximum allowed number of concurrent user sessions by an administrator.

The following action related to FTA\_MCS.2 can be considered for the management functions in FMT (Security management):

- management of the rules that govern the maximum allowed number of concurrent user sessions by an administrator.

### 17.4.4 Component audit

The following actions related to FTA\_MCS.1 and FTA\_MCS.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: rejection of a new session based on the limitation of multiple concurrent sessions;
- detailed: capture of the number of currently concurrent user sessions and the user security attribute(s).

### 17.4.5 Application notes

This family defines how many sessions a user may have at the same time (concurrent sessions). This number of concurrent sessions may either be set for a group of users or for each individual user.

### 17.4.6 FTA\_MCS.1 Basic limitation on multiple concurrent sessions

#### 17.4.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

#### 17.4.6.2 Component rationale

This component allows the system to limit the number of sessions in order to effectively use the resources of the TOE.

#### 17.4.6.3 Element FTA\_MCS.1.1

**The TSF shall restrict the maximum number of concurrent sessions that belong to the same user.**

#### **17.4.6.4 Element FTA\_MCS.1.2**

**The TSF shall enforce, by default, a limit of [assignment <sup>(a1)</sup>: *default number*] sessions per user.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the default number of maximum concurrent sessions to be used.

#### **17.4.7 FTA\_MCS.2 Per user attribute limitation on multiple concurrent sessions**

##### **17.4.7.1 Component relationships**

Hierarchical to: FTA\_MCS.1 (Basic limitation on multiple concurrent sessions) ([17.4.6](#))

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

##### **17.4.7.2 Component rationale**

This component provides additional capabilities over those of FTA\_MCS.1 (Basic limitation on multiple concurrent sessions) ([17.4.6](#)), by allowing further constraints to be placed on the number of concurrent sessions that users are able to invoke. These constraints are in terms of a user's security attributes, such as a user's identity, or membership of a role.

##### **17.4.7.3 Element FTA\_MCS.2.1**

**The TSF shall restrict the maximum number of concurrent sessions that belong to the same user according to the rules [assignment<sup>(a1)</sup>:*rules for the number of maximum concurrent sessions*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the rules that determine the maximum number of concurrent sessions.

##### **17.4.7.4 Element FTA\_MCS.2.2**

The TSF shall enforce, by default, a limit of [assignment <sup>(a1)</sup>: *default number*] sessions per user.

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the default number of maximum concurrent sessions to be used.

### **17.5 Session locking and termination (FTA\_SSL)**

#### **17.5.1 Family Behaviour**

This family defines requirements for the TSF to provide the capability for TSF-initiated and user-initiated locking, unlocking, and termination of interactive sessions.

#### **17.5.2 Component levelling and description**

[Figure 85](#) illustrates the component levelling of family FTA\_SSL.

## ISO/IEC 15408-2:2026(en)



### Key

1,2,3,4 components

**Figure 85 — FTA\_SSL: Component levelling**

FTA\_SSL.1 (TSF-initiated session locking) (17.5.6), includes system-initiated locking of an interactive session after a specified period of user inactivity.

FTA\_SSL.2 (User-initiated locking) (17.5.7), provides capabilities for the user to lock and unlock the user's own interactive sessions.

FTA\_SSL.3 (TSF-initiated termination) (17.5.8), provides requirements for the TSF to terminate the session after a specified period of user inactivity.

FTA\_SSL.4 (User-initiated termination) (17.5.9), provides capabilities for the user to terminate the user's own interactive sessions.

### 17.5.3 Component management

The following actions related to FTA\_SSL.1 can be considered for the management functions in FMT (Security management):

- specification of the time of user inactivity after which lock-out occurs for an individual user;
- specification of the default time of user inactivity after which lock-out occurs;
- management of the events that occur prior to unlocking the session.

The following action related to FTA\_SSL.2 can be considered for the management functions in FMT (Security management):

- management of the events that occur prior to unlocking the session.

The following actions related to FTA\_SSL.3 can be considered for the management functions in FMT (Security management):

- specification of the time of user inactivity after which termination of the interactive session occurs for an individual user;
- specification of the default time of user inactivity after which termination of the interactive session occurs.

There are no management activities foreseen related to FTA\_SSL.4.

**ISO/IEC 15408-2:2026(en)****17.5.4 Component audit**

The following actions related to FTA\_SSL.1 and FTA\_SSL.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: locking of an interactive session by the session locking mechanism;
- minimal: successful unlocking of an interactive session;
- basic: any attempts at unlocking an interactive session.

The following action related to FTA\_SSL.3 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: termination of an interactive session by the session locking mechanism.

The following action related to FTA\_SSL.4 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: termination of an interactive session by the user.

**17.5.5 Application notes**

This family defines requirements for the TSF to provide the capability for TSF-initiated and 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 author of a PP, PP-Module, functional package or ST, 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.

PP, PP-Module, functional package or ST authors consider whether FTP\_TRP.1 (Trusted path) ([18.5.6](#)) should be included. In that case, the function "session locking" shall be included in the assignment of FTP\_TRP.1.3 as a service.

**17.5.6 FTA\_SSL.1 TSF-initiated session locking****17.5.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FIA\_UAU.1 (Timing of authentication) ([12.7.6](#))

**17.5.6.2 Component rationale**

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 are not required to be static (i.e. "screen savers" are permitted).

This component allows the author of a PP, PP-Module, functional package or ST to specify what events will unlock the session. These events may be related to the terminal, the user, or time.

EXAMPLE Examples of events include:

- terminal related: a fixed set of keystrokes to unlock the session;

**ISO/IEC 15408-2:2026(en)**

- user related: reauthentication;
- time related: after 15 min.

**17.5.6.3 Element FTA\_SSL.1.1**

**The TSF shall lock an interactive session after [assignment <sup>(a1)</sup>: *time interval of user inactivity*] by:**

- **clearing or overwriting display devices, making the current contents unreadable;**
- **disabling any activity of the user's data access/display devices other than unlocking the session.**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the interval of user inactivity that will trigger the locking of an interactive session. If requested, the author of a PP, PP-Module, functional package or ST can, through the assignment, specify that the time interval is left to the authorized administrator or the user. The management functions in FMT (Security management) can specify the capability to modify this time interval, making it the default value.

**17.5.6.4 Element FTA\_SSL.1.2**

**The TSF shall require the following events to occur prior to unlocking the session: [assignment <sup>(a1)</sup>: *events to occur*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the event(s) that should occur before the session is unlocked.

**17.5.7 FTA\_SSL.2 User-initiated locking****17.5.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FIA\_UID.1 (Timing of identification) ([12.8.6](#))

**17.5.7.2 Component rationale**

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 are not required to be static (i.e. "screen savers" are permitted).

**17.5.7.3 Element FTA\_SSL.2.1**

**The TSF shall allow user-initiated locking of the user's own interactive session, by:**

- **clearing or overwriting display devices, making the current contents unreadable;**
- **disabling any activity of the user's data access/display devices other than unlocking the session.**

**17.5.7.4 Element FTA\_SSL.2.2**

**The TSF shall require the following events to occur prior to unlocking the session: [assignment <sup>(a1)</sup>: *events to occur*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the event(s) that shall occur before the session is unlocked.

**ISO/IEC 15408-2:2026(en)****17.5.8 FTA\_SSL.3 TSF-initiated termination****17.5.8.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FMT\_SMR.1 (Security roles) ([13.10.6](#))

**17.5.8.2 Component rationale**

FTA\_SSL.3 (TSF-initiated termination) requires that the TSF terminate an interactive user session after a period of inactivity.

The author of a PP, PP-Module, functional package or ST should be aware that a session may continue after the user terminated his/her activity. This requirement would terminate this background subject after a period of inactivity of the user without regard to the status of the subject.

EXAMPLE An example of a continuing session after a user terminated activity is background processing.

**17.5.8.3 Element FTA\_SSL.3.1**

**The TSF shall terminate an interactive session after a [assignment (a1): *time interval of user inactivity*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the interval of user inactivity that will trigger the termination of an interactive session. If requested, the author of a PP, PP-Module, functional package or ST can, through the assignment, specify that the interval is left to the authorized administrator or the user. The management functions in FMT (Security management) can specify the capability to modify this time interval, making it the default value.

**17.5.9 FTA\_SSL.4 User-initiated termination****17.5.9.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**17.5.9.2 Component rationale**

FTA\_SSL.4 (User-initiated termination), provides the capability for an authorized user to terminate his/her interactive session.

The author of a PP, PP-Module, functional package or ST should be aware that a session can continue after the user terminated his/her activity.

EXAMPLE An example of a continuing session after a user terminated activity is background processing.

The requirement for user-initiated session termination allows the user to terminate a background process regardless of its status.

**17.5.9.3 Element FTA\_SSL.4.1**

**The TSF shall allow user-initiated termination of the user's own interactive session.**

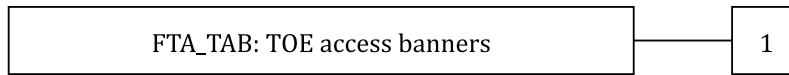
## 17.6 TOE access banners (FTA\_TAB)

### 17.6.1 Family Behaviour

This family defines requirements to display a configurable advisory warning message to users regarding the appropriate use of the TOE.

### 17.6.2 Component levelling and description

[Figure 86](#) illustrates the component levelling of family FTA\_TAB.



#### Key

1 component

**Figure 86 — FTA\_TAB: Component levelling**

FTA\_TAB.1 (Default TOE access banners) ([17.6.6](#)), provides the requirement for a TOE Access Banner. This banner is displayed prior to the establishment dialogue for a session.

### 17.6.3 Component management

The following action related to FTA\_TAB.1 can be considered for the management functions in FMT (Security management):

- maintenance of the banner by the authorized administrator.

### 17.6.4 Component audit

There are no auditable events foreseen related to FTA\_TAB.1.

### 17.6.5 Application notes

Prior to identification and authentication, TOE access requirements provide the ability for the TOE to display an advisory warning message to potential users pertaining to appropriate use of the TOE.

### 17.6.6 FTA\_TAB.1 Default TOE access banners

#### 17.6.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 17.6.6.2 Component rationale

This component requires that there is an advisory warning regarding the unauthorized use of the TOE. A PP, PP-Module, functional package or ST author can refine the requirement to include a default banner.

#### 17.6.6.3 Element FTA\_TAB.1.1

**Before establishing a user session, the [selection <sup>(s1)</sup>: *TSF, TOE platform*] shall display an [assignment <sup>(a1)</sup>: *description of the message*] message.**

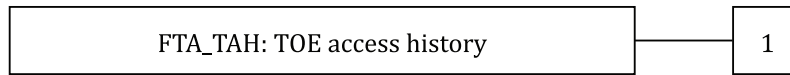
## 17.7 TOE access history (FTA\_TAH)

### 17.7.1 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.

### 17.7.2 Component levelling and description

[Figure 87](#) illustrates the component levelling of family FTA\_TAH.



#### Key

1 component

**Figure 87 — FTA\_TAH: Component levelling**

FTA\_TAH.1 (TOE access history) ([17.7.6](#)), provides the requirement for a TOE to display information related to previous attempts to establish a session.

### 17.7.3 Component management

There are no management activities foreseen related to FTA\_TAH.1.

### 17.7.4 Component audit

There are no auditable events foreseen related to FTA\_TAH.1.

### 17.7.5 Application notes

This family defines requirements for the TSF to display to users, upon successful session establishment to the TOE, a history of unsuccessful attempts to access the account. This history can include the date, time, means of access, and port of the last successful access to the TOE, as well as the number of unsuccessful attempts to access the TOE since the last successful access by the identified user.

### 17.7.6 FTA\_TAH.1 TOE access history

#### 17.7.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 17.7.6.2 Component rationale

This family can provide authorized users with information that can indicate the possible misuse of their user account.

This component requests that the user is presented with the information. The user should be able to review the information but is not forced to do so.

**EXAMPLE** A user can create scripts that ignore this information and start other processes.

### 17.7.6.3 Element FTA\_TAH.1.1

**Upon successful session establishment, the TSF shall display the [selection <sup>(s1)</sup>: *date, time, method, location*] of the last successful session establishment to the user.**

In selection (s1), the author of a PP, PP-Module, functional package or ST selects the security attributes of the last successful session establishment that will be shown at the user interface. The items are: Date, time, method of access, and/or location.

### 17.7.6.4 Element FTA\_TAH.1.2

**Upon successful session establishment, the TSF shall display the [selection <sup>(s1)</sup>: *date, time, method, location*] of the last unsuccessful attempt to session establishment and the number of unsuccessful attempts since the last successful session establishment.**

In selection (s1), the author of a PP, PP-Module, functional package or ST selects the security attributes of the last unsuccessful session establishment that will be shown at the user interface. The items are: Date, time, method of access, and/or location.

### 17.7.6.5 Element FTA\_TAH.1.3

**The TSF shall not erase the access history information from the user interface without giving the user an opportunity to review the information.**

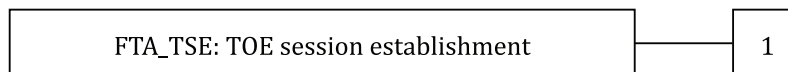
## 17.8 TOE session establishment (FTA\_TSE)

### 17.8.1 Family Behaviour

This family defines requirements to deny a user permission to establish a session with the TOE.

### 17.8.2 Component levelling and description

[Figure 88](#) illustrates the component levelling of family FTA\_TSE.



#### Key

1 component

**Figure 88 — FTA\_TSE: Component levelling**

FTA\_TSE.1 (TOE session establishment) ([17.8.6](#)), provides requirements for denying users access to the TOE based on attributes.

### 17.8.3 Component management

The following action related to FTA\_TSE.1 can be considered for the management functions in FMT (Security management):

- management of the session establishment conditions by the authorized administrator.

### 17.8.4 Component audit

The following actions related to FTA\_TSE.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: denial of a session establishment due to the session establishment mechanism;

## ISO/IEC 15408-2:2026(en)

- basic: all attempts at establishment of a user session;
- detailed: capture of the value of the selected access parameters.

### 17.8.5 Application notes

This family defines requirements to deny a user permission to establish a session with the TOE based on attributes such as the location or port of access, the user's security attribute, ranges of time or combinations of parameters.

EXAMPLE 1 Security attribute can include identity, clearance level, integrity level, membership in a role.

Ranges of time can include specific time-of-day, day-of-week, calendar dates.

This family provides the capability for the author of a PP, PP-Module, functional package or ST to specify requirements for the TOE to place constraints on the ability of an authorized user to establish a session with the TOE. The identification of relevant constraints can be achieved through the use of the selection operation.

EXAMPLE 2 Examples of attributes that can be used to specify the session establishment constraints are:

- 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.
- 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;
  - 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.
- 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 can be based upon time-of-day, day-of-week, or calendar dates. This constraint provides some operational protection against actions that can occur at a time where proper monitoring or where proper procedural measures might not be in place.

### 17.8.6 FTA\_TSE.1 TOE session establishment

#### 17.8.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 17.8.6.2 Element FTA\_TSE.1.1

**The TSF shall be able to deny session establishment based on [assignment <sup>(a1)</sup>: *attributes*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the attributes that can be used to restrict the 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).

## 18 Class FTP Trusted path/channels

### 18.1 Introduction

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:

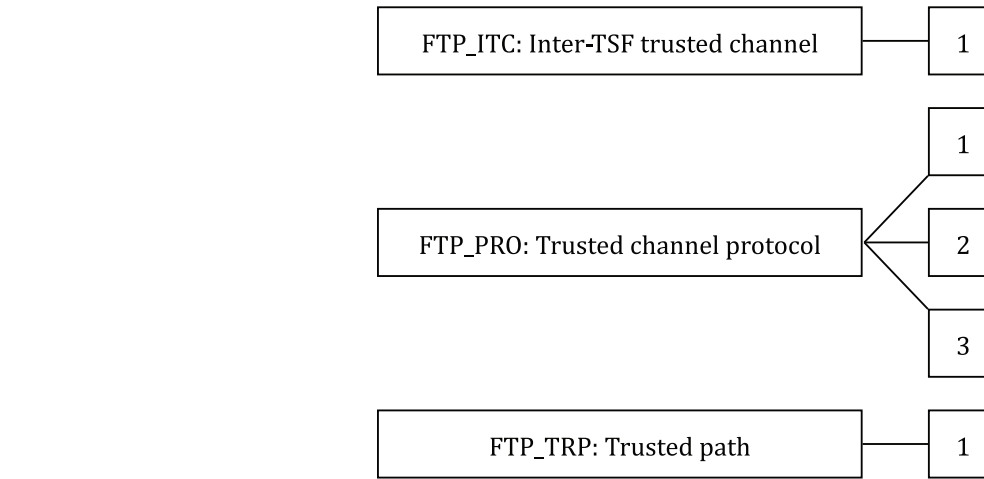
- the communications path is constructed using internal and external communications channels (as appropriate for the component) that isolate an identified subset of TSF data and commands from the remainder of the TSF and user data;
- use of the communications path can be initiated by the user and/or 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 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 can also be desired at other times during a user's session. Trusted path exchanges 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 89](#) illustrates the family hierarchy of class FTP.



**Key**  
 1,2,3 components

**Figure 89 — FTP: Trusted path/channels, class decomposition**

Direct, indirect and optional dependencies for the components of class FTP are shown in [Table 13](#).

**ISO/IEC 15408-2:2026(en)**

**Table 13 — Dependency table for class FTP: Trusted path/channels**

	FCS_CKM.1 (10.3.7)	FCS_CKM.2 (10.3.8)	FCS_CKM.5 (10.3.10)	FCS_COP.1 (10.4.6)	FTP_PRO.1 (18.4.6)	FTP_PRO.2 (18.4.7)	FTP_PRO.3 (18.4.8)
FTP_ITC.1 (18.3.6)							
FTP_PRO.1 (18.4.6)	-	-	-	-	-	X	X
FTP_PRO.2 (18.4.7)	O <sup>1</sup>	O <sup>1</sup>	X	X	X	-	-
FTP_PRO.3 (18.4.8)	-	-	-	X	X	X	-
FTP_TRP.1 (18.5.6)							

**18.2 Notes on class FTP**

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 can 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 can output erroneous information on an unsuspecting user's display, resulting in subsequent user actions that can be erroneous and can lead to a security breach.

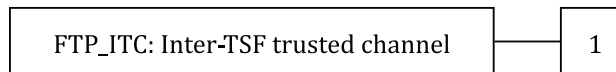
**18.3 Inter-TSF trusted channel (FTP\_ITC)**

**18.3.1 Family Behaviour**

This family defines requirements for the creation of a trusted channel between the TSF and other trusted IT products for the performance of security critical operations. The components of this family may be included whenever there are requirements for the secure communication of user or TSF data between the TOE and other trusted IT products.

**18.3.2 Component levelling and description**

Figure 90 illustrates the component levelling of family FTP\_ITC.



**Key**

1 component

**Figure 90 — FTP\_ITC: Component levelling**

FTP\_ITC.1 (Inter-TSF trusted channel) (18.3.6) requires that the TSF provide a trusted communication channel between itself and another trusted IT product.

**ISO/IEC 15408-2:2026(en)****18.3.3 Component management**

The following action related to FTP\_ITC.1 can be considered for the management functions in FMT (Security management):

- configuring the actions that require trusted channel, if supported.

**18.3.4 Component audit**

The following actions related to FTP\_ITC.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: failure of the trusted channel functions;
- minimal: identification of the initiator and target of failed trusted channel functions;
- basic: all attempted uses of the trusted channel functions;
- basic: identification of the initiator and target of all trusted channel functions.

**18.3.5 Application 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.

**18.3.6 FTP\_ITC.1 Inter-TSF trusted channel****18.3.6.1 Component relationships**

Hierarchical to: No other components.

Dependencies: No dependencies.

**18.3.6.2 Component rationale**

This component is used when a trusted communication channel between the TSF and another trusted IT product is required.

**18.3.6.3 Element FTP\_ITC.1.1**

**The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.**

**18.3.6.4 Element FTP\_ITC.1.2**

**The TSF shall permit [selection (s1): *the TSF, another trusted IT product*] to initiate communication via the trusted channel.**

In selection (s1), the author of a PP, PP-Module, functional package or ST specifies whether the local TSF, another trusted IT product, or both shall have the capability to initiate the trusted channel.

**18.3.6.5 Element FTP\_ITC.1.3**

**The TSF shall initiate communication via the trusted channel for [assignment (a1): *list of functions for which a trusted channel is required*].**

## ISO/IEC 15408-2:2026(en)

In assignment (a1), the author of a PP, PP-Module, functional package or ST specifies the functions for which a trusted channel is required.

**EXAMPLE** Examples of these functions can include transfer of user, subject, and/or object security attributes and ensuring consistency of TSF data.

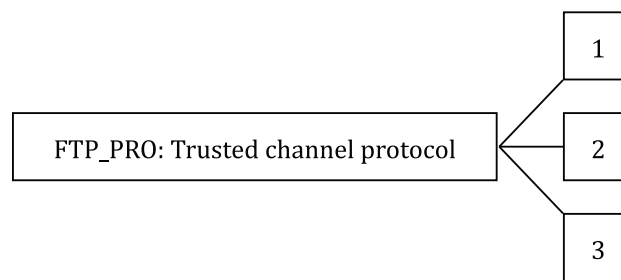
### 18.4 Trusted channel protocol (FTP\_PRO)

#### 18.4.1 Family Behaviour

This family defines requirements for establishing a trusted channel and using the trusted channel to transfer the TSF data or user data securely.

#### 18.4.2 Component levelling and description

[Figure 91](#) illustrates the component levelling of family FTP\_PRO.



**Key**

1,2,3 components

**Figure 91 — FTP\_PRO: Component levelling**

FTP\_PRO.1 (Trusted channel protocol) ([18.4.6](#)) requires that communication be established in accordance with a defined protocol.

FTP\_PRO.2 (Trusted channel establishment) ([18.4.7](#)) requires that keys be securely established between the peers.

FTP\_PRO.3 (Trusted channel data protection) ([18.4.8](#)) requires that data in transit be protected.

#### 18.4.3 Component management

The following actions related to FTP\_PRO.1 can be considered for the management functions in FMT (Security management):

- configuring the protocols needed for the trusted channel;
- configuring the credentials for using the trusted channel;
- configuring the conditions for initializing and terminating the trusted channel.

The following actions related to FTP\_PRO.2 can be considered for the management functions in FMT (Security management):

- configuring the parameters for shared secrets;
- configuring the parameters for cryptographic key derivation.

The following action related to FTP\_PRO.3 can be considered for the management functions in FMT (Security management):

- configuring the encryption and integrity mechanisms used by the trusted channel.

**ISO/IEC 15408-2:2026(en)****18.4.4 Component audit**

The following actions related to FTP\_PRO.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: failure of the trusted channel establishment;
- minimal: identification of the initiator and target of failed trusted channel establishment;
- basic: all attempted uses of the trusted channel;
- basic: identification of the initiator and target of all trusted channel attempts;
- basic: other events should be considered according to the specific protocols used.

The following actions related to FTP\_PRO.2 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: authentication failures during channel establishment;
- basic: all authentication attempts.

The following actions related to FTP\_PRO.3 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: failures when attempting to verify channel properties in FTP\_PRO.3.2.

**18.4.5 Application 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 protection of data transfers. In contrast with FTP\_ITC (Inter-TSF trusted channel) or FTP\_TRP (Trusted path) (18.5), FTP\_PRO (Trusted channel protocol) is concerned with security details of the protocol used for a channel and provides a focus for protocol properties that can otherwise be split between a larger number of separate SFRs. It can improve clarity of a PP, PP-Module, functional package or ST by highlighting mechanisms within the protocol that may then be linked to cryptographic functions described in other SFRs (such as FCS\_COP.1 (Cryptographic operation) (10.4.6)).

The components of FTP\_PRO (Trusted channel protocol) are not hierarchical but are intended to be used together to separately specify different aspects of a trusted channel, such as its confidentiality and integrity protection features.

There is no dependency from FTP\_PRO.2 (Trusted channel establishment) (18.4.7) to FTP\_PRO.3 (Trusted channel data protection) (18.4.8) because any mechanisms for security of the shared secret establishment will be part of the mechanism described in FTP\_PRO.2 (Trusted channel establishment) (18.4.7) itself.

In cases where some cryptographic operations used in the trusted channel protocol are performed outside the TOE, FTP\_PRO.2 (Trusted channel establishment) (18.4.7) and/or FTP\_PRO.3 (Trusted channel data protection) (18.4.8) can be omitted from a PP, PP-Module, functional package or ST, and the ST author would then need to supply a rationale for the unsatisfied dependencies between FTP\_PRO (Trusted channel protocol) components.

Separate iterations of the relevant FTP\_PRO (Trusted channel protocol) components may be used for different channels where the completion of the SFR is different for each channel. In general, each separate iteration will need to include all three components with its own dependencies' rationale.

## ISO/IEC 15408-2:2026(en)

### 18.4.6 FTP\_PRO.1 Trusted channel protocol

#### 18.4.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: FTP\_PRO.2 (Trusted channel establishment) ([18.4.7](#))  
FTP\_PRO.3 (Trusted channel data protection) ([18.4.8](#))

#### 18.4.6.2 Component rationale

Where values used in the completion of FTP\_PRO (Trusted channel protocol) operations have dependencies between different SFR elements, these need to be made clear in the instantiation of the SFR.

NOTE A table can be given in which the columns represent the relevant selections and assignments, and the rows define the valid combination of completion values.

#### 18.4.6.3 Element FTP\_PRO.1.1

**The TSF shall implement [assignment <sup>(a1)</sup>: *trusted channel protocol*] acting as [assignment <sup>(a2)</sup>: *defined protocol role(s)*] in accordance with: [assignment <sup>(a3)</sup>: *list of standards*].**

In assignment (a1), and (a2), if selected, the author of a PP, PP-Module, functional package or ST should specify a trusted channel protocol and the defined protocol roles.

EXAMPLE Examples of 'defined protocol roles' include 'client' or 'server' in a secure communication protocol, 'initiator' or 'responder' in key exchange mechanisms, 'Trust Center' in wireless security frameworks, or 'Key Distribution Centre' in authentication services.

#### 18.4.6.4 Element FTP\_PRO.1.2

**The TSF shall enforce usage of the trusted channel for [assignment <sup>(a1)</sup>: *purpose(s) of the trusted channel*] in accordance with: [assignment <sup>(a2)</sup>: *list of standards*].**

In assignment (a1), and (a2), the first assignment is intended to state rules for when the trusted channel is required to be used by the TOE, such as mandating its use for communications with an audit server. This assignment may take the value "None specified" (also with "None specified" in the second assignment) if no specific uses of the channel are mandated for the TOE.

#### 18.4.6.5 Element FTP\_PRO.1.3

**The TSF shall permit [selection <sup>(s1)</sup>: *itself, its peer*] to initiate communication via the trusted channel.**

In selection (s1), the author of a PP, PP-Module, functional package or ST selects which entity is allowed to initiate the establishment of the trusted channel.

#### 18.4.6.6 Element FTP\_PRO.1.4

**The TSF shall enforce the following rules for the trusted channel: [assignment <sup>(a1)</sup>: *rules governing operation and use of the trusted channel and/or its protocol*].**

#### 18.4.6.7 Element FTP\_PRO.1.5

**The TSF shall enforce the following static protocol options: [assignment <sup>(a1)</sup>: *list of options and references to standards in which each is defined*].**

In assignment (a1), the assignment is intended to state rules related to implementation of the protocol(s). It may take the value "None specified" if no rules are required, or if the standards referenced in other elements

**ISO/IEC 15408-2:2026(en)**

of the SFR include the relevant rules and no specific evaluator check is required for the context in which the SFR is being used.

EXAMPLE Rules include those for maximum packet sizes or rekey intervals

**18.4.6.8 Element FTP\_PRO.1.6**

**The TSF shall negotiate one of the following protocol configurations with its peer: [assignment (a1): list of configurations and reference to standards in which each is defined].**

In assignment (a1), the assignment is intended to state rules related to negotiable aspects of the protocol, when intending to narrow the options provided by the TOE compared to the standard that defines the protocol.

EXAMPLE Specification of acceptable older protocol versions.

**18.4.7 FTP\_PRO.2 Trusted channel establishment****18.4.7.1 Component relationships**

Hierarchical to: No other components.

Dependencies: FTP\_PRO.1 (Trusted channel protocol) ([18.4.6](#))  
[FCS\_CKM.1 (Cryptographic key generation) ([10.3.7](#)), or  
FCS\_CKM.2 (Cryptographic key distribution) ([10.3.8](#))  
FCS\_CKM.5 (Cryptographic key derivation) ([10.3.10](#))  
FCS\_COP.1 (Cryptographic operation) ([10.4.6](#))

**18.4.7.2 Component rationale**

In FTP\_PRO.2 (Trusted channel establishment), the "list of rules for carrying out the authentication" may be used to limit available parameters for the authentication mechanisms.

EXAMPLE Rules can be stated for the format (e.g. FQDN or IP address, use of wildcards) or prioritization of identifiers when alternative sources of an identifier are available in the authentication data exchanged.

**18.4.7.3 Element FTP\_PRO.2.1**

**The TSF shall establish a shared secret with its peer using one of the following mechanisms: [assignment (a1): list of key establishment mechanisms].**

In assignment (a1), The author of a PP, PP-Module, functional package or ST provides a list of key establishment mechanisms.

**18.4.7.4 Element FTP\_PRO.2.2**

**The TSF shall authenticate [selection (s1): its peer, itself to its peer] using one of the following mechanisms: [assignment (a1): list of authentication mechanisms] and according to the following rules: [assignment (a2): list of rules for carrying out the authentication].**

In selection (s1), the selection indicating the direction of the authentication should be chosen.

In assignment (a1), and (a2), the assignments include providing a list of authentication mechanisms used during the authentication and a list of rules used during the authentication.

**18.4.7.5 Element FTP\_PRO.2.3**

**The TSF shall use [assignment (a1): key derivation function] to derive the following cryptographic keys from a shared secret: [assignment (a2): list of cryptographic keys].**

## ISO/IEC 15408-2:2026(en)

### 18.4.8 FTP\_PRO.3 Trusted channel data protection

#### 18.4.8.1 Component relationships

Hierarchical to: No other components.

Dependencies: FTP\_PRO.1 (Trusted channel protocol) ([18.4.6](#))  
FTP\_PRO.2 (Trusted channel establishment) ([18.4.7](#))  
FCS\_COP.1 (Cryptographic operation) ([10.4.6](#))

#### 18.4.8.2 Component rationale

The FTP\_PRO.3 (Trusted channel data protection) component addresses protection (confidentiality and integrity) of data in transit through a trusted channel.

#### 18.4.8.3 Element FTP\_PRO.3.1

**The TSF shall protect data in transit from unauthorised disclosure using one of the following mechanisms: [assignment <sup>(a1)</sup>: *list of encryption mechanisms*].**

In assignment (a1), the author of a PP, PP-Module, functional package or ST completes the assignment by specifying lists of encryption and integrity protection mechanisms.

EXAMPLE Examples of integrity protection mechanism include protection of contents and file-system permissions of system files and directories; protection of processes against code injection, and protection against unsigned kernel extensions.

#### 18.4.8.4 Element FTP\_PRO.3.2

**The TSF shall protect data in transit from [selection <sup>(s1)</sup>: *modification, deletion, insertion, replay, [assignment <sup>(a1)</sup>: other]*] using one of the following mechanisms: [assignment <sup>(a2)</sup>: *list of integrity protection mechanisms*].**

In assignment (a1), and (a2), the author of a PP, PP-Module, functional package or ST completes the assignment by specifying lists of encryption and integrity protection mechanisms.

In selection (s1), the author of a PP, PP-Module, functional package or ST selects the threats that are mitigated by the TSF.

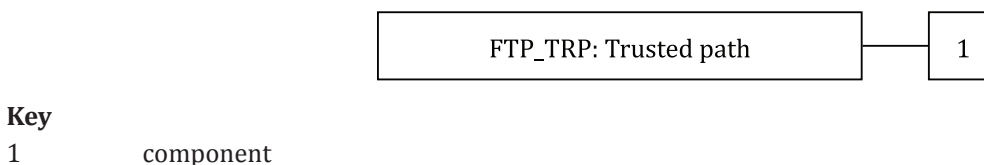
### 18.5 Trusted path (FTP\_TRP)

#### 18.5.1 Family Behaviour

This family defines the requirements to establish and maintain trusted communication to or from users and the TSF. A trusted path can be required for any security-relevant interaction. Trusted path exchanges can be initiated by a user during an interaction with the TSF, or the TSF can establish communication with the user via a trusted path.

#### 18.5.2 Component levelling and description

[Figure 92](#) illustrates the component levelling of family FTP\_TRP.



**Figure 92 — FTP\_TRP: Component levelling**

## ISO/IEC 15408-2:2026(en)

FTP\_TRP.1 (Trusted path) (18.5.6) requires that a trusted path between the TSF and a user be provided for a set of events defined by a PP, PP-Module, functional package or ST author. The user and/or the TSF can have the ability to initiate the trusted path.

### 18.5.3 Component management

The following action related to FTP\_TRP.1 can be considered for the management functions in FMT (Security management):

- configuring the actions that require trusted path, if supported.

### 18.5.4 Component audit

The following actions related to FTP\_TRP.1 should be auditable if FAU\_GEN (Security audit data generation) is included in the PP, PP-Module, functional package or ST:

- minimal: failures of the trusted path functions;
- minimal: identification of the user associated with all trusted path failures, if available;
- basic: all attempted uses of the trusted path functions;
- basic: identification of the user associated with all trusted path invocations, if available.

### 18.5.5 Application 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.

### 18.5.6 FTP\_TRP.1 Trusted path

#### 18.5.6.1 Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

#### 18.5.6.2 Component rationale

This component is used when trusted communication between a user and the TSF is required, either for initial authentication purposes only or for additional specified user operations.

#### 18.5.6.3 Element FTP\_TRP.1.1

**The TSF shall provide a communication path between itself and [selection (s1): *remote, local*] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from [selection (s2): *modification, disclosure, [assignment (a1): other types of integrity or confidentiality violation]*].**

In selection (s1), the author of a PP, PP-Module, functional package or ST specifies whether the trusted path is intended to be extended to remote and/or local users.

In assignment (a1), if selected, the author of a PP, PP-Module, functional package or ST identifies any additional types of integrity or confidentiality violation against which the trusted path shall protect the data.

**ISO/IEC 15408-2:2026(en)**

In selection (s2), the author of a PP, PP-Module, functional package or ST specifies whether the trusted path shall protect the data from modification, disclosure, and/or other types of integrity or confidentiality violation.

**18.5.6.4 Element FTP\_TRP.1.2**

**The TSF shall permit [selection <sup>(s1)</sup>: *the TSF, local users, remote users*] to initiate communication via the trusted path.**

In selection (s1), the author of a PP, PP-Module, functional package or ST specifies whether the TSF, local users, and/or remote users are able to initiate the trusted path.

**18.5.6.5 Element FTP\_TRP.1.3**

**The TSF shall require the use of the trusted path for [selection <sup>(s1)</sup>: *initial user authentication, [assignment <sup>(a1)</sup>: other services for which trusted path is required]*].**

In assignment (a1), if selected, the author of a PP, PP-Module, functional package or ST identifies other services for which trusted path is required, if any.

In selection (s1), the author of a PP, PP-Module, functional package or ST specifies whether the trusted path is intended to be used for initial user authentication and/or for other specified services.

**ISO/IEC 15408-2:2026(en)****Bibliography**

- [1] AIS 31, *Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren. Version 3, May 15,2020*
- [2] ISO/IEC 9594-8, *Information technology — Open systems interconnection — Part 8: The Directory: Public-key and attribute certificate frameworks*
- [3] ISO/IEC 15408-4:2026, *Information security, cybersecurity and privacy protection — Evaluation criteria for IT security — Part 4: Framework for the specification of evaluation methods and activities*
- [4] ISO/IEC 15408-5:2026, *Information security — Evaluation criteria for IT security — Part 5: Pre-defined packages of security requirements*
- [5] ISO/IEC/TS 19249, *Information technology — Security techniques — Catalogue of architectural and design principles for secure products, systems and applications*
- [6] ISO/IEC/TS 19608, *Guidance for developing security and privacy functional requirements based on ISO/IEC 15408*
- [7] ISO/IEC 29100, *Information technology — Security techniques — Privacy framework*
- [8] ITU-T Recommendation X.509, *Information technology – Open systems interconnection – The directory: Public-key and attribute certificate frameworks.*
- [9] NIST SP 800-90A, *The National Institute of Standards and Technology (NIST). Special Publication 800-90A Recommendation for Random Number Generation Using Deterministic Random Bit Generators, June 2015*
- [10] NIST SP 800-90B, *The National Institute of Standards and Technology (NIST). Special Publication 800-90B Recommendation for the Entropy Sources Used for Random Bit Generation, January 2018*



**ICS 35.030**

Price based on 243 pages