



Common Criteria
for Information Technology
Security Evaluation

Part 2 : Security functional requirements

19 December 1997

Version 2.0 Draft

CCIB-97/082R

Foreword

The CC Project Sponsoring Organisations are pleased to provide this **version 2.0 draft** of the *Common Criteria for Information Technology Security Evaluation*. This version is to be used by CC Project Sponsoring Organisations for their internal review. It will also be made available for information purposes to ISO/IEC, JTC 1, SC27/WG3 experts via the NIST website (see below). As previously agreed with WG3, the Common Criteria Implementation Board (CCIB) will continue to develop this document though early April, 1998. **Version 2.0 pre-final** will be released at that time, made available to WG 3 experts via the NIST website, and will also be provided in paper form at the WG3 meeting in Stockholm, Sweden.

LEGAL NOTICE:

The following seven governmental organisations (collectively called “the CC Project Sponsoring Organisations”), as the joint holders of the copyright in the Common Criteria for Information Technology Security, Parts 1 through 3 (called “the CC”), hereby grant non-exclusive license to ISO/IEC to use the CC in the development of an International Standard. However, the CC Project Sponsoring Organisations retain the right to use, copy, distribute, or modify the CC as they see fit.

CANADA:

Communications Security Establishment
Criteria Coordinator
R2B IT Security Standards and Initiatives
P.O. Box 9703, Terminal
Ottawa, Canada K1G 3Z4
Tel: +1.613.991.7409, Fax: +1.613.991.7411
E-mail: criteria@cse-cst.gc.ca
WWW: <http://www.cse.dnd.ca/cse/english/cc.html>
FTP: <ftp://ftp.cse.dnd.ca/pub/criteria/CC1.0>

FRANCE:

Service Central de la Sécurité des Systèmes d'Information (SCSSI)
Centre de Certification de la Sécurité des Technologies de l'Information
18, rue du docteur Zamenhof
F-92131 Issy les Moulineaux
France
Tel: +33.1.41463784, Fax: +33.1.41463701
E-mail: ssi20@calva.net

GERMANY:

D R A F T

German Information Security Agency (GISA)
Bundesamt für Sicherheit in der Informationstechnik
Abteilung V
Postfach 20 03 63
D-53133 Bonn
Germany
Tel: +49.228.9582.300, Fax: +49.228.9582.427
E-mail: cc@bsi.de
WWW: <http://www.bsi.bund.de>

NETHERLANDS:

Netherlands National Communications Security Agency
P.O. Box 20061
NL 2500 EB The Hague
The Netherlands
Tel: +31.70.3485637, Fax: +31.70.3486503
E-mail: criteria@nlncsa.minbuza.nl
WWW: <http://www.tno.nl/instit/fel/refs/cc.html>

UNITED KINGDOM:

Communications-Electronics Security Group
Compusec Evaluation Methodology
P.O. Box 144
Cheltenham GL52 5UE
United Kingdom
Tel: +44.1242.221.491 ext. 4134, Fax: +44.1242.235.233
E-mail: criteria@cesg.gov.uk
WWW: <http://www.cesg.gov.uk/cchtml>
FTP: <ftp://ftp.itsec.gov.uk/pub/ccv1.0>

UNITED STATES - NIST:

National Institute of Standards and Technology
Computer Security Division
820 Diamond, MS: NN426
Gaithersburg, Maryland 20899
U.S.A.
Tel: +1.301.975.2934, Fax: +1.301.948.0279
E-mail: criteria@nist.gov
WWW: <http://csrc.nist.gov/cc>

UNITED STATES - NSA:

D R A F T

National Security Agency

Attn: V2, Common Criteria Technical Advisor
Fort George G. Meade, Maryland 20755-6740
U.S.A.

Tel: +1.410.859.4458, Fax: +1.410.684.7512

E-mail: common_criteria@radium.ncsc.mil

WWW: <http://www.radium.ncsc.mil/tpep/>

D R A F T

Table of Contents

	Chapter 1	
	Introduction	1
1.1	Scope	1
1.1.1	Extending and maintaining functional requirements	1
1.2	Organisation of Part 2	2
1.3	Functional requirements paradigm	2
	Chapter 2	
	Security functional components	9
2.1	Overview	9
2.1.1	Class structure	9
2.1.1.1	Class name	9
2.1.1.2	Class introduction	10
2.1.2	Family structure	10
2.1.2.1	Family name	10
2.1.2.2	Family behaviour	10
2.1.2.3	Component levelling	11
2.1.2.4	Management	11
2.1.2.5	Audit	11
2.1.3	Component structure	12
2.1.3.1	Component identification	12
2.1.3.2	Functional elements	12
2.1.3.3	Dependencies	13
2.1.4	Permitted functional component operations	13
2.1.4.1	Iteration	14
2.1.4.2	Assignment	14
2.1.4.3	Selection	14
2.1.4.4	Refinement	14
2.2	Component catalogue	15
2.2.1	Component changes highlighting	16
	Class FAU	
	Security Audit	17
FAU_ARP	Security Audit Automatic Response	19
	FAU_ARP.1 Security Alarms	19
FAU_GEN	Security Audit Data Generation	20
	FAU_GEN.1 Audit Data Generation	20
	FAU_GEN.2 User Identity Generation	21
FAU_SAA	Security Audit Analysis	22
	FAU_SAA.1 Imminent Violation Analysis	23
	FAU_SAA.2 Profile Based Anomaly Detection	24
	FAU_SAA.3 Simple Attack Heuristics	24
	FAU_SAA.4 Complex Attack Heuristics	24
FAU_SAR	Security Audit Review	26

D R A F T

	FAU_SAR.1	Audit Review	27
	FAU_SAR.2	Restricted Audit Review	27
	FAU_SAR.3	Selectable Audit Review	27
FAU_SEL		Security Audit Event Selection	28
	FAU_SEL.1	Selective Audit	28
FAU_STG		Security Audit Event Storage	29
	FAU_STG.1	Permanent Audit Trail Storage	30
	FAU_STG.2	Guarantees of Audit Data Availability	30
	FAU_STG.3	Action in Case of Possible Audit Data Loss	31
	FAU_STG.4	Prevention of Audit Data Loss	31
		Class FCO	
		Communication	33
FCO_NRO		Non-Repudiation of Origin	34
	FCO_NRO.1	Selective Proof of Origin	35
	FCO_NRO.2	Enforced Proof of Origin	35
FCO_NRR		Non-Repudiation of Receipt	36
	FCO_NRR.1	Selective Proof of Receipt	37
	FCO_NRR.2	Enforced Proof of Receipt	37
		Class FCS	
		Cryptographic Support	39
FCS_CKM		Cryptographic Key Management	40
	FCS_CKM.1	Cryptographic Key Generation	41
	FCS_CKM.2	Standards-Based Cryptographic Key Generation	41
	FCS_CKM.3	Cryptographic Key Distribution	42
	FCS_CKM.4	Standards-Based Cryptographic Key Distribution	42
	FCS_CKM.5	Cryptographic Key Access	42
	FCS_CKM.6	Standards-Based Cryptographic Key Access	43
	FCS_CKM.7	Cryptographic Key Destruction	43
	FCS_CKM.8	Standards-Based Cryptographic Key Destruction	43
FCS_COP		Cryptographic Operation	45
	FCS_COP.1	Cryptographic Operation	46
	FCS_COP.2	Standards-Based Cryptographic Operation	46
		Class FDP	
		User Data Protection	47
FDP_ACC		Access Control Policy	50
	FDP_ACC.1	Subset Access Control	50
	FDP_ACC.2	Complete Access Control	51
FDP_ACF		Access Control Functions	52
	FDP_ACF.1	Security Attribute Based Access Control	53
	FDP_ACF.2	Access Authorisation	53
	FDP_ACF.3	Access Authorisation and Denial	54
	FDP_ACF.4	Fixed Access Control	54
FDP_DAU		Data Authentication	55
	FDP_DAU.1	Basic Data Authentication	56

D R A F T

	FDP_DAU.2 Data Authentication with Identity of Guarantor	56
FDP_ETC	Export to Outside TSF Control	57
	FDP_ETC.1 Export of User Data Without Security Attributes	58
	FDP_ETC.2 Export of User Data With Security Attributes	58
FDP_IFC	Information Flow Control Policy	59
	FDP_IFC.1 Subset Information Flow Control	59
	FDP_IFC.2 Complete Information Flow Control	60
FDP_IFF	Information Flow Control Functions	61
	FDP_IFF.1 Simple Security Attributes	63
	FDP_IFF.2 Hierarchical Security Attributes	63
	FDP_IFF.3 Limited Illicit Information Flows	64
	FDP_IFF.4 Partial Elimination of Illicit Information Flows	65
	FDP_IFF.5 No Illicit Information Flows	65
	FDP_IFF.6 Illicit Information Flow Monitoring	65
	FDP_IFF.7 Information Flow Authorisation	65
	FDP_IFF.8 Information Flow Authorisation and Denial	66
FDP_ITC	Import from Outside TSF Control	67
	FDP_ITC.1 Import of User Data Without Security Attributes	68
	FDP_ITC.2 Import of User Data with Security Attributes	68
FDP_ITT	Internal TOE Transfer	69
	FDP_ITT.1 Basic Internal Transfer Protection	70
	FDP_ITT.2 Transmission Separation by Attribute	70
	FDP_ITT.3 Integrity Monitoring	71
	FDP_ITT.4 Attribute-Based Integrity Monitoring	71
FDP_RIP	Residual Information Protection	72
	FDP_RIP.1 Subset Residual Information Protection	72
	FDP_RIP.2 Full Residual Information Protection	73
FDP_ROL	Rollback	74
	FDP_ROL.1 Basic Rollback	75
	FDP_ROL.2 Advanced Rollback	75
FDP_SDI	Stored Data Integrity	76
	FDP_SDI.1 Stored Data Integrity Monitoring	77
	FDP_SDI.2 Stored Data Integrity Monitoring and Action	77
FDP_UCT	Inter-TSF User Data Confidentiality Transfer Protection	78
	FDP_UCT.1 Basic Data Exchange Confidentiality	78
FDP_UIT	Inter-TSF User Data Integrity Transfer Protection	79
	FDP_UIT.1 Data Exchange Integrity	80
	FDP_UIT.2 Source Data Exchange Recovery	80
	FDP_UIT.3 Destination Data Exchange Recovery	81
	 Class FIA	
	Identification and Authentication	83
FIA_AFL	Authentication Failures	85
	FIA_AFL.1 Basic Authentication Failure Handling	85
FIA_ATD	User Attribute Definition	87
	FIA_ATD.1 User Attribute Definition	87
FIA_SOS	Specification of Secrets	88
	FIA_SOS.1 Verification of Secrets	88
	FIA_SOS.2 TSF Generation of Secrets	89

D R A F T

FIA_UAU	User Authentication	90
	FIA_UAU.1 Timing of authentication	92
	FIA_UAU.2 User authentication before any action	93
	FIA_UAU.3 Unforgeable Authentication	93
	FIA_UAU.4 Single-use Authentication Mechanisms	93
	FIA_UAU.5 Multiple Authentication Mechanisms	93
	FIA_UAU.6 Re-authenticating	94
	FIA_UAU.7 Protected authentication feedback	94
FIA_UID	User Identification	95
	FIA_UID.1 Timing of Identification	96
	FIA_UID.2 User Identification before any action	96
FIA_USB	User-Subject Binding	97
	FIA_USB.1 User-Subject Binding	97
	Class FMT	
	Security Management	99
FMT_MOF	Management of functions in TSF	102
	FMT_MOF.1 Management of security functions behaviour	102
FMT_MSA	Management of Security Attributes	103
	FMT_MSA.1 Management of security attributes	104
	FMT_MSA.2 Safe security attributes	104
	FMT_MSA.3 Static Attribute Initialisation	105
FMT_MTD	Management of TSF data	106
	FMT_MTD.1 Management of TSF data	107
	FMT_MTD.2 Management of limits on TSF data	107
	FMT_MTD.3 Safe TSF data	107
FMT_REV	Revocation	109
	FMT_REV.1 Revocation	109
FMT_SAE	Security Attribute Expiration	110
	FMT_SAE.1 Time-Limited Authorisation	110
FMT_SMR	Security Management Roles	111
	FMT_SMR.1 Security roles	112
	FMT_SMR.2 Restrictions on security roles	112
	FMT_SMR.3 Assuming roles	113
	Class FPR	
	Privacy	115
FPR_ANO	Anonymity	116
	FPR_ANO.1 Anonymity	116
	FPR_ANO.2 TSF Anonymity	117
FPR_PSE	Pseudonymity	118
	FPR_PSE.1 Pseudonymity	118
	FPR_PSE.2 Reversible Pseudonymity	119
	FPR_PSE.3 Alias Pseudonymity	119
FPR_UNL	Unlinkability	120
	FPR_UNL.1 Unlinkability	120
FPR_UNO	Unobservability	121
	FPR_UNO.1 Unobservability	121

D R A F T

FPR_UNO.2	Authorised Administrator Observability	122
Class FPT		
	Protection of the TOE Security Functions	123
FPT_AMT	Underlying Abstract Machine Test	126
	FPT_AMT.1 Abstract Machine Testing	126
FPT_FLS	Fail Secure	127
	FPT_FLS.1 Failure with Preservation of Secure State	127
FPT_ITA	Availability of exported TSF Data	128
	FPT_ITA.1 Inter-TSF Availability Within a Defined Availability Metric	128
FPT_ITC	Confidentiality of exported TSF Data	129
	FPT_ITC.1 Inter-TSF Confidentiality During Transmission	129
FPT_ITI	Integrity of exported TSF Data	130
	FPT_ITI.1 Inter-TSF Detection of Modification	131
	FPT_ITI.2 Inter-TSF Detection and Correction of Modification	131
FPT_ITT	Internal TOE TSF Data Transfer	132
	FPT_ITT.1 Basic Internal TSF Data Transfer Protection	133
	FPT_ITT.2 TSF Data Transfer Separation	133
	FPT_ITT.3 TSF Data Integrity Monitoring	134
FPT_PHP	TSF Physical Protection	135
	FPT_PHP.1 Passive Detection of Physical Attack	136
	FPT_PHP.2 Notification of Physical Attack	136
	FPT_PHP.3 Resistance to Physical Attack	137
FPT_RCV	Trusted Recovery	138
	FPT_RCV.1 Manual Recovery	139
	FPT_RCV.2 Automated Recovery	139
	FPT_RCV.3 Automated Recovery without Undue Loss	140
	FPT_RCV.4 Function Recovery	140
FPT_RPL	Replay Detection and Correction	141
	FPT_RPL.1 Replay Detection	141
FPT_RVM	Reference Mediation	142
	FPT_RVM.1 Non-Bypassability of the TSP	142
FPT_SEP	Domain Separation	144
	FPT_SEP.1 TSF Domain Separation	145
	FPT_SEP.2 SFP Domain Separation	145
	FPT_SEP.3 Complete Reference Monitor	145
FPT_SSP	State Synchrony Protocol	147
	FPT_SSP.1 Simple Trusted Acknowledgement	147
	FPT_SSP.2 Mutual Trusted Acknowledgement	148
FPT_STM	Time Stamps	149
	FPT_STM.1 Reliable Time Stamps	149
FPT_TDC	Inter-TSF TSF Data Consistency	150
	FPT_TDC.1 Inter-TSF Basic TSF Data Consistency	150
FPT_TRC	Internal TOE TSF Data Replication Consistency	151
	FPT_TRC.1 Internal TSF Consistency	151
FPT_TST	TSF Self Test	152
	FPT_TST.1 TSF Testing	153

D R A F T

	Class FRU	
	Resource Utilisation	155
FRU_FLT	Fault Tolerance	156
	FRU_FLT.1 Degraded Fault Tolerance	156
	FRU_FLT.2 Limited Fault Tolerance	157
FRU_PRS	Priority of Service	158
	FRU_PRS.1 Limited Priority of Service	158
	FRU_PRS.2 Full Priority of Service	159
FRU_RSA	Resource Allocation	160
	FRU_RSA.1 Maximum Quotas	161
	FRU_RSA.2 Minimum and Maximum Quotas	161
	Class FTA	
	TOE Access	163
FTA_LSA	Limitation on Scope of Selectable Attributes	164
	FTA_LSA.1 Limitation on Scope of Selectable Attributes	164
FTA_MCS	Limitation on Multiple Concurrent Sessions	165
	FTA_MCS.1 Basic Limitation on Multiple Concurrent Sessions	166
	FTA_MCS.2 Per User Attribute Limitation on Multiple Concurrent Sessions	166
FTA_SSL	Session Locking	167
	FTA_SSL.1 TSF-initiated Session Locking	168
	FTA_SSL.2 User-initiated Locking	168
	FTA_SSL.3 TSF-initiated Termination	168
FTA_TAB	TOE Access Banners	169
	FTA_TAB.1 Default TOE Access Banners	169
FTA_TAH	TOE Access History	170
	FTA_TAH.1 TOE Access History	170
FTA_TSE	TOE Session Establishment	171
	FTA_TSE.1 TOE Session Establishment	171
	Class FTP	
	Trusted Path/Channels	173
FTP_ITC	Inter-TSF Trusted Channel	174
	FTP_ITC.1 Inter-TSF Trusted Channel	174
FTP_TRP	Trusted Path	176
	FTP_TRP.1 Trusted Path	176

D R A F T

List of Figures

Figure 1.1 - Security functional requirements paradigm (Monolithic TOE)	3
Figure 1.2 - Diagram of security function requirements paradigm (Distributed TOE)	3
Figure 1.3 - Relationship Between User Data and TSF Data	7
Figure 1.4 - Relationship between “authentication data” and “secrets”.	8
Figure 2.1 - Functional class structure	9
Figure 2.2 - Functional family structure	10
Figure 2.3 - Functional component structure	12
Figure 2.4 - Sample class decomposition diagram	15
Figure 2.5 - Security Audit Class decomposition	18
Figure 2.6 - Communication class decomposition	33
Figure 2.7 - Cryptographic Support class decomposition	39
Figure 2.8 - User Data Protection class decomposition	48
Figure 2.9 - User Data Protection class decomposition (cont.)	49
Figure 2.10 - Identification and Authentication class decomposition	84
Figure 2.11 - Security Management class decomposition	100
Figure 2.12 - Privacy class decomposition	115
Figure 2.13 - Protection of the TOE Security Functions class decomposition	124
Figure 2.14 - Protection of the TOE Security Functions class decomposition (Cont.)	125
Figure 2.15 - Resource Utilisation class decomposition	155
Figure 2.16 - TOE Access class decomposition	163
Figure 2.17 - Trusted Path / Channels Class decomposition	173

D R A F T

D R A F T

List of Tables

Table 2.1 -	Mapping from CC 1.0 to 2.0 draft for security management requirements . . .	101
-------------	---	-----

D R A F T

Chapter 1

Introduction

1.1 Scope

- 1 Security functional components, as defined in this Part 2, are the basis for the TOE IT security functional requirements expressed in a Protection Profile (PP) or a Security Target (ST). These requirements describe the desired security behaviour expected of a Target of Evaluation (TOE) and are intended to meet the security objectives as stated in a PP or an ST. These requirements describe security properties that users can detect by direct interaction with the TOE (i.e. inputs, outputs) or by the TOE's response to stimulus.
- 2 Security functional components express security requirements intended to counter threats in the assumed operating environment of the TOE and/or cover any identified organisational security policies and assumptions.
- 3 The audience for Part 2 includes consumers, developers, and evaluators of secure IT systems and products. Part 1 chapter 3 provides additional information on the target audience of the Common Criteria (CC), and on the use of the CC by the groups that comprise the target audience. These groups may use Part 2 as follows:
 - Consumers may use Part 2 when selecting components to express functional requirements to satisfy the security objectives expressed in a PP or ST. Part 1 chapter 4.3 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, may find a standardised method to understand those requirements in this part. They can also use the contents of this part as a basis for further defining the TOE security functions and mechanisms that comply with those requirements.
 - Evaluators should use the functional requirements defined in this part in verifying that the TOE functional requirements expressed in the PP or ST satisfy the IT security objectives and that all dependencies are accounted for and shown to be satisfied. Evaluators also should use this part to assist in determining whether a given TOE satisfies stated requirements.

1.1.1 Extending and maintaining functional requirements

- 4 The CC and the associated security functional requirements described herein are not meant to be a definitive answer to all the problems of IT security. Rather, the CC offers a set of well understood security functional requirements which can be used to create trusted products or systems reflecting the needs of the market. These security functional requirements are presented as the current state of the art in requirements specification and evaluation.

5 This part does not presume to include all possible security functional requirements but rather contains those which are known and agreed to be of value by the CC sponsoring organisations at the time of release.

6 Since the understanding and needs of consumers may change, the functional requirements in this part will need to be maintained. It is envisioned that some PP/ST authors may have security needs not (yet) covered by the functional requirement components in the Common Criteria. In those cases the PP/ST author may consider using functional requirements not taken from the CC (extensibility), as explained in part 1 annexes B and C.

1.2 Organisation of Part 2

7 Chapter 1 is the introductory material for Part 2.

8 Chapter 2 is the catalogue of CC functional components.

9 Annex A provides additional information of interest to potential users of the functional components. It is a repository for informative supporting material for the users of this part, which may help them to apply relevant operations and select appropriate audit or documentation information.

10 Annex B provides the Common Criteria observation report guidance, example observations and an example printed form.

11 These annexes, A and B, are not included in this document but are contained in a separate, companion document.

12 Those who author PPs or STs should refer to Part 1 for relevant structures, rules, and guidance:

- Part 1, Chapter 2 defines the terms used in the CC.
- Part 1, Annex B defines the structure for PPs.
- Part 1, Annex C defines the structure for STs.

1.3 Functional requirements paradigm

13 This section describes the paradigm used in the security functional requirements of Part 2. Figures 1.1 and 1.2 depict some of the key concepts of the paradigm. This section provides descriptive text for those figures and for other key concepts not depicted. Key concepts discussed are highlighted in bold/italics. This section is not intended to replace or supersede any of the terms found in the CC glossary in Part 1, Chapter 2.

D R A F T

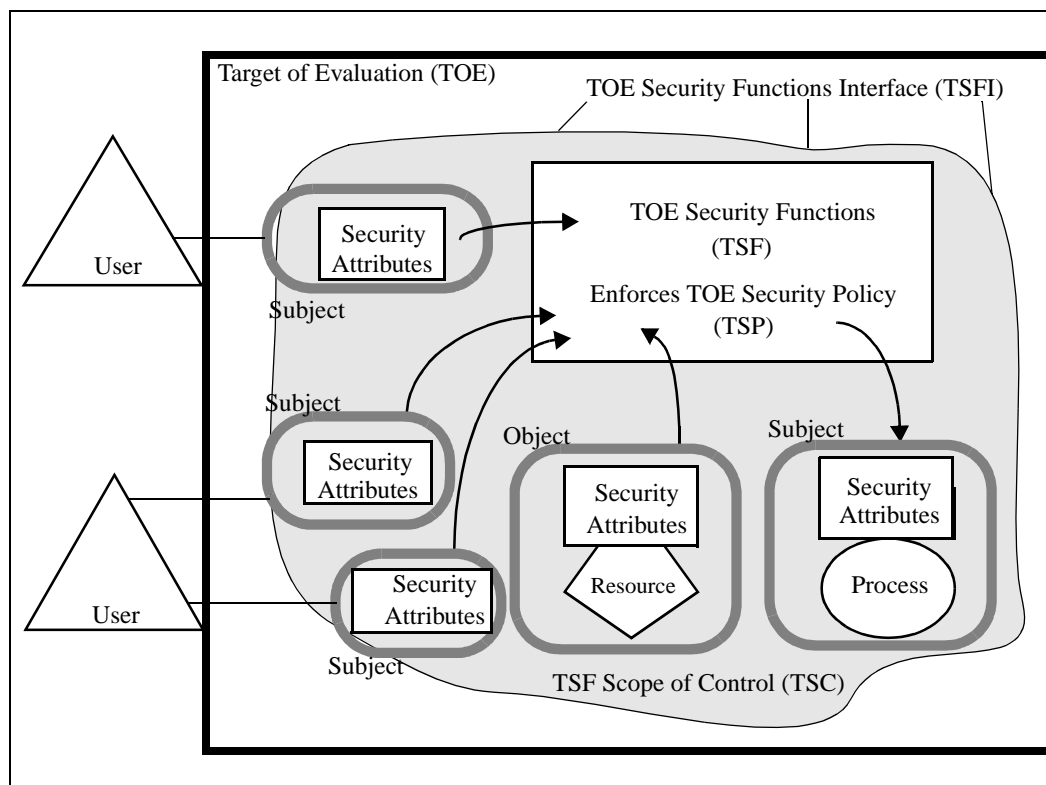
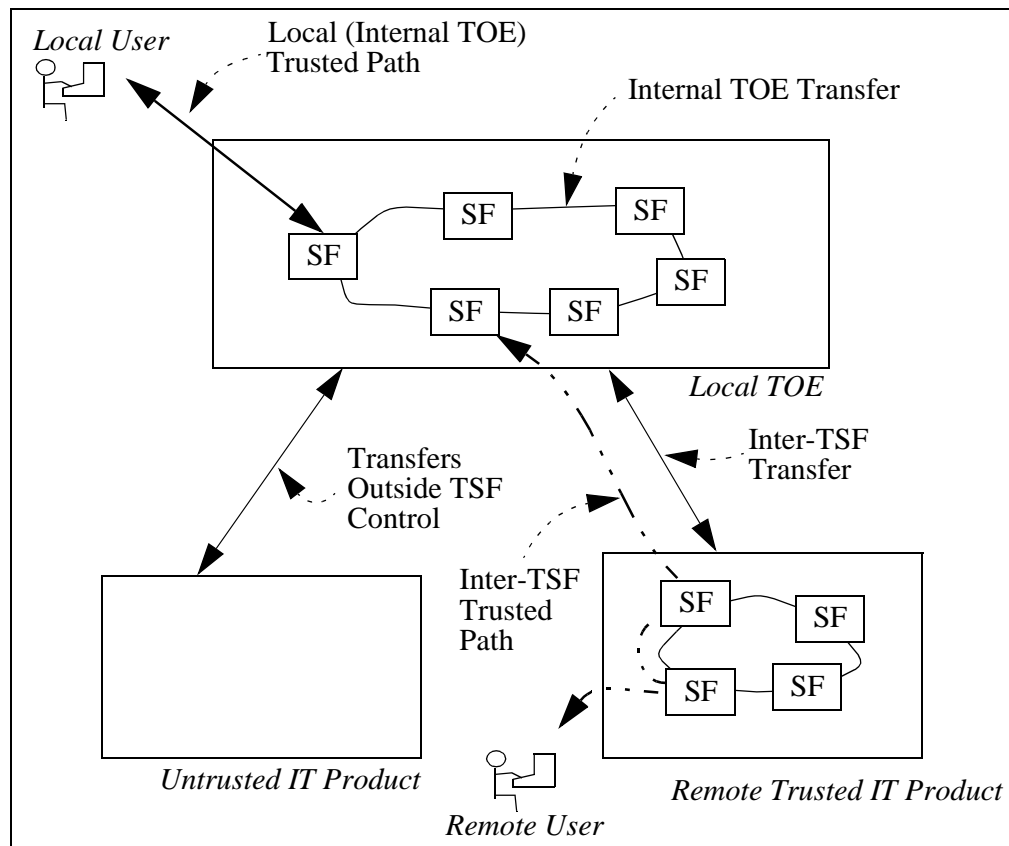


Figure 1.1 - Security functional requirements paradigm (Monolithic TOE)

Figure 1.2 - Diagram of security function requirements paradigm (Distributed TOE)

- 14 This part 2 is a catalogue of security functional requirements which can be specified for a **Target of Evaluation (TOE)**. A TOE is an IT product or system containing resources such as electronic storage media (e.g. disks), peripheral devices (e.g. printers), and computing capacity (e.g. CPU time) that can be used for processing and storing information and is the subject of an evaluation.
- 15 TOE evaluation is concerned primarily with ensuring that a defined **TOE Security Policy (TSP)** is enforced over the TOE resources. The TSP defines the rules by which the TOE governs access to its resources, and thus all information and services controlled by the TOE.
- 16 The TSP is, in turn, made up of multiple **Security Function Policies (SFPs)**. Each SFP has a scope of control, which defines the subjects, objects, and operations controlled under the SFP. The SFP is implemented by a **Security Function (SF)**, whose mechanisms enforce the policy and provide necessary capabilities.

D R A F T



- 17 There are two types of SFPs that apply to data protection: **access control SFPs** and **information flow SFPs**. The mechanisms that implement access control SFPs base their policy decisions on **attributes** of the subjects, objects, 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.
- 18 The mechanisms that implement information flow SFPs base their policy decisions on the attributes assigned to subjects and objects within the scope of control and the set of rules that govern the transfer of information between subjects and objects.
- 19 All the portions of a TOE which must be relied on for the correct enforcement of the TSP (i.e. this collection of multiple SFPs) is referred to as the **TOE Security Functions (TSF)**. The TSF consists of all hardware, software, and firmware of a TOE which either directly enforce or contribute to the enforcement of the TSP.
- 20 A **reference monitor** is an abstract machine that enforces the access control policies of a TOE. A **reference validation mechanism** is an implementation of the reference monitor concept that possesses the following properties: tamperproof, always invoked, and simple enough to be subjected to thorough analysis and testing. The **TSF** consists of a reference validation mechanism (potentially) and other functions necessary for the operation of the TOE.
- 21 The TOE may be a monolithic product containing hardware, firmware, and software, or it may consist internally of multiple physically-separated parts. Each

D R A F T

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.

- 22 When the TOE consists of multiple parts, each part of the TOE may have its own part of the TSF. When the TOE is viewed as a whole, the separate parts of the TSF abstractly form the composite TSF, which enforces the TSP. In order to do this, the parts of the TOE exchange user and TSF data over internal communication channels. This interaction is called *internal TOE transfer*.
- 23 TOE interfaces may be localised to the particular TOE, or they may allow interaction with other IT products over *external communication channels*. These external interactions with other IT products may take two forms:
- a) The TSPs of the 'Remote Trusted IT product' and local TOEs have been administratively coordinated and evaluated. Exchanges of information in this situation are called *inter-TSF transfers*, as they are between the TSFs of distinct trusted products.
 - b) The remote IT product may not have been evaluated, indicated in figure 1.2 as 'untrusted IT product', therefore its TSP is unknown. Exchanges of information in this situation are called *transfers outside TSF control*, as there is no TSF (or its policy characteristics are unknown) on the remote IT product.
- 24 The set of interactions which can occur with or within a TOE and are subject to the rules of the TSP is called the *TSF Scope of Control (TSC)*. The TSC encompasses a defined set of interactions based on subjects, objects, and operations within the TOE, but it need not encompass all resources of a TOE.
- 25 The set of interfaces, whether interactive (man-machine interface) or programmatic (application programming interface), through which resources are accessed which 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 functions which provide for the enforcement of the TSP.
- 26 Users are outside of the TOE, and therefore outside of the TSC. However, in order to request that services be performed by the TOE, users interact with the TOE through the TSFI. There are two types of users of interest to the Part 2 security functional requirements, *human users* and *machine users*. Human users are further differentiated as *local human users*, meaning they interact directly with the TOE via TOE devices (e.g. workstations), or *remote human users*, meaning they interact indirectly with the TOE through another TOE.
- 27 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 including, for example: user authentication, time of day, method of accessing the TOE, and number of allowed concurrent sessions per user.
- 28 Part 2 uses the term *authorised* to signify a user who possesses the rights and/or privileges necessary to perform an operation. The term *authorised user*, therefore,

indicates that it is allowable for a user to perform an operation as defined by the TSP.

- 29 The term *authorised administrator* is used to indicate a human user who is trusted to perform security critical operations within the TOE, such as setting TOE configuration parameters that may affect the enforcement of the TSP, and therefore possesses the specific rights necessary to perform those operations.

When the term “the authorised administrator” is used in the security functional components, it is referring specifically to an administrator authorised with respect to the SFP related to the functions in the component. When the more general phrase “authorised administrators” is used, it refers to administrators who are authorised for SFPs other than the one related to the functions in question.

- 30 To express requirements that call for the separation of administrator duties, the relevant Part 2 security functional components (from family FMT_SMF) explicitly state that administrative *roles* are required. A role is a pre-defined set of allowed authorisations that may be granted to a user. A TOE may support the definition of any number of roles. For example, roles related to the secure operation of a TOE may include “Audit Administrator” and “User Accounts Administrator”. Roles may also be defined specifically for the application environment in which the TOE will be used. For example, in a TOE used in a hospital, a “Doctor” role might be established for users who are authorised to prescribe medication. However, any user operating in a “Nurse” role may only be authorised to administer any such medication. Roles required by the components in Part 2 are security related.

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

- 32 TOE resources can be structured and utilised in many different ways. However, Part 2 makes a specific distinction which allows for the specification of desired security properties. All entities which can be created from resources can be characterised in one of two ways. The entities may be active, meaning that they are the cause of actions which occur internal to the TOE and cause operations to be performed on information. Alternatively, the entities may be passive, meaning that they are either the container from which information originates or to which information is stored.

- 33 Active entities are referred to as *subjects*. Several types of subjects may exist within a TOE:

- a) those acting on behalf of an authorised user and which are subject to all the rules of the TSP (e.g. UNIX processes);
- b) those acting as a specific functional process which may in turn act on behalf of multiple users (e.g. client/server architecture);
- c) those acting on behalf of authorised administrators; or
- d) those acting as part of the TOE itself (e.g. trusted processes).

DRAFT

- 34 Part 2 addresses the enforcement of the TSP over types of subjects as those listed above.
- 35 Passive entities (i.e. information containers) are referred to in the Part 2 security functional requirements as *objects*. Objects are the targets of operations that may be performed by subjects. In the case where a subject (an active entity) is the target of an operation (e.g. interprocess communication), a subject may also be acted on as an object.
- 36 Users, subjects and objects possess certain attributes which contain information that allows the TOE to behave correctly. Some attributes, such as file names, may be intended to be informational (i.e. to increase the user-friendliness of the TOE) while others, such as access control information, may exist specifically for the enforcement of the TSP. These latter attributes are generally referred to as security attributes. The word attribute shall be used as a shorthand in this part for the word security attribute. However, no matter what the intended purpose of the attribute information, it may be necessary to have controls on attributes as dictated by the TSP.
- 37 Data in a TOE is categorised as either user data or TSF data. Figure 1.3 depicts this relationship. *User Data* is information stored in TOE resources that can be operated upon by users in accordance with the TSP and upon which the TSF places no special meaning. For example, the contents of an electronic mail message is user data. *TSF Data* is information used by the TSF in making TSP decisions. TSF Data may be influenced by users if allowed by the TSP. Security attributes, authentication data and access control list entries are examples of TSF data.

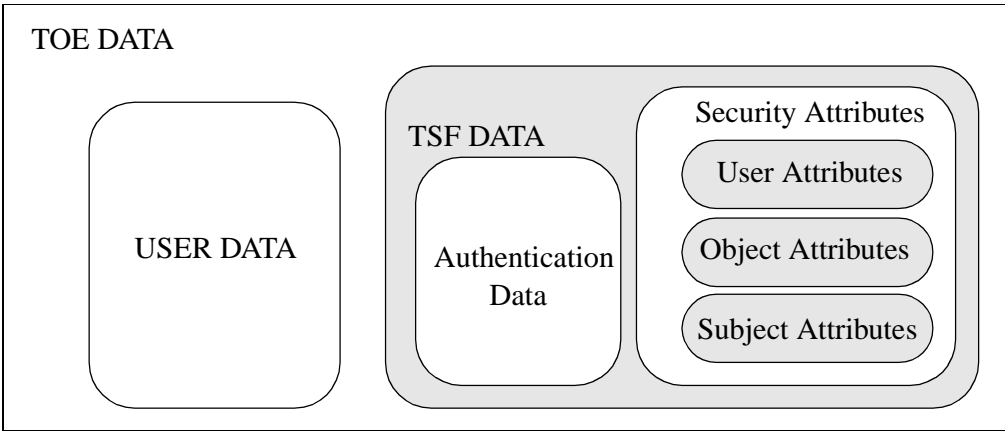


Figure 1.3 - Relationship Between User Data and TSF Data

- 38 Two specific types of TSF data addressed by Part 2 can be, but are not necessarily, the same. These are *authentication data* and *secrets*.
- 39 Authentication data is used to verify the claimed identity of a user requesting services from a TOE. The most common form of authentication data is the password, which depends on being kept secret in order to be an effective security mechanism. However, not all forms of authentication data need to be kept secret.

D R A F T

Biometric authentication devices (e.g. fingerprint readers, retinal scanners) 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.

40 The term secrets, as used in CC functional requirements, while applicable to authentication data, is intended to also be applicable to other types of data that must be kept secret in order to enforce a specific SFP. For example, a trusted channel mechanism that relies on cryptography to preserve the confidentiality of information being transmitted via the channel can only be as strong as the method used to keep the cryptographic keys secret from unauthorised disclosure.

41 Therefore, some, but not all, authentication data needs to be kept secret and some, but not all, secrets are used as authentication data. Figure 1.4 shows this relationship between secrets and authentication data.

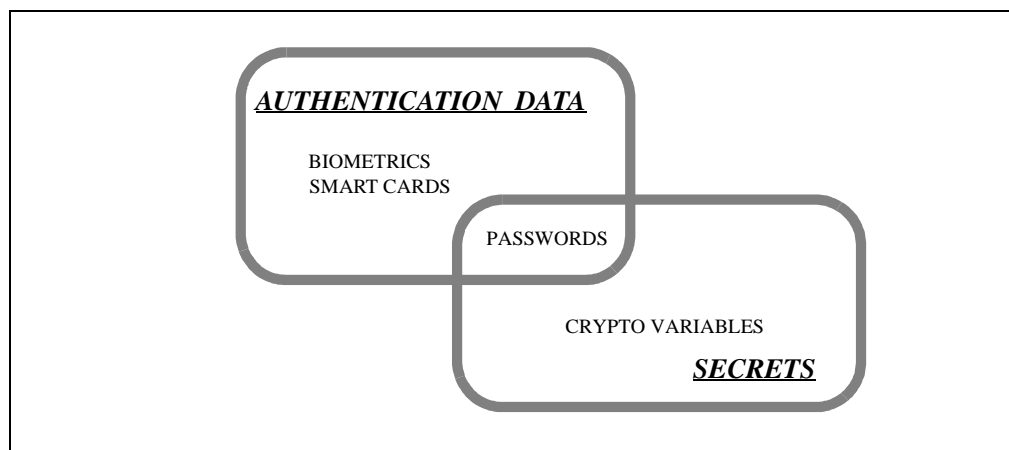


Figure 1.4 - Relationship between “authentication data” and “secrets”.

Chapter 2

Security functional components

2.1 Overview

42 This section defines the content and presentation of the functional requirements of the CC, and provides guidance on the organisation of the requirements for new components to be included in a security target and to be evaluated. The functional requirements are expressed in classes, families, and components.

2.1.1 Class structure

43 Figure 2.1 illustrates the functional class structure in diagrammatic form.

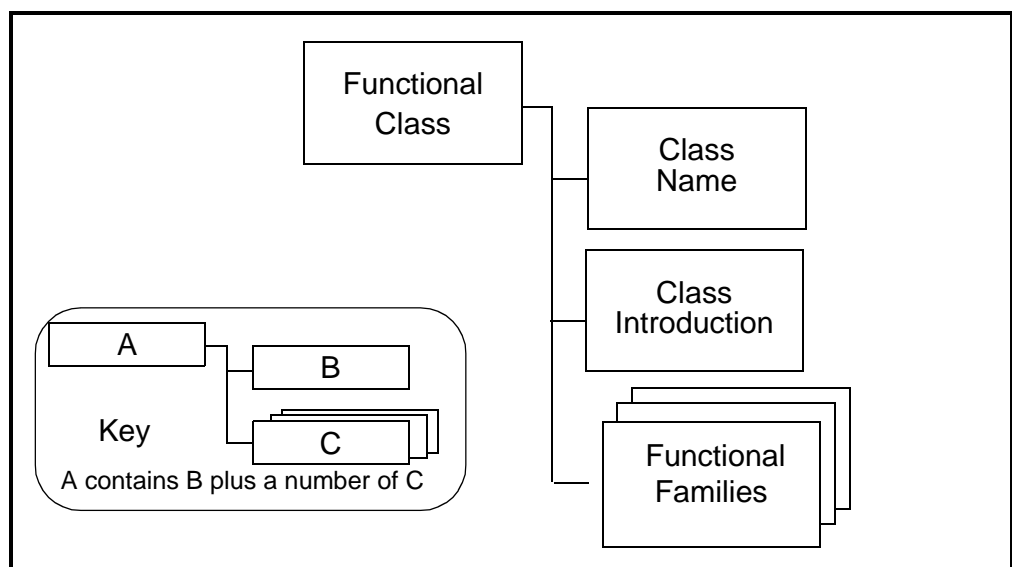


Figure 2.1 - Functional class structure

2.1.1.1 Class name

44 The class name section provides information necessary to identify and categorise 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.

D R A F T

2.1.1.2 Class introduction

45 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.

46 The class introduction provides a figure describing the families in this class and the hierarchy of the components in each family.

2.1.2 Family structure

47 Figure 2.2 illustrates the functional family structure in diagrammatic form.

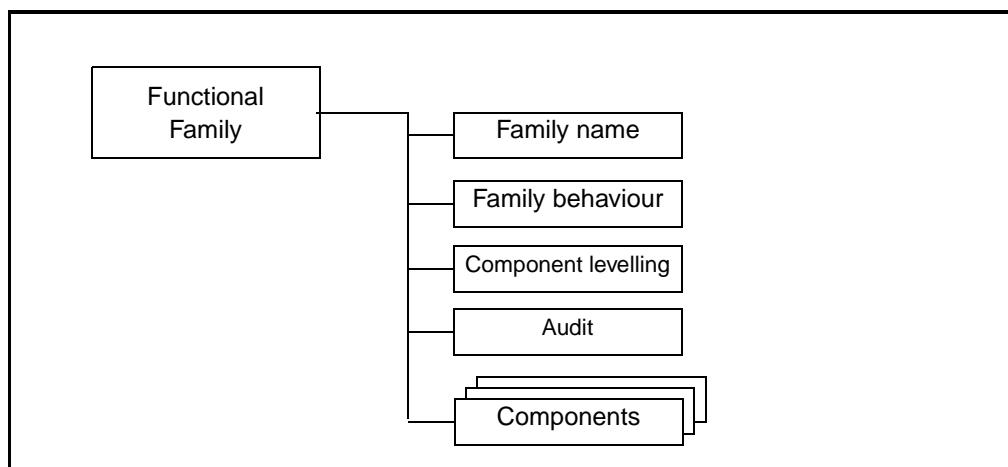


Figure 2.2 - Functional family structure

2.1.2.1 Family name

48 The family name section provides categorical and descriptive information necessary to identify and categorise 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 components.

2.1.2.2 Family behaviour

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

- a) The *security objectives* of the component family is a clear and concise statement of the security problem for which a TOE incorporating a component belonging to the family contributes to the solution.

D R A F T

- b) The description of the *functional requirements* summarises all the requirements that are included in the component(s). The description is aimed at authors of PPs, STs and functional packages who wish to assess whether the family is relevant to their specific requirements.

2.1.2.3 Component levelling

- 50 Functional families contain one or more components, any one of which can be selected for inclusion in PPs, STs and functional packages. The goal of this section 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.
- 51 This section of the functional family description describes the components available, their rationale, and the relationships between components. The exact details of the components are contained within each component.
- 52 The relationships between components within a functional family may or may not be hierarchical. A component is hierarchical to another if it offers more security; for example, the TOE limits the use of security services.

2.1.2.4 Management

- 53 The *management* requirements contain information for the PP/ST authors to consider as management activities for a given component. The management requirements are detailed in components of the management class (FMT).
- 54 A PP/ST author may select the indicated management requirements or may include other management requirements not listed. As such the information should be considered informative.

2.1.2.5 Audit

- 55 The *audit* requirements contain information for the PP/ST authors to select auditable events, if requirements from the class FAU Security Audit are included in the PP/ST. 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 family. For example, an audit note might include actions that are in terms of: Minimal - successful use of the security mechanism; Basic - any use of the security mechanism as well as relevant information regarding the security attributes involved; Detailed - any configuration changes made to the mechanism, including the actual configuration values before and after the change.
- 56 It should be observed that the categorisation of auditable events is hierarchical. For example, when Basic Audit Generation is desired, all auditable events identified as being both Minimal and Basic should be included in the PP/ST through the use of the appropriate assignment operation, except when the higher level event simply provides more detail than the lower level event. When Detailed Audit Generation is desired, all identified auditable events (Minimal, Basic and Detailed) should be included in the PP/ST.

D R A F T

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

2.1.3 Component structure

58 Figure 2.3 illustrates the functional component structure.

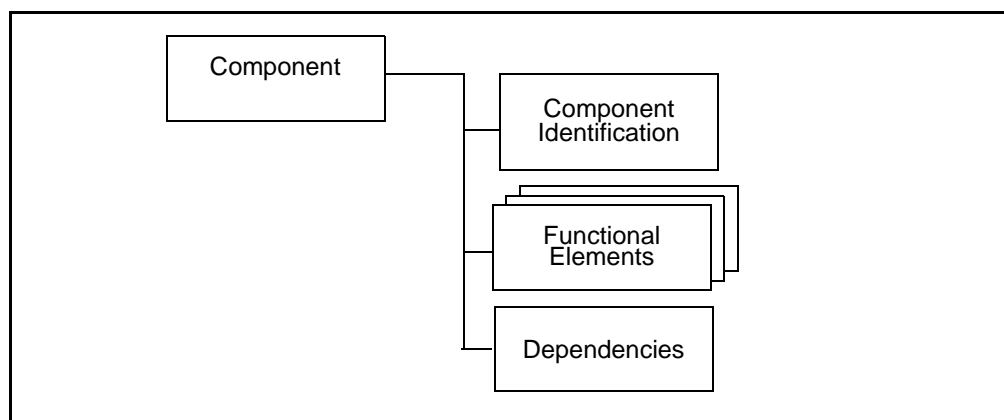


Figure 2.3 - Functional component structure

2.1.3.1 Component identification

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

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

61 *A short name.* A unique short form of the functional component name. This short name serves as the principal reference name for the categorisation, 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.

62 *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.

2.1.3.2 Functional elements

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

64 A functional element is a security functional requirement that if further divided would not yield a meaningful evaluation result. It is the smallest security functional requirement identified and recognised in the CC.

D R A F T

65 When building PPs/STs, it is not permitted to select only one or more elements from a component. The complete set of elements of a component must be selected for inclusion in a PP/ST.

66 A unique short form of the functional element name is provided. For example the requirement 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.

2.1.3.3 Dependencies

67 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.

68 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.

69 The dependency list identifies the minimum functional or assurance components needed to satisfy the security requirements associated with this component. Components which are hierarchical to the identified component may also be used to satisfy the dependency with the risk of introducing additional potential vulnerabilities.

70 In specific situations the indicated dependencies might not be applicable. The PP/ST author, by providing the rationale why it is not applicable, may remove the dependency.

2.1.4 Permitted functional component operations

71 The functional components used in the definition of the requirements in a PP, an ST or a functional package may be exactly as specified in Chapter 2 of this part, or they may be tailored to meet a specific security objective. However, selecting and tailoring these functional components is complicated by the fact that identified component dependencies shall be considered. Thus, this tailoring is restricted to an approved set of operations.

72 A list of permitted operations is included with each functional component. Not all operations are permitted on all functional components.

73 The permitted operations are selected from the following set:

- iteration, allows a component to be used more than once with varying operations,
- assignment: allows the specification of an identified parameter,
- selection: allows the specification of one or more elements from a list,
- refinement: allows the addition of details.

D R A F T

2.1.4.1 Iteration

- 74 Where necessary to cover different aspects of the same requirement (e.g. identification of more than one type of user), repetitive use of the same Part 2 component to cover each aspect is permitted.

2.1.4.2 Assignment

- 75 Some functional component elements contain parameters or variables that enable the PP/ST author to specify a policy or a set of values for incorporation into the PP or ST to meet a specific security objective. These elements clearly identify each parameter and constraint on values that may be assigned to that parameter.
- 76 Any aspect of an element whose acceptable values can be unambiguously described or enumerated can be represented by a parameter. The description or enumeration limits the permissible values in such a way that all possible choices will have the same dependencies (i.e. no choice will cause the listed dependencies to change).
- 77 The parameter may be an attribute or rule that narrows the requirement to a specific value or range of values. For instance, based on a specified security objective, the functional component element may state that a given operation should be performed a number of times. In this case, the assignment would provide the number, or range of numbers, to be used in the parameter.

2.1.4.3 Selection

- 78 This is the operation of picking one or more items from a list in order to narrow the scope of a component element.

2.1.4.4 Refinement

- 79 For all functional component elements the PP/ST author is permitted to limit the set of acceptable implementations by specifying additional detail in order to meet a security objective. Refinement of an element consists of adding these technical details.
- 80 Within a ST, the meanings of the terms subject and object might need to be explained for the TOE to be meaningful, and are therefore subject to refinement.
- 81 Like the other operations, the refinement does not levy any completely new requirements. It applies an elaboration, interpretation, or a special meaning to a requirement, rule, constant or condition based on security objectives. The refinement shall only further restrict the set of possible acceptable functions or mechanisms to implement the requirements, but never increase it. Because refinement does not allow new requirements to be created or existing requirements to be deleted, refinement does not have any impact on the list of dependencies associated with a component. The PP/ST author must be careful that the dependency needs of other requirements that depend on this requirement, are satisfied.

DRAFT

2.2 Component catalogue

- 82 The grouping of the components in this section does not reflect any formal taxonomy.
- 83 This part 2 contains classes of families and components, which are rough groupings on the basis of related function or purpose, presented in alphabetic order. At the start of each class is an informative diagram that indicates the taxonomy of each class, indicating the families in each class and the components in each family. The diagram is a useful indicator of the hierarchical relationship that may exist between components.
- 84 In the description of the functional components, a section identifies the dependencies between the component and any other components. These dependencies are normative. If a component is selected in a PP, ST or functional package the dependencies of this component should be satisfied in order to fulfil its intended functions, or rationale shall be provided why the dependency need not be satisfied.
- 85 In Figure 2.4 the class as shown contains three families. The first family, Family 1, contains three hierarchical components, where component 2 and component 3 can both be used to satisfy dependencies on component 1. Component 3 is hierarchical to component 2 and can also be used to satisfy dependencies on component 2.

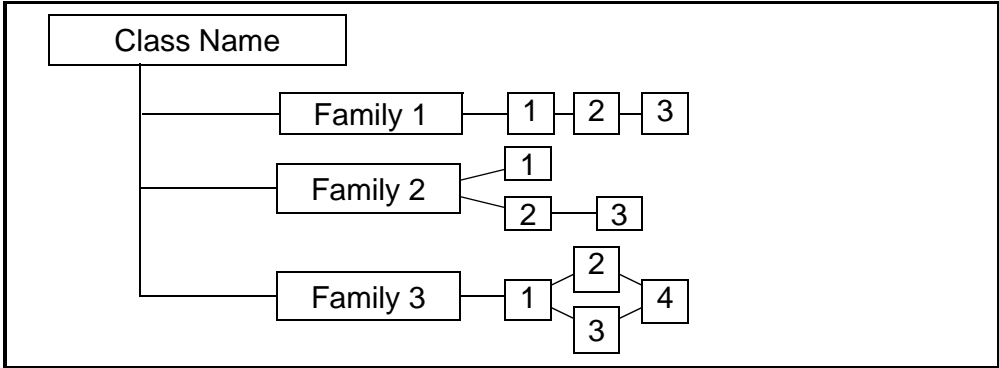


Figure 2.4 - Sample class decomposition diagram

- 86 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.
- 87 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.
- 88 These diagrams are meant to complement the text of the families and make identification of the relationships easier. They do not replace the “Hierarchical to:”

D R A F T

note in each component which is the mandatory claim of hierarchy for each component.

2.2.1 Component changes highlighting

89 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 and/or dependencies are bolded when they are enhanced or modified beyond the requirements of the previous component. In addition, any new or enhanced threats, application notes, and/or permitted operations beyond the previous component are also highlighted using **bold** type, whether it is in the main body or the Annexes of Part 2.

D R A F T

Class FAU

Security Audit

- 90 Security auditing involves recognising, recording, storing, and analysing information related to security relevant activities (i.e. any activity controlled by the TSP). The resulting audit records can be examined to determine which security relevant activities took place and who (which user) is responsible for them.

D R A F T

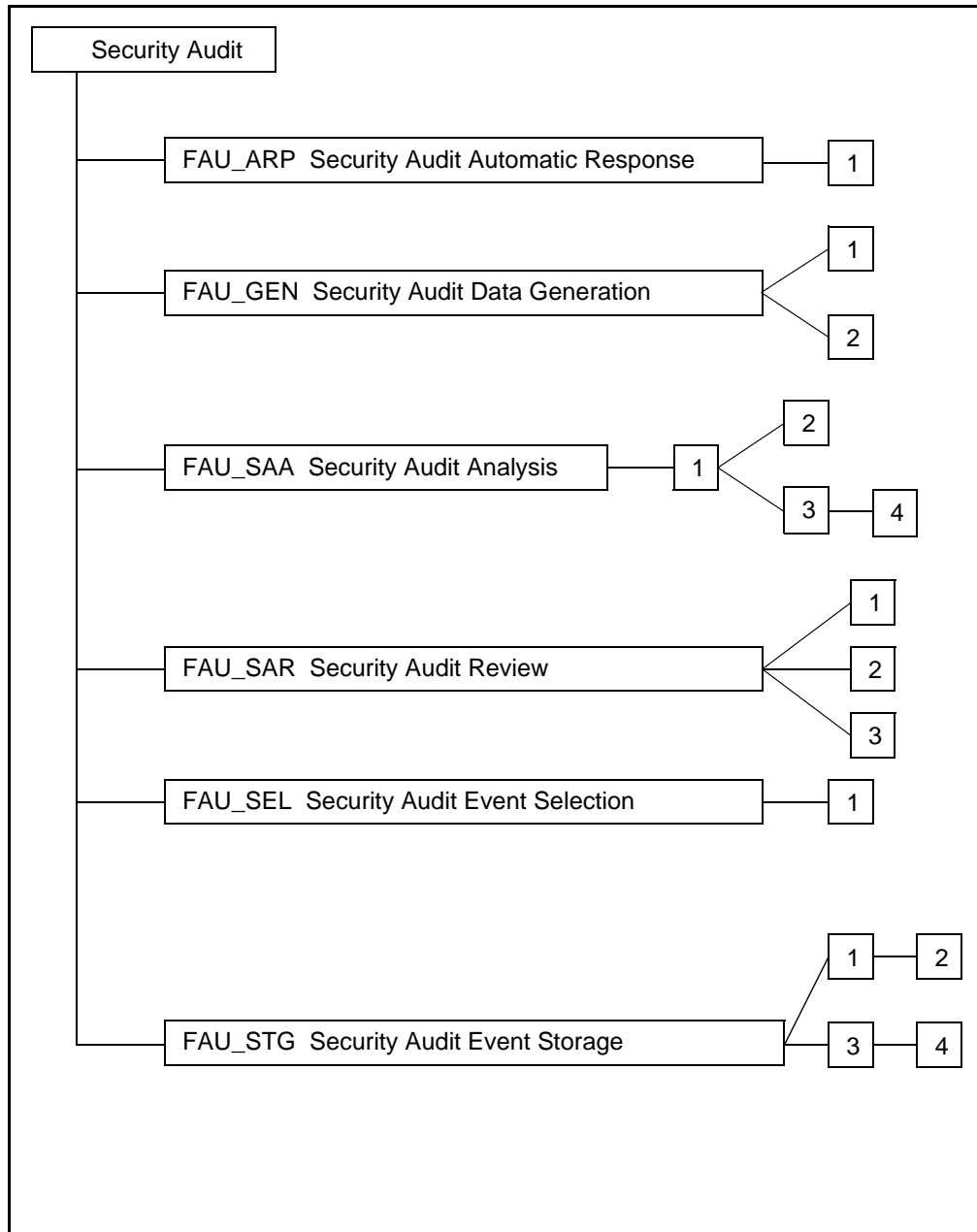


Figure 2.5 - Security Audit Class decomposition

D R A F T

FAU_ARP Security Audit Automatic Response

Family behaviour

- 91 This family defines the requirements specifying the conditions under which the TSF, after detection of events indicative of an imminent security violation, shall automatically react, and the reaction to those conditions that should be taken by the TSF.

Component levelling

FAU_ARP Security Audit Automatic Response

1

- 92 At FAU_ARP.1 Security Alarms, the TSF shall take actions in case a possible security violation is detected.

Management : FAU_ARP.1

- 93 The following actions could be considered for the management functions in FMT:
- a) the management (addition, removal, or modification) of actions.

Audit : FAU_ARP.1

- 94 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:
- a) Minimal: Actions taken due to imminent security violations.

FAU_ARP.1 Security Alarms

Hierarchical to: no other components.

- FAU_ARP.1.1 The TSF shall take [assignment: *list of the least disruptive actions*] to terminate the occurrence of security relevant events upon detection of a possible security violation.**

Dependencies : FAU_SAA.1 Imminent Violation Analysis

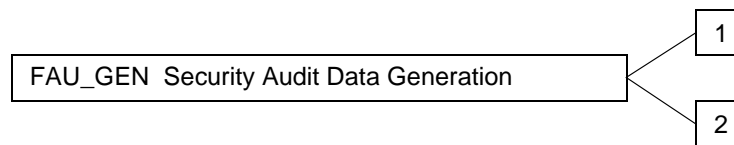
D R A F T

FAU_GEN Security Audit Data Generation

Family behaviour

- 95 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.

Component levelling



- 96 FAU_GEN.1 Audit Data Generation defines the level of auditable events, and specifies the list of data that shall be recorded in each record.

- 97 At FAU_GEN.2 User Identity Generation, the TSF shall associate auditable events to individual user identities.

Management :

- 98 There are no management activities foreseen.

Audit :

- 99 There are no actions identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

FAU_GEN.1 Audit Data Generation

Hierarchical to: no other components.

- FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:**

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

- FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:**

D R A F T

- a) **Date and time of the event, type of event, subject identity, and [selection: *success, failure*] of the event; and**
- b) **For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [assignment: *other audit relevant information*]**

Dependencies : FPT_STM.1 Time stamps

FAU_GEN.2 User Identity Generation

Hierarchical to: no other components.

FAU_GEN.2.1 The TSF shall be able to associate each auditable event with the identity of the user that caused the event.

Dependencies : **FAU_GEN.1 Audit Data Generation**
FIA_UID.1 Timing of Identification

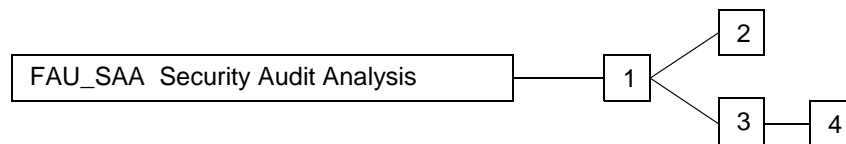
D R A F T

FAU_SAA Security Audit Analysis

Family behaviour

- 100 This family defines requirements for automated means which 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 an imminent security violation.
- 101 The actions to be taken based on the detection can be specified using the FAU_ARP family as desired.

Component levelling



- 102 In FAU_SAA.1 Imminent Violation Analysis, basic threshold detection on the basis of a fixed rule set is required.
- 103 In FAU_SAA.2 Profile Based Anomaly Detection, 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 (e.g. a single user, users who share a group ID or group account, users who operate under an assigned role, users of an entire system or network node) 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.
- 104 In FAU_SAA.3 Simple Attack Heuristics, the TSF shall be able to detect the occurrence of signature events that represent a significant threat to TSP enforcement. This search for signature events may occur in real-time or during a post-collection batch-mode analysis.
- 105 In FAU_SAA.4 Complex Attack Heuristics, 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 to match a signature event that indicates a potential violation of the TSP.

Management : for FAU_SAA.1

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

D R A F T

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

Management : for FAU_SAA.2

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

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

Management : for FAU_SAA.3

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

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

Management : for FAU_SAA.4

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

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

Audit : for FAU_SAA.1,FAU_SAA.2, FAU_SAA.3, FAU_SAA.4

110 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

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

FAU_SAA.1 Imminent Violation Analysis

Hierarchical to: no other components.

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 TSP.

FAU_SAA.1.2 The set of rules shall be:

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

D R A F T

Dependencies : FAU_GEN.1 Audit Data Generation

FAU_SAA.2 Profile Based Anomaly Detection

Hierarchical to: FAU_SAA.1

- 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: *specify the profile target group*].
- 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.
- FAU_SAA.2.3** The TSF shall be able to indicate an imminent violation of the TSP when a user's suspicion rating exceeds the following threshold conditions [assignment: *conditions under which anomalous activity is reported by the TSF*].

Dependencies : FIA_UID.1 Timing of Identification

FAU_SAA.3 Simple Attack Heuristics

Hierarchical to: FAU_SAA.1

- FAU_SAA.3.1** The TSF shall be able to maintain an internal representation of the following signature events [assignment: *a subset of system events*] that may indicate a violation of the TSP.
- FAU_SAA.3.2** The TSF shall be able to compare the signature events against the record of system activity discernable from an examination of [assignment: *specify the information to be used to determine system activity*].
- FAU_SAA.3.3** The TSF shall be able to indicate an imminent violation of the TSP when a system event is found to match a signature event that indicates a potential violation of the TSP.

Dependencies : No dependencies.

FAU_SAA.4 Complex Attack Heuristics

Hierarchical to: FAU_SAA.3

- FAU_SAA.4.1** The TSF shall be able to maintain an internal representation of the following **event sequences of known intrusion scenarios** [assignment: *list of sequences of system events whose occurrence are representative of known penetration scenarios*] and

D R A F T

the following signature events [assignment: *a subset of system events*] that may indicate a potential violation of the TSP.

FAU_SAA.4.2 The TSF shall be able to compare the signature events **and event sequences** against the record of system activity discernable from an examination of [assignment: *specify the information to be used to determine system activity*].

FAU_SAA.4.3 The TSF shall be able to indicate an imminent violation of the TSP when **system activity** is found to match a signature event **or event sequence** that indicates a potential violation of the TSP.

Dependencies : No dependencies.

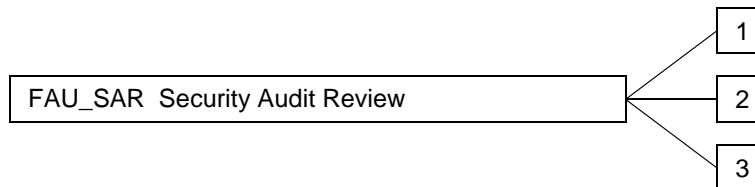
D R A F T

FAU_SAR Security Audit Review

Family behaviour

- 111 This family defines the requirements for audit tools that should be available to authorised users to assist in the review of audit data.

Component levelling



- 112 FAU_SAR.1 Audit Review provides the capability to read information from the audit records.
- 113 FAU_SAR.2 Restricted Audit Review requires that there are no other users except those that have been identified in FAU_SAR.1 that can read the information.
- 114 FAU_SAR.3 Selectable Audit Review requires audit review tools to select the audit data to be reviewed based on multiple criteria.

Management : for FAU_SAR.1

- 115 The following actions could be considered for the management functions in FMT:
- a) maintenance (deletion, modification, addition) of the group of users with read access right to the audit records.

Management : for FAU_SAR.2, FAU_SAR.3

- 116 There are no management activities foreseen.

Audit : FAU_SAR.1

- 117 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:
- a) Basic: Reading of information from the audit records.

Audit : FAU_SAR.2

- 118 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:
- a) Basic: Unsuccessful attempts to read information from the audit records.

D R A F T

Audit : FAU_SAR.3

119 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Detailed: the parameters used for the viewing.

FAU_SAR.1 Audit Review

120 This component will provide authorised administrators and users the capability to obtain and interpret the information. In case of human users this information needs to be in a human understandable presentation. In case of machine users the information needs to be unambiguously represented in an electronic fashion.

Hierarchical to: no other components.

FAU_SAR.1.1 The TSF shall provide [selection: *authorised administrator, authorised users*] with the capability to read [assignment: *list of audit information*] from the audit records.

FAU_SAR.1.2 The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

Dependencies : FAU_GEN.1 Audit Data Generation

FAU_SAR.2 Restricted Audit Review

Hierarchical to: no other components.

FAU_SAR.2.1 The TSF shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access.

Dependencies : FAU_SAR.1 Audit Review

FAU_SAR.3 Selectable Audit Review

Hierarchical to: no other components.

FAU_SAR.3.1 The TSF shall provide the ability to perform [selection: *searches, sorting, ordering*] of audit data based on [assignment: *multiple criteria with logical relations*].

Dependencies : FAU_SAR.1 Audit Review

D R A F T

FAU_SEL Security Audit Event Selection

Family behaviour

- 121 This family defines requirements to select the events to be audited during TOE operation. It defines requirements to include or exclude events from the set of auditable events.

FAU_SEL Security Audit Event Selection

1

- 122 FAU_SEL.1 Selective Audit, requires the ability to include or exclude events from the set of audited events based upon attributes to be specified by the PP/ST author.

Management : FAU_SEL.1

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

a) maintenance of the rights to view/modify the audit events.

Audit : FAU_SEL.1

- 124 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

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

FAU_SEL.1 Selective Audit

Hierarchical to: no other components.

- FAU_SEL.1.1 The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:**

a) [selection: *Object identity, User identity, Subject identity, Host identity, Event Type*]

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

- FAU_SEL.1.2 The TSF shall provide only the [selection: *authorised administrator, authorised users*] with the ability to [selection: *select, display*] which events are to be audited.**

Dependencies : FAU_GEN.1 Audit Data Generation

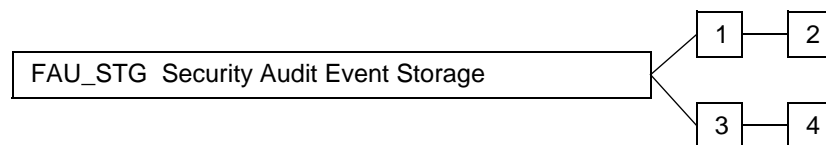
D R A F T

FAU_STG Security Audit Event Storage

Family behaviour

125 This family defines the requirement that the TSF be able to create a permanent audit trail of security audit events for later use and its maintenance.

Component levelling



126 At FAU_STG.1 Permanent Audit Trail Storage, a permanent audit trail is created.

127 FAU_STG.2 Guarantees of Audit Data Availability specifies the guarantees that the TSF maintains over the audit data given the occurrence of an undesired condition .

128 FAU_STG.3 Action in Case of Possible Audit Data Loss specifies actions to be taken if a threshold on the audit trail is triggered.

129 FAU_STG.4 Prevention of Audit Data Loss specifies actions in case the audit trail is full.

Management : for FAU_STG.1

130 There are no management activities foreseen.

Management : for FAU_STG.2

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

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

Management : for FAU_STG.3

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

- a) maintenance of the threshold;
- b) maintenance (deletion, modification, addition) of actions to be taken in case of imminent audit storage failure.

Management : for FAU_STG.4

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

D R A F T

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

Audit : FAU_STG.1, FAU_STG.2

- 134 There are no actions identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

Audit : FAU_STG.3

- 135 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Basic: Actions taken due to exceeding of a threshold.

Audit : FAU_STG.4

- 136 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Basic: Actions taken due to the audit storage failure.

FAU_STG.1 Permanent Audit Trail Storage

Hierarchical to: no other components.

FAU_STG.1.1 The TSF shall store generated audit records in a permanent audit trail.

Dependencies : FAU_GEN.1 Audit Data Generation

FAU_STG.2 Guarantees of Audit Data Availability

137

Hierarchical to: FAU_STG.1

FAU_STG.2.1 The TSF shall store generated audit records in a permanent audit trail.

FAU_STG.2.2 The TSF shall ensure that [assignment: *metric for saving audit records*] audit records will be maintained independent of [selection: *audit storage exhaustion, failure, attack*].

Dependencies : FAU_GEN.1 Audit Data Generation

FAU_SAR.1 Audit Review

D R A F T

FAU_STG.3 Action in Case of Possible Audit Data Loss

Hierarchical to: no other components.

FAU_STG.3.1 The TSF shall take [assignment: *actions to be taken in case of possible audit storage failure*] if the audit trail exceeds [assignment: *pre-defined limit*].

Dependencies : FAU_STG.1 Permanent Audit Trail Storage

FAU_STG.4 Prevention of Audit Data Loss

Hierarchical to: FAU_STG.3 Action in Case of Possible Audit Data Loss

FAU_STG.4.1 The TSF shall [selection: *ignoring, preventing*] the occurrence of auditable actions, except those taken by the authorised administrator, and [assignment: *actions to be taken in case of possible audit storage failure*] if the audit trail is full.

Dependencies : FAU_GEN.1 Audit Data Generation
FAU_SAR.1 Audit Review

D R A F T

D R A F T

Class FCO

Communication

138 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.

139 Figure 2.6 shows the decomposition of this class into its constituent components.

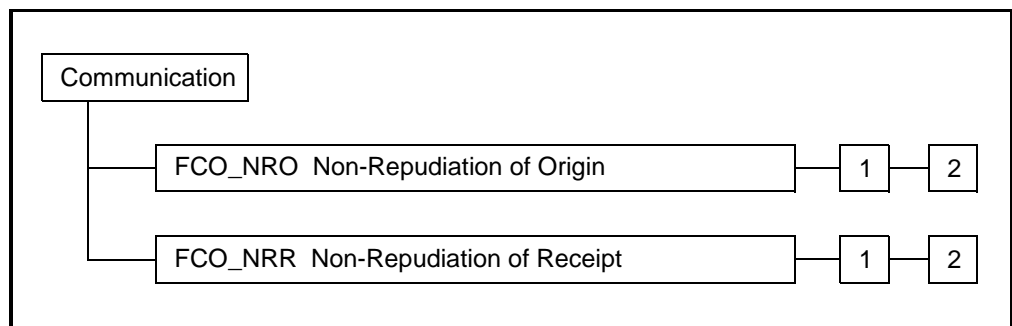


Figure 2.6 - Communication class decomposition

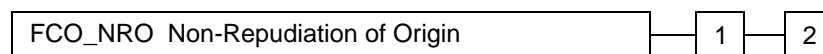
D R A F T

FCO_NRO Non-Repudiation of Origin

Family behaviour

- 140 Non-repudiation of origin ensures that the originator of information cannot successfully deny having sent the information. The TSF shall 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 be verified by either this subject or other subjects.

Component levelling



- 141 FCO_NRO.1 Selective Proof of Origin requires the TSF to provide subjects with the capability to request evidence of the origin of information.

- 142 FCO_NRO.2 Enforced Proof of Origin requires that the TSF always generate evidence of origin for transmitted information.

Management : for FCO_NRO.1 and FCO_NRO.2

- 143 The following actions could be considered for the management functions in FMT:
- a) The management of changes to information types, fields, originator attributes and recipients of evidence.

Audit : for FCO_NRO.1

- 144 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:
- a) Minimal: The identity of the user who requested that evidence of origin would be generated.
 - b) Minimal: The invocation of the non-repudiation service.
 - c) Basic: Identification of the information, the destination, and a copy of the evidence provided.
 - d) Detailed: The identity of the user who requested a verification of the evidence.

Audit : for FCO_NRO.2

- 145 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:
- a) Minimal: The invocation of the non-repudiation service.

D R A F T

- b) Basic: Identification of the information, the destination, and a copy of the evidence provided.
- c) Detailed: The identity of the user who requested a verification of the evidence.

FCO_NRO.1 Selective Proof of Origin

Hierarchical to: no other components.

FCO_NRO.1.1 The TSF shall be able to generate evidence of origin for transmitted [assignment: *list of information types*] at the request of the [selection: *originator, recipient*, [assignment: *list of third parties*]].

FCO_NRO.1.2 The TSF shall be able to relate the [assignment: *list of attributes*] of the originator of the information, and the [assignment: *list of information fields*] of the information to which the evidence applies.

FCO_NRO.1.3 The TSF shall provide a capability to verify the evidence of origin of information to [selection: *originator, recipient*, [assignment: *list of third parties*]] given [assignment: *limitations on the evidence of origin*].

Dependencies : FIA_UID.1 Timing of Identification

FCO_NRO.2 Enforced Proof of Origin

Hierarchical to: FCO_NRO.1

FCO_NRO.2.1 The TSF shall **enforce the generation of** evidence of origin for transmitted [assignment: *list of information types*] at all times.

FCO_NRO.2.2 The TSF shall be able to relate the [assignment: *list of attributes*] of the originator of the information, and the [assignment: *list of information fields*] of the information to which the evidence applies.

FCO_NRO.2.3 The TSF shall provide a capability to verify the evidence of origin of information to [selection: *originator, recipient*, [assignment: *list of third parties*]] given [assignment: *limitations on the evidence of origin*].

Dependencies : FIA_UID.1 Timing of Identification

D R A F T

FCO_NRR Non-Repudiation of Receipt

Family behaviour

146 Non-repudiation of receipt ensures that the recipient of information cannot successfully deny receiving the information. The TSF shall 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 be verified by either this subject or other subjects.

Component levelling

FCO_NRR Non-Repudiation of Receipt

1

2

147 FCO_NRR.1 Selective Proof of Receipt requires the TSF to provide subjects with a capability to request evidence of the receipt of information.

148 FCO_NRR.2 Enforced Proof of Receipt requires that the TSF always generate evidence of receipt for received information.

Management : for FCO_NRR.1 and FCO_NRR.2

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

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

Audit : for FCO_NRR.1

150 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

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

Audit : for FCO_NRR.2

151 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Minimal: The invocation of the non-repudiation service.

D R A F T

- b) Basic: Identification of the information, the destination, and a copy of the evidence provided.
- c) Detailed: The identity of the user who requested a verification of the evidence.

FCO_NRR.1 Selective Proof of Receipt

Hierarchical to: no other components.

FCO_NRR.1.1 The TSF shall be able to generate evidence of receipt for received [assignment: *list of information types*] at the request of the [selection: *originator, recipient, [assignment: *list of third parties*]*].

FCO_NRR.1.2 The TSF shall be able to relate the [assignment: *list of attributes*] of the recipient of the information, and the [assignment: *list of information fields*] of the information to which the evidence applies.

FCO_NRR.1.3 The TSF shall provide a capability to verify the evidence of receipt of information to [selection: *originator, recipient, [assignment: *list of third parties*]*] given [assignment: *limitations on the evidence of receipt*].

Dependencies : FIA_UID.1 Timing of Identification

FCO_NRR.2 Enforced Proof of Receipt

Hierarchical to: FCO_NRR.1

FCO_NRR.2.1 The TSF shall **enforce the generation of** evidence of receipt for received [assignment: *list of information types*].

FCO_NRR.2.2 The TSF shall be able to relate the [assignment: *list of attributes*] of the recipient of the information, and the [assignment: *list of information fields*] of the information to which the evidence applies.

FCO_NRR.2.3 The TSF shall provide a capability to verify the evidence of receipt of information to [selection: *originator, recipient, [assignment: *list of third parties*]*] given [assignment: *limitations on the evidence of receipt*].

Dependencies : FIA_UID.1 Timing of Identification

D R A F T

D R A F T

Class FCS

Cryptographic Support

- 152 The TSF may employ cryptographic functionality to help satisfy several high-level security objectives. These include (but are not limited to): identification and authentication, non-repudiation, trusted path, trusted channel and data separation. This class is used when the TOE implements cryptographic functions, the implementation of which could be in hardware, firmware and/or software.
- 153 The FCS class is organised into two families: FCS_CKM Cryptographic Key Management and FCS_COP Cryptographic Operation. The FCS_CKM family addresses the management aspects of cryptographic keys, while the FCS_COP family is concerned with the operational use of those cryptographic keys.
- 154 Figure 2.7 shows the decomposition of this class into its constituent components.

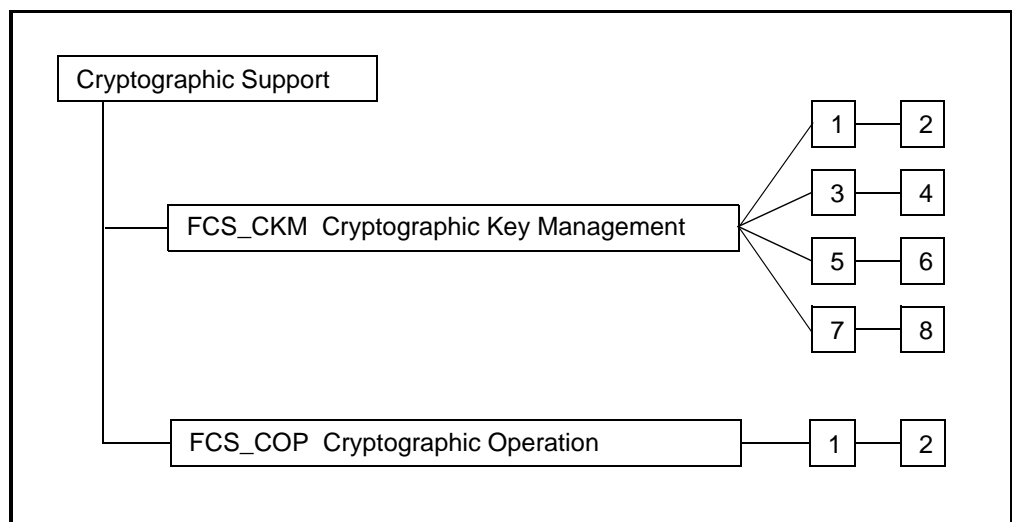


Figure 2.7 - Cryptographic Support class decomposition

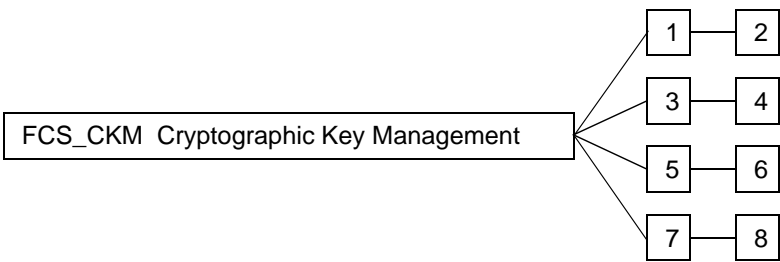
DRAFT

FCS_CKM Cryptographic Key Management

Family behaviour

155 Cryptographic keys must 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 and cryptographic key destruction. This family should be included whenever there are functional requirements for the management of cryptographic keys.

Component levelling



- 156 FCS_CKM.1 Cryptographic Key Generation requires cryptographic keys to be generated in accordance with a specified algorithm and key size.
- 157 FCS_CKM.2 Standards-Based Cryptographic Key Generation is hierarchical to FCS_CKM.1 and requires cryptographic keys to be generated in accordance with a specified algorithm and key sizes based on an assigned standard.
- 158 FCS_CKM.3 Cryptographic Key Distribution requires cryptographic keys to be distributed in accordance with a specified distribution method.
- 159 FCS_CKM.4 Standards-Based Cryptographic Key Distribution is hierarchical to FCS_CKM.3 and requires cryptographic keys to be distributed in accordance with a specified distribution method which is based on an assigned standard.
- 160 FCS_CKM.5 Cryptographic Key Access requires access to cryptographic keys to be performed in accordance with a specified access method.
- 161 FCS_CKM.6 Standards-Based Cryptographic Key Access is hierarchical to FCS_CKM.5 and requires access to cryptographic keys to be performed in accordance with a specified access method which is based on an assigned standard.
- 162 FCS_CKM.7 Cryptographic Key Destruction requires cryptographic keys to be destroyed in accordance with a specified destruction method.
- 163 FCS_CKM.8 Standards-Based Cryptographic Key Destruction is hierarchical to FCS_CKM.7 and requires cryptographic keys to be destroyed in accordance with a specified destruction method which is based on an assigned standard.

D R A F T

Management : for FCS_CKM.1, FCS_CKM.2, FCS_CKM.3, FCS_CKM.4, FCS_CKM.5, FCS_CKM.6, FCS_CKM.7 and FCS_CKM.8

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

- a) the management of changes to cryptographic key attributes, examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption).

Audit : for FCS_CKM.1, FCS_CKM.2, FCS_CKM.3, FCS_CKM.4, FCS_CKM.5, FCS_CKM.6, FCS_CKM.7 and FCS_CKM.8

165 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Minimal: Success and failure of the activity.
- b) Basic: The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys).

FCS_CKM.1 Cryptographic Key Generation

Hierarchical to: no other components.

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm and specified cryptographic key sizes.

Dependencies : [FCS_CKM.3 Cryptographic Key Distribution
or
FCS_COP.1 Cryptographic Operation]
FCS_CKM.7 Cryptographic Key Destruction
FMT_MSA.2 Safe security attributes

FCS_CKM.2 Standards-Based Cryptographic Key Generation

Hierarchical to: FCS_CKM.1

FCS_CKM.2.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm and specified cryptographic key sizes which meet the following standard: [assignment: *list of international standards, list of national standards, list of industry standards, list of organisational standards*].

D R A F T

Dependencies : [FCS_CKM.3 Cryptographic Key Distribution
or
FCS_COP.1 Cryptographic Operation]
FCS_CKM.7 Cryptographic Key Destruction
FMT_MSA.2 Safe security attributes

FCS_CKM.3 Cryptographic Key Distribution

Hierarchical to: no other components.

FCS_CKM.3.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method.

Dependencies : [FDP_ETC.1 Export of User Data Without Security Attributes
or
FCS_CKM.1 Cryptographic Key Generation]
FCS_CKM.7 Cryptographic Key Destruction
FMT_MSA.2 Safe security attributes

FCS_CKM.4 Standards-Based Cryptographic Key Distribution

Hierarchical to: FCS_CKM.3

FCS_CKM.4.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method **which meets the following standard:** [assignment: *list of international standards, list of national standards, list of industry standards, list of organisational standards*].

Dependencies : [FDP_ETC.1 Export of User Data Without Security Attributes
or
FCS_CKM.1 Cryptographic Key Generation]
FCS_CKM.7 Cryptographic Key Destruction
FMT_MSA.2 Safe security attributes

FCS_CKM.5 Cryptographic Key Access

Hierarchical to: no other components.

FCS_CKM.5.1 The TSF shall perform [assignment: *type of cryptographic key access*] in accordance with a specified cryptographic key access method.

D R A F T

Dependencies : FMT_MSA.2 Safe security attributes
[FDP_ITC.1 Import of User Data Without Security Attributes
or
FCS_CKM.1 Cryptographic Key Generation]
FCS_CKM.7 Cryptographic Key Destruction

FCS_CKM.6 Standards-Based Cryptographic Key Access

Hierarchical to: FCS_CKM.5

FCS_CKM.6.1 The TSF shall perform [assignment: *type of cryptographic key access*] in accordance with a specified cryptographic key access method **which meets the following standard:** [assignment: *list of international standards, list of national standards, list of industry standards, list of organisational standards*].

Dependencies : FMT_MSA.2 Safe security attributes
[FDP_ITC.1 Import of User Data Without Security Attributes
or
FCS_CKM.1 Cryptographic Key Generation]
FCS_CKM.7 Cryptographic Key Destruction

FCS_CKM.7 Cryptographic Key Destruction

Hierarchical to: no other components.

FCS_CKM.7.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method.

Dependencies : FMT_MSA.2 Safe security attributes
[FDP_ITC.1 Import of User Data Without Security Attributes
or
FCS_CKM.1 Cryptographic Key Generation]

FCS_CKM.8 Standards-Based Cryptographic Key Destruction

Hierarchical to: FCS_CKM.7

FCS_CKM.8.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method **which meets the following standard:**

D R A F T

[assignment: *list of international standards, list of national standards, list of industry standards, list of organisational standards*].

Dependencies : FMT_MSA.2 Safe security attributes

[FDP_ITC.1 Import of User Data Without Security Attributes

or

FCS_CKM.1 Cryptographic Key Generation]

D R A F T

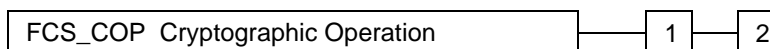
FCS_COP Cryptographic Operation

Family behaviour

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

167 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.

Component levelling



168 FCS_COP.1 Cryptographic Operation requires a cryptographic operation to be performed in accordance with a specified algorithm and with a cryptographic key of a specified size.

169 FCS_COP.2 Standards-Based Cryptographic Operation is hierarchical to FCS_COP.1. It requires a cryptographic operation to be performed in accordance with a specified algorithm and with a cryptographic key of a specified size and which is based on an assigned standard.

Management : for FCS_COP.1 and FCS_COP.2

170 There are no management activities foreseen for these components.

Audit : for FCS_COP.1 and FCS_COP.2

171 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

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

D R A F T

FCS_COP.1 Cryptographic Operation

Hierarchical to: no other components.

FCS_COP.1.1 The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm and cryptographic key size.

Dependencies : FMT_MSA.2 Safe security attributes

[FDP_ITC.1 Import of User Data Without Security Attributes

or

FCS_CKM.1 Cryptographic Key Generation]

FCS_CKM.7 Cryptographic Key Destruction

FCS_COP.2 Standards-Based Cryptographic Operation

Hierarchical to: FCS_COP.1

FCS_COP.2.1 The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm and cryptographic key size **which meet the following standard:** [assignment: *list of international standards, list of national standards, list of industry standards, list of organisational standards*].

Dependencies : FMT_MSA.2 Safe security attributes

[FDP_ITC.1 Import of User Data Without Security Attributes

or

FCS_CKM.1 Cryptographic Key Generation]

FCS_CKM.7 Cryptographic Key Destruction

D R A F T

Class FDP

User Data Protection

172 This class contains families specifying requirements for TOE security functions and TOE security function policies related to protecting user data. FDP is split into five groups of families (listed below) which address user data within a TOE, during import, export, and storage as well as security attributes directly related to user data.

173 The families in this class are organised into five groups:

a) User Data Protection Security Function Policies:

- FDP_ACC Access Control Policy; and
- FDP_IFC Information Flow Control Policy.

Components in these families permit the PP/ST author to specify data protection security function policies to address the security objectives.

b) Forms of User Data Protection:

- FDP_ACF Access Control Functions;
- FDP_IFF Information Flow Control Functions;
- FDP_ITT Internal TOE Transfer;
- FDP_RIP Residual Information Protection;
- FDP_ROL Rollback; and
- FDP_SDI Stored Data Integrity.

Components in these families address functions that protect information and objects through enforcement of the data protection.

c) Off-line Storage, Import and Export:

- FDP_DAU Data Authentication;
- FDP_ETC Export to Outside TSF Control; and
- FDP_ITC Import from Outside TSF Control.

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

d) Inter-TSF Communication:

- FDP_UCT Inter-TSF User Data Confidentiality Transfer Protection; and
- FDP_UIT Inter-TSF User Data Integrity Transfer Protection.

Components in these families address communication between the TSF of the TOE and another Trusted IT Product.

Figures 2.8 and 2.9 show the decomposition of this class into its constituent components.

D R A F T

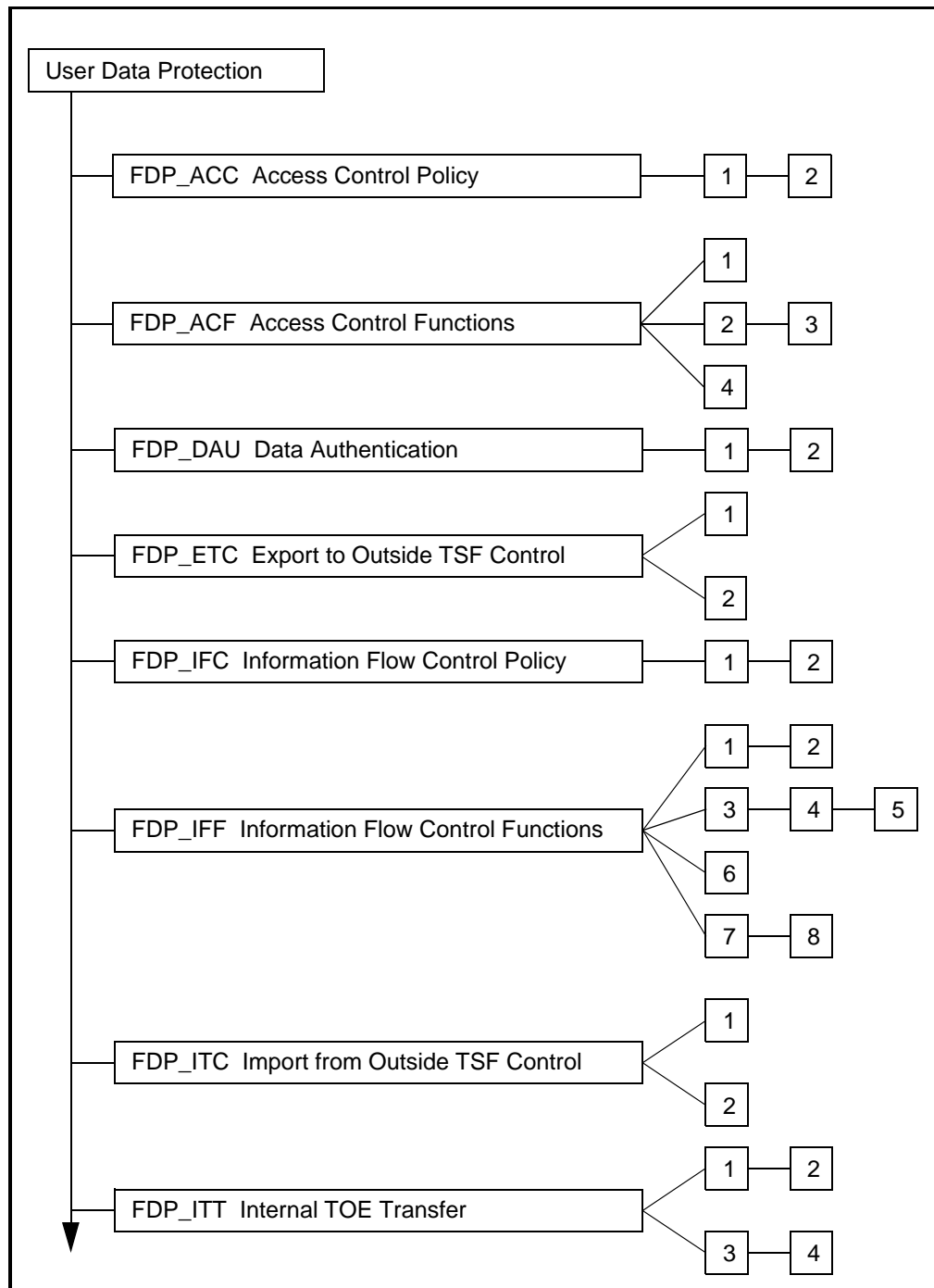


Figure 2.8 - User Data Protection class decomposition

D R A F T

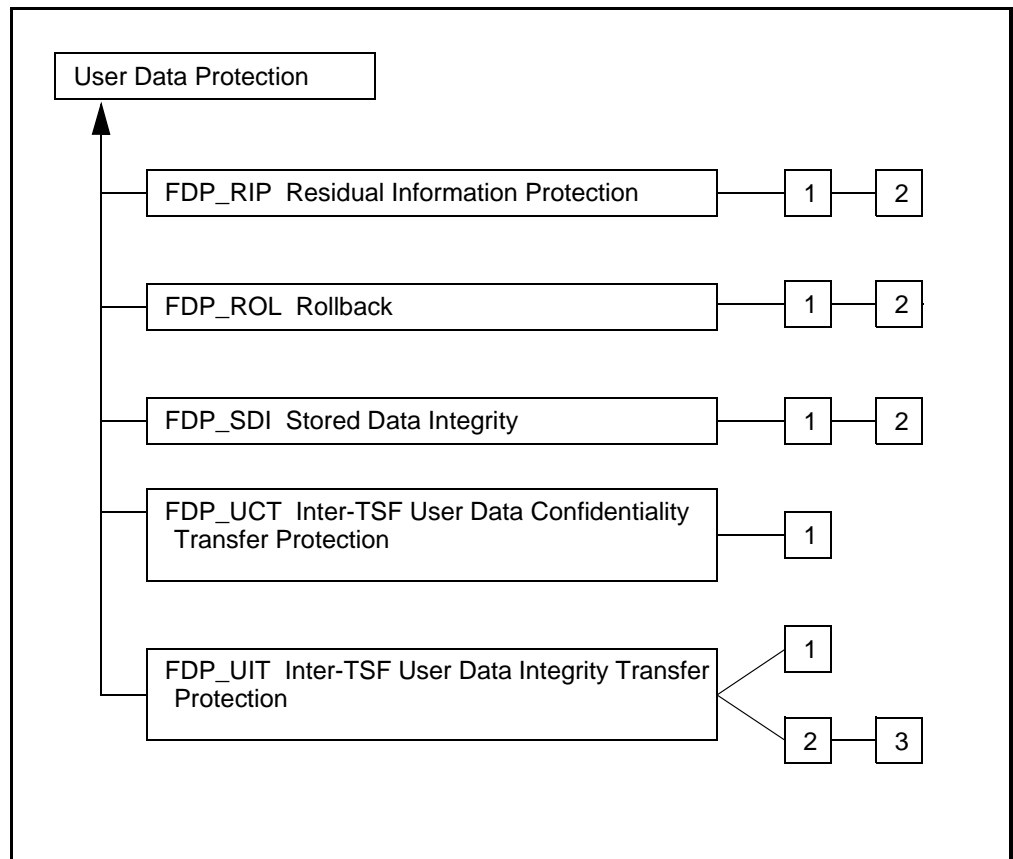


Figure 2.9 - User Data Protection class decomposition (cont.)

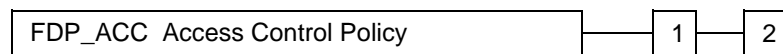
D R A F T

FDP_ACC Access Control Policy

Family behaviour

- 174 This family defines the scope of control of the access control policies that form the access control portion of the TSP. This scope of control is characterised 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 allows multiple policies to exist, each having a unique name.

Component levelling



- 175 FDP_ACC.1 Subset Access Control requires that each access control SFPs be in place for a subset of the possible operations on a subset of the objects in the TOE.

- 176 FDP_ACC.2 Complete Access Control requires that each access control SFP cover all operations on subjects and objects covered by that SFP. It further requires that all objects and operations with the TSC are covered by at least one access control SFP.

Management : for FDP_ACC.1 and FDP_ACC.2

- 177 There are no management activities foreseen for this component.

Audit : for FDP_ACC.1 and FDP_ACC.2

- 178 There are no events identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

FDP_ACC.1 Subset Access Control

Hierarchical to: no other components.

- FDP_ACC.1.1 **The TSF shall enforce the [assignment: *access control SFP*] on [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*].**

Dependencies : FDP_ACF.1 Security Attribute Based Access Control

D R A F T

FDP_ACC.2 Complete Access Control

Hierarchical to: FDP_ACC.1

FDP_ACC.2.1 The TSF shall enforce the [assignment: *access control SFP*] on [assignment: *list of subjects and objects*] **and all operations among subjects and objects covered by the SFP.**

FDP_ACC.2.2 **The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.**

Dependencies : FDP_ACF.1 Security Attribute Based Access Control

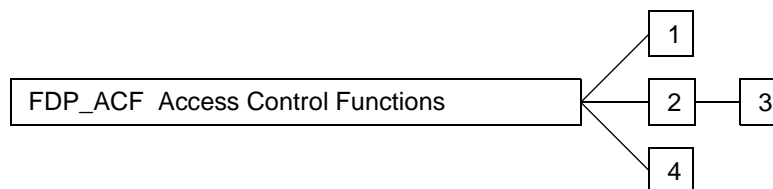
D R A F T

FDP_ACF Access Control Functions

Family behaviour

179 This family describes specific functions that can implement the rules for access control and is to be used in conjunction with FDP_ACC which specifies the access control policies.

Component levelling



180 The family addresses security attribute usage (FDP_ACF.1), flexible characteristics of policies (FDP_ACF.2 and FDP_ACF.3), and fixed characteristics of policies (FDP_ACF.4). These components are to be combined to describe the function implementing the SFP as defined in FDP_ACC. The PP/ST author may also iterate some components multiple times to address multiple policies in the TOE.

181 FDP_ACF.1 Security Attribute Based Access Control allows the TSF to enforce access based upon security attribute and named groups of attributes.

182 FDP_ACF.2 Access Authorisation adds the ability to explicitly authorise access to an object based upon security attributes.

183 FDP_ACF.3 Access Authorisation and Denial provides the ability to explicitly deny access to an object in addition to the requirements of FDP_ACF.2.

184 FDP_ACF.4 Fixed Access Control prohibits the ability to change security attributes, deny access, or authorise access to the object.

Management : for FDP_ACF.1, and FDP_ACF.4

185 There are no management activities foreseen for these components.

Management : for FDP_ACF.2 and FDP_ACF.3

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

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

Audit : for FDP_ACF.1

187 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

D R A F T

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

Audit : for FDP_ACF.2 and FDP_ACF.3

188 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Minimal: Successful attempts to specify the authorising or denying of access to an object.
- b) Basic: Unsuccessful attempts to specify the authorising or denying of access to an object.
- c) Detailed: The identity of the user or subject who specifies, or attempts to specify, the authorising or denying of access to an object.

Audit : for FDP_ACF.4

189 There are no events identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

FDP_ACF.1 Security Attribute Based Access Control

Hierarchical to: no other components.

FDP_ACF.1.1 The TSF shall enforce the [assignment: *access control SFP*] to objects based on [assignment: *security attributes, named groups of security attributes*].

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: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*].

Dependencies : **FDP_ACC.1** Subset Access Control

FMT_MSA.3 Static Attribute Initialisation

FDP_ACF.2 Access Authorisation

Hierarchical to: no other components.

FDP_ACF.2.1 The TSF shall ensure that the access control SF that enforces the [assignment: *access control SFP*] shall explicitly authorise access based on the [assignment: *value of security attributes of subjects and objects*].

D R A F T

Dependencies : **FDP_ACC.1 Subset Access Control**
FMT_MSA.3 Static Attribute Initialisation

FDP_ACF.3 Access Authorisation and Denial

Hierarchical to: FDP_ACF.2

FDP_ACF.3.1 The TSF shall ensure that the access control SF that enforces the [assignment: *access control SFP*] shall explicitly authorise access based on the [assignment: *value of security attributes of subjects and objects*].

FDP_ACF.3.2 **The TSF shall ensure that the access control SF that enforces the [assignment: *access control SFP*] shall explicitly deny access based on the [assignment: *value of security attributes of subjects and objects*].**

Dependencies : FDP_ACC.1 Subset Access Control
FMT_MSA.3 Static Attribute Initialisation

FDP_ACF.4 Fixed Access Control

Hierarchical to: no other components.

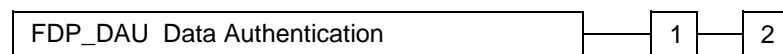
FDP_ACF.4.1 **The TSF shall enforce the [assignment: *access control SFP*] so that the security attributes of the controlled objects and subjects cannot be changed.**

Dependencies : **FDP_ACC.1 Subset Access Control**

D R A F T

FDP_DAU Data Authentication**Family behaviour**

190 Data authentication permits an entity to accept responsibility for the authenticity of information, for example by digitally signing it. 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.

Component levelling

191 FDP_DAU.1 Basic Data Authentication requires that the TSF is capable of generating a guarantee of authenticity of the information content of objects (e.g. documents).

192 FDP_DAU.2 Data Authentication with Identity of Guarantor additionally requires that the TSF is capable of establishing the identity of the subject who provided the guarantee of authenticity.

Management : for FDP_DAU.1 and FDP_DAU.2

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

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

Audit : for FDP_DAU.1

194 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

- a) Minimal: Successful generation of validity evidence.
- b) Basic: Unsuccessful generation of validity evidence.
- c) Detailed: The identity of the subject that requested the evidence.

Audit : for FDP_DAU.2

195 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

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

D R A F T

- c) Detailed: The identity of the subject that requested the evidence.
- d) Detailed: The identity of the subject that generated the evidence.

FDP_DAU.1 Basic Data Authentication

Hierarchical to: no other components

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: *list of objects or information types*].

FDP_DAU.1.2 The TSF shall provide [assignment: *list of subjects*] with the ability to verify evidence of the validity of the indicated information.

Dependencies: None

FDP_DAU.2 Data Authentication with Identity of Guarantor

Hierarchical to: **FDP_DAU.1**

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: *list of objects or information types*].

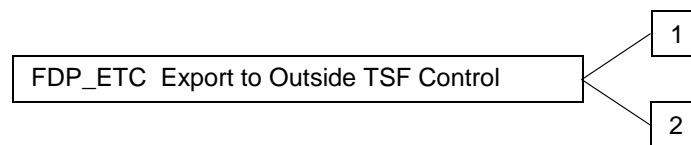
FDP_DAU.2.2 The TSF shall provide [assignment: *list of subjects*] with the ability to verify evidence of the validity of the indicated information **and the identity of the subject that generated the evidence.**

Dependencies: None

D R A F T

FDP_ETC Export to Outside TSF Control**Family behaviour**

196 This family defines functions for exporting 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.

Component levelling

197 FDP_ETC.1 Export of User Data Without Security Attributes requires that the TSF enforce 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.

198 FDP_ETC.2 Export of User Data With Security Attributes requires that the TSF enforce the appropriate SFPs using a function that accurately and unambiguously associates security attributes with the user data that is exported.

Management : for FDP_ETC.1

199 There are no management activities foreseen for this component.

Management : for FDP_ETC.2

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

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

Audit : for FDP_ETC.1 and FDP_ETC.2

201 The following events shall be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

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

D R A F T

FDP_ETC.1 Export of User Data Without Security Attributes

Hierarchical to: no other components.

FDP_ETC.1.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] when exporting user data, controlled under the SFP, outside of the TSC.

FDP_ETC.1.2 The TSF shall export the user data without the user data's associated security attributes.

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]

FDP_ETC.2 Export of User Data With Security Attributes

Hierarchical to: no other components.

FDP_ETC.2.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] when exporting user data, controlled under the SFP, outside of the TSC.

FDP_ETC.2.2 The TSF shall export the user data with the user data's associated security attributes.

FDP_ETC.2.3 The TSF shall ensure that the security attributes, when exported outside the TSC, are unambiguously associated with the exported user data.

FDP_ETC.2.4 The TSF shall enforce [assignment: *additional exportation control rules*] when user data is exported from the TSC.

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]
[FTP_ITC.1 Inter-TSF Trusted Channel or
FTP_TRP.1 Trusted Path]
FPT_TDC.1 Inter-TSF Basic TSF Data Consistency

D R A F T

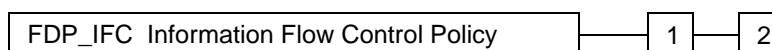
FDP_IFC Information Flow Control Policy

Family behaviour

202 This family defines the scope of control of the information flow control SFPs that enforce rules preventing the unauthorised flow of information among subjects and objects. This family is distinct from FDP_IFF Information Flow Control Functions in order to separate policy from mechanism. The family defines a set of named information flow control SFPs; and, for each, specifies the scope of control.

203 The TSF mechanism controls the flow of information in accordance with the information flow control SFP. Operations which would change the information flow control security attributes are not generally permitted as this would be in violation of an information flow control SFP.

Component levelling



204 FDP_IFC.1 Subset Information Flow Control requires that each information flow control SFPs be in place for a subset of the possible operations on a subset of the objects in the TOE.

205 FDP_IFC.2 Complete Information Flow Control requires that each information flow control SFP cover all operations on subjects and objects covered by that SFP. It further requires that all objects and operations with the TSC are covered by at least one information flow control SFP. In conjunction with the FPT_RVM.1 component, this gives the “always invoked” aspect of a reference monitor.

Management : for FDP_IFC.1 and FDP_IFC.2

206 There are no management activities foreseen for this component.

Audit : for FDP_IFC.1 and FDP_IFC.2

207 There are no events identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

FDP_IFC.1 Subset Information Flow Control

Hierarchical to: no other components.

FDP_IFC.1.1 The TSF shall enforce the [assignment: *information flow control SFP*] on [assignment: *list of subjects, objects and operations among subjects and objects covered by the SFP*].

Dependencies : FDP_IFF.1 Simple Security Attributes

D R A F T

FDP_IFC.2 Complete Information Flow Control

Hierarchical to: FDP_IFC.1

FDP_IFC.2.1 The TSF shall enforce the [assignment: *information flow control SFP*] on [assignment: *list of subjects and objects*] and **all operations among subjects and objects covered by the SFP.**

FDP_IFC.2.2 **The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an information flow control SFP.**

Dependencies : FDP_IFF.1 Simple Security Attributes

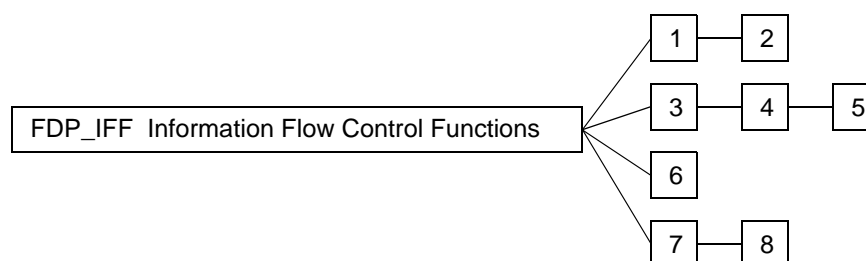
D R A F T

FDP_IFF Information Flow Control Functions

Family behaviour

208 This component specifies the requirements on functions with respect to the information flow control SFPs. 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 them from occurrence.

Component levelling



209 FDP_IFF.1 Simple Security Attributes requires security attributes on containers of information, and on active recipients of information. It specifies the key rules that must be enforced by the function, and describes how security attributes are derived by the function.

210 FDP_IFF.2 Hierarchical Security Attributes expand on the requirements of FDP_IFF.1 Simple Security Attributes by requiring that all information flow control SFPs in the TSP use hierarchical security attributes that form a lattice.

211 FDP_IFF.3 Limited Illicit Information Flows requires the SFP to cover illicit information flows, but not necessarily eliminate them.

212 FDP_IFF.4 Partial Elimination of Illicit Information Flows requires the SFP to cover the elimination of some (but not necessarily all) illicit information flows.

213 FDP_IFF.5 No Illicit Information Flows requires SFP to cover the elimination of all illicit information flows.

214 FDP_IFF.6 Illicit Information Flow Monitoring requires the SFP to monitor illicit information flows for specified and maximum capacities.

215 FDP_IFF.7 Information Flow Authorisation adds the ability to explicitly authorise an information flow.

216 FDP_IFF.8 Information Flow Authorisation and Denial adds the ability to explicitly deny an information flow based upon security attributes, in addition to the requirements of FDP_IFF.7.

D R A F T

Management : for FDP_IFC.1, FDP_IFC.2, FDP_IFF.3, FDP_IFF.4, and FDP_IFF.5

217 There are no management activities foreseen for this component.

Management : for FDP_IFF.6

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

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

Management : for FDP_IFF.7 and FDP_IFF.8

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

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

Audit : for FDP_IFF.1, FDP_IFF.2, and FDP_IFF.5

220 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in a PP/ST:

- a) Minimal: Decisions to permit requested information flows.
- b) Basic: All decisions on requests for information flow.
- c) Detailed: The specific security attributes used in making an information flow enforcement decision.
- d) Detailed: Some specific subsets of the information which has flowed based upon policy goals (e.g. auditing of downgraded material).

Audit : for FDP_IFF.3, FDP_IFF.4, and FDP_IFF.6

221 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in a PP/ST:

- a) Minimal: Decisions to permit requested information flows.
- b) Basic: All decisions on requests for information flow.
- c) Basic: The use of identified illicit information flow channels.
- d) Detailed: The specific security attributes used in making an information flow enforcement decision.
- e) Detailed: Some specific subsets of the information which has flowed based upon policy goals (e.g. auditing of downgraded material).

D R A F T

- f) Detailed: The use of identified illicit information flow channels with estimated maximum capacity exceeding a specified value.

Audit : for FDP_IFF.7 and FDP_IFF.8

222 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in a PP/ST:

- a) Minimal: Successful attempts to specify the authorising or denying of access to an object.
- b) Basic: Unsuccessful attempts to specify the authorising or denying of access to an object.
- c) Detailed: The identity of the user or subject who specifies, or attempts to specify, the authorising or denying of access to an object.

FDP_IFF.1 Simple Security Attributes

Hierarchical to: no other components.

FDP_IFF.1.1 The TSF shall enforce the [assignment: *information flow control SFP*] to enforce at least the following types of subject and object security attributes [assignment: *the minimum number and type of security attributes*].

FDP_IFF.1.2 The TSF shall permit an information flow between a controlled subject and a controlled object via a controlled operation if the following rules hold [assignment: *for each operation, the security attribute-based relationship that must hold between subject and object security attributes*].

FDP_IFF.1.3 The TSF shall enforce the [assignment: *additional information flow control SFP rules*].

FDP_IFF.1.4 The TSF shall enforce the following [assignment: *list of additional SFP capabilities*].

Dependencies : **FDP_IFC.1** Subset Information Flow Control
FMT_MSA.3 Static Attribute Initialisation

FDP_IFF.2 Hierarchical Security Attributes

Hierarchical to: FDP_IFF.1

FDP_IFF.2.1 The TSF shall enforce the [assignment: *information flow control SFP*] to enforce at least the following types of subject and object security attributes [assignment: *the minimum number and type of security attributes*].

FDP_IFF.2.2 The TSF shall permit an information flow between a controlled subject and a controlled object via a controlled operation if the following rules, **based on the**

D R A F T

ordering relationships between security attributes hold; [assignment: *for each operation, the security attribute-based relationship that must hold between subject and object security attributes*].

FDP_IFF.2.3 The TSF shall enforce the [assignment: *additional information flow control SFP rules*].

FDP_IFF.2.4 The TSF shall enforce the following [assignment: *list of additional SFP capabilities*]

FDP_IFF.2.5 The TSF shall enforce the following relationships for any two valid information flow control security attributes:

- a) 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; and
- b) 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; and
- c) 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.

Dependencies : FDP_IFC.1 Subset Information Flow Control
FMT_MSA.3 Static Attribute Initialisation

FDP_IFF.3 Limited Illicit Information Flows

Hierarchical to: no other components.

FDP_IFF.3.1 The TSF shall enforce the [assignment: *information flow control SFP*] to limit the capacity of [assignment: *types of illicit information flows*] to a [assignment: *maximum capacity*].

Dependencies : AVA_CCA.1 Covert channel analysis
FDP_IFC.1 Subset Information Flow Control

D R A F T

FDP_IFF.4 Partial Elimination of Illicit Information Flows

Hierarchical to: FDP_IFF.3

FDP_IFF.4.1 The TSF shall enforce the [assignment: *information flow control SFP*] to limit the capacity of [assignment: *non-empty list of types of illicit information flows*] to a [assignment: *maximum capacity*].

FDP_IFF.4.2 **The TSF shall prevent the following types of** [assignment: *non-empty list of types of illicit information flows*].

Dependencies : AVA_CCA.1 Covert channel analysis

FDP_IFC.1 Subset Information Flow Control

FDP_IFF.5 No Illicit Information Flows

Hierarchical to: FDP_IFF.4

FDP_IFF.5.1 **The TSF shall ensure that no illicit information flows exist to circumvent** [assignment: *name of information flow control SFP*].

Dependencies : AVA_CCA.3 Exhaustive covert channel analysis

FDP_IFC.1 Subset Information Flow Control

FDP_IFF.6 Illicit Information Flow Monitoring

Hierarchical to: no other components.

FDP_IFF.6.1 The TSF shall enforce the [assignment: *information flow control SFP*] to monitor the [assignment: *list of types of illicit information flows*] for the [assignment: *maximum capacity*].

Dependencies : AVA_CCA.1 Covert channel analysis

FDP_IFC.1 Subset Information Flow Control

FDP_IFF.7 Information Flow Authorisation

Hierarchical to: no other components.

FDP_IFF.7.1 **The TSF shall ensure that the information flow control SF that enforces the** [assignment: *information flow control SFP*] **shall explicitly authorise an information flow based on the** [assignment: *value of security attributes of subjects and objects*].

D R A F T

Dependencies : **FDP_IFC.1** Subset Information Flow Control
FMT_MSA.3 Static Attribute Initialisation

FDP_IFF.8 Information Flow Authorisation and Denial

Hierarchical to: FDP_IFF.7

- FDP_IFF.8.1** The TSF shall ensure that the information flow control SF that enforces the [assignment: *information flow control SFP*] shall explicitly authorise an information flow based on the [assignment: *value of security attributes of subjects and objects*].
- FDP_IFF.8.2** **The TSF shall ensure that the information flow control SF that enforces the [assignment: *information flow control SFP*] shall explicitly deny an information flow based on the [assignment: *value of security attributes of subjects and objects*].**

Dependencies : FDP_IFC.1 Subset Information Flow Control
FMT_MSA.3 Static Attribute Initialisation

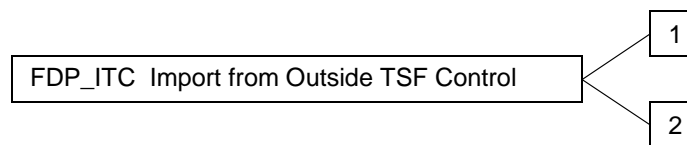
D R A F T

FDP_ITC Import from Outside TSF Control

Family behaviour

- 223 This family defines the mechanisms for introduction 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.

Component levelling



- 224 This family contains two components to address the preservation of security attributes of imported user data for access control and information control policies.

- 225 Component FDP_ITC.1 Import of User Data Without Security Attributes requires that the security attributes correctly represent the user data and are supplied separately from the object.

- 226 Component FDP_ITC.2 Import of User Data with Security Attributes requires that security attributes correctly represent the user data and are accurately and unambiguously associated with the user data imported from outside the TSC.

Management : for FDP_ITC.1 and FDP_ITC.2

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

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

Audit : for FDP_ITC.1 and FDP_ITC.2

- 228 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

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

D R A F T

FDP_ITC.1 Import of User Data Without Security Attributes

Hierarchical to: no other components.

FDP_ITC.1.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] when importing user data, controlled under the SFP, from outside of the TSC.

FDP_ITC.1.2 The TSF shall ignore the security attributes associated with the user data when imported from outside the TSC.

FDP_ITC.1.3 The TSF shall enforce the following [assignment: *additional importation control rules*] when importing user data controlled under the SFP from outside the TSC.

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]
FMT_MSA.3 Static Attribute Initialisation

FDP_ITC.2 Import of User Data with Security Attributes

Hierarchical to: no other components.

FDP_ITC.2.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] when importing user data, controlled under the SFP, from outside of the TSC.

FDP_ITC.2.2 The TSF shall use the security attributes associated with the imported user data.

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.

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 TSF.

FDP_ITC.2.5 The TSF shall enforce the following [assignment: *additional importation control rules*] when importing user data controlled under the SFP from outside the TSC.

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]
[FTP_ITC.1 Inter-TSF Trusted Channel or
FTP_TRP.1 Trusted Path]
FPT_TDC.1 Inter-TSF Basic TSF Data Consistency

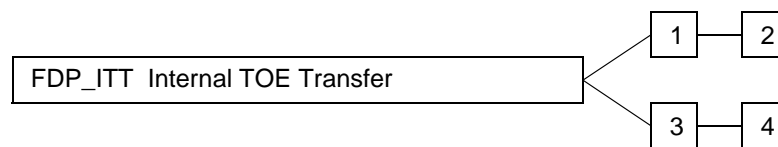
D R A F T

FDP_ITT Internal TOE Transfer

Family behaviour

229 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 and FDP_UIT families, which provide protection for user data when it is transferred between distinct TSFs across an external channel, and FDP_ETC and FDP_ITC, which address transfer of data to or from outside the TSF's control.

Component levelling



230 FDP_ITT.1 Basic Internal Transfer Protection requires that user data be protected when transmitted between parts of the TOE.

231 FDP_ITT.2 Transmission Separation by Attribute requires separation of data based on SFP-relevant attributes in addition to the first component.

232 FDP_ITT.3 Integrity Monitoring requires that the SF monitor user data transmitted between parts of the TOE for identified integrity errors.

233 FDP_ITT.4 Attribute-Based Integrity Monitoring expands on the third component by allowing the form of integrity monitoring to differ by SFP-relevant attribute.

Management : for FDP_ITT.1 and FDP_ITT.2

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

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

Management : for FDP_ITT.3 and FDP_ITT.4

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

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

D R A F T

Audit : for FDP_ITT.1 and FDP_ITT.2

236 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Minimal: Successful transfers of user data, including identification of the transmission channel and protection method.
- b) Basic: All attempts to transfer user data, including identification of the transmission channel used, the protection method used, and any error which occurred.
- c) Basic: Unauthorised attempts to configure the separation mechanism.

Audit : for FDP_ITT.3 and FDP_ITT.4

237 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Minimal: Successful transfers of user data, including identification of the transmission channel and the integrity protection method used.
- b) Basic: All attempts to transfer user data, including identification of the transmission channel used, the integrity protection method used, and any errors which occurred.
- c) Basic: Unauthorised attempts to change the integrity protection method.
- d) Detailed: The action taken upon detection of an integrity error.

FDP_ITT.1 Basic Internal Transfer Protection

Hierarchical to: no other components.

FDP_ITT.1.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] to protect user data from [selection: *disclosure, modification, loss of use*] when it is transmitted between physically-separated parts of the TOE.

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]

FDP_ITT.2 Transmission Separation by Attribute

Hierarchical to: FDP_ITT.1

FDP_ITT.2.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] to protect user data from [selection: *disclosure, modification, loss of use*] when it is transmitted between physically-separated parts of the TOE.

D R A F T

FDP_ITT.2.2 The TSF shall provide separate transmission channels for data controlled by the SFP and transmitted between physically-separated parts of the TOE based on the following [assignment: *security attributes that require separate transmission channels*].

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]

FDP_ITT.3 Integrity Monitoring

Hierarchical to: no other components.

FDP_ITT.3.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] to monitor user data transmitted between physically-separated parts of the TOE for [assignment: *integrity errors*].

FDP_ITT.3.2 Upon detection of a data integrity error, the TSF shall [assignment: *specify the action to be taken upon integrity error*].

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control
FDP_ITT.1 Basic Internal Transfer Protection

FDP_ITT.4 Attribute-Based Integrity Monitoring

Hierarchical to: FDP_ITT.3

FDP_ITT.4.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] to monitor user data transmitted between physically-separated parts of the TOE for [assignment: *integrity errors*], **based on the following** [assignment: *security attributes that require separate transmission channels*].

FDP_ITT.4.2 Upon detection of a data integrity error, the TSF shall [assignment: *specify the action to be taken upon integrity error*].

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]
FDP_ITT.2 Transmission Separation by Attribute

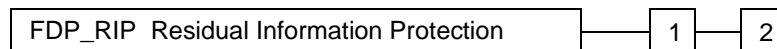
D R A F T

FDP_RIP Residual Information Protection

Family behaviour

238 This family addresses the need to ensure that deleted information is no longer accessible, and that newly created objects do not contain information that should not be accessible. This family requires protection for information that has been logically deleted or released, but may still be present within the TOE.

Component levelling



239 FDP_RIP.1 Subset Residual Information Protection requires that the TSF ensure that any residual information content of any resources is unavailable to a defined subset of the objects in the TSC upon the resource's allocation or deallocation.

240 FDP_RIP.2 Full Residual Information Protection requires that the TSF ensure that any residual information content of any resources is unavailable to all objects upon the resource's allocation or deallocation.

Management : for FDP_RIP.1 and FDP_RIP.2

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

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

Audit : for FDP_RIP.1 and FDP_RIP.2

242 There are no events identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

FDP_RIP.1 Subset Residual Information Protection

Hierarchical to: no other components.

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the [selection: *allocation of the resource to, deallocation of the resource from*] [assignment: *list of objects*].

Dependencies : No dependencies.

D R A F T

FDP_RIP.2 Full Residual Information Protection

Hierarchical to: **FDP_RIP.1**

FDP_RIP.2.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the [selection: *allocation of the resource to, deallocation of the resource from*] **all objects**.

Dependencies : No dependencies.

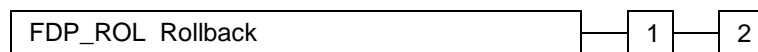
D R A F T

FDP_ROL Rollback

Family behaviour

- 243 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.

Component levelling



- 244 FDP_ROL.1 Basic Rollback addresses a need to roll back or undo a limited number of operations within the defined bounds.
- 245 FDP_ROL.2 Advanced Rollback addresses the need to roll back or undo all operations within the defined bounds.

Management : for FDP_ROL.1 and FDP_ROL.2

- 246 The following actions could be considered for the management functions in FMT Management:
- a) The boundary limit to which rollback may be performed could be a configurable item within the TOE.
 - b) Permission to perform a rollback operation could be restricted to a well defined role.

Audit : for FDP_ROL.1 and FDP_ROL.2

- 247 The following events should be auditable if FAU_GEN Security Audit Data Generation is specified in the PP/ST:
- a) Minimal: All successful rollback operations.
 - b) Basic: All attempts to perform rollback operations.
 - c) Detailed: All attempts to perform rollback operations, including identification of the types of operations rolled back.

D R A F T

FDP_ROL.1 Basic Rollback

Hierarchical to: no other components.

FDP_ROL.1.1 The TSF shall enforce [assignment: *access control SFP and/or information flow control SFP*] to permit the rollback of the [assignment: *list of operations*] on the [assignment: *list of objects*].

FDP_ROL.1.2 The TSF shall permit operations to be rolled back within the [assignment: *boundary limit to which rollback may be performed*].

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]

FDP_ROL.2 Advanced Rollback

Hierarchical to: FDP_ROL.1

FDP_ROL.2.1 The TSF shall enforce [selection: *access control SFP and/or information flow control SFP*] to permit the rollback of **all the operations** on the [assignment: *list of objects*].

FDP_ROL.2.2 The TSF shall permit operations to be rolled back within the [assignment: *boundary limit to which rollback may be performed*].

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]

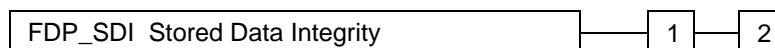
D R A F T

FDP_SDI Stored Data Integrity

Family behaviour

248 This family provides requirements that address protection of user data while it is stored within the TSC. 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.

Component levelling



249 FDP_SDI.1 Stored Data Integrity Monitoring requires that the SF monitor user data stored within the TSC for identified integrity errors.

250 FDP_SDI.2 Stored Data Integrity Monitoring and Action adds the additional capability to the first component by allowing for actions to be taken as a result of an error detection.

Management : for FDP_SDI.1

251 There are no management activities foreseen for this component.

Management : for FDP_SDI.2

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

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

Audit : FDP_SDI.1

253 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Minimal: Successful attempts to check the integrity of user data, including an indication of the results of the check.
- b) Basic: All attempts to check the integrity of user data, including an indication of the results of the check, if performed.
- c) Detailed: The type of integrity error which occurred.

Audit : for FDP_SDI.2

254 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

D R A F T

- a) Minimal: Successful attempts to check the integrity of user data, including an indication of the results of the check.
- b) Basic: All attempts to check the integrity of user data, including an indication of the results of the check, if performed.
- c) Detailed: The type of integrity error which occurred.
- d) Detailed: The action taken upon detection of an integrity error.

FDP_SDI.1 Stored Data Integrity Monitoring

Hierarchical to: no other components.

FDP_SDI.1.1 The TSF shall monitor user data stored within the TSC for [assignment: *integrity errors*] on all objects, based on the following [assignment: *user data attributes*].

Dependencies : No dependencies.

FDP_SDI.2 Stored Data Integrity Monitoring and Action

Hierarchical to: FDP_SDI.1

FDP_SDI.2.1 The TSF shall monitor user data stored within the TSC for [assignment: *integrity errors*] on all objects, based on the following [assignment: *user data attributes*].

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall [assignment: *action to be taken*].

Dependencies : No dependencies.

D R A F T

FDP_UCT Inter-TSF User Data Confidentiality Transfer Protection

Family behaviour

255 This family defines the requirements for ensuring the confidentiality of user data when it is transferred using an external channel between distinct TOEs or users on distinct TOEs.

Component levelling

FDP_UCT Inter-TSF User Data Confidentiality Transfer Protection

1

256 In FDP_UCT.1 Basic Data Exchange Confidentiality, the goal is to provide protection from disclosure of user data while in transit.

Management : for FDP_UCT.1

257 There are no management activities foreseen for this component.

Audit : for FDP_UCT.1

258 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

- a) Minimal: The identity of any user or subject using the data exchange mechanisms.
- b) Basic: The identity of any unauthorised user or subject attempting to use the data exchange mechanisms.
- c) Basic: A reference to the names or other indexing information useful in identifying that data that was transmitted or received. This could include security attributes associated with the information.

FDP_UCT.1 Basic Data Exchange Confidentiality

Hierarchical to: no other components.

FDP_UCT.1.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] to be able to [selection: *transmit, receive*] objects in a manner protected from unauthorised disclosure.

Dependencies : [FTP_ITC.1 Inter-TSF Trusted Channel, or
FTP_TRP.1 Trusted Path]
[FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]

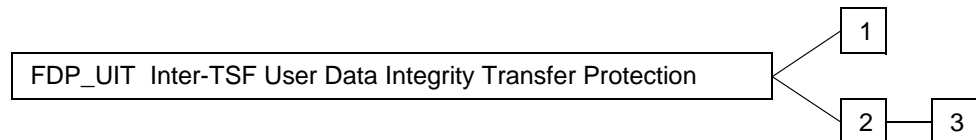
D R A F T

FDP_UIT Inter-TSF User Data Integrity Transfer Protection

Family behaviour

259 This family defines the requirements for protecting user data in transit between the TSF and another Trusted IT product and recovering from detectable errors.

Component levelling



260 FDP_UIT.1 Data Exchange Integrity addresses detection of modifications, deletions, insertions, and replay errors of the user data transmitted.

261 FDP_UIT.2 Source Data Exchange Recovery addresses recovery of the original user data by the receiving TSF with help from the source Trusted IT Product.

262 FDP_UIT.3 Destination Data Exchange Recovery addresses recovery of the original user data by the receiving TSF on its own without any help from the source Trusted IT Product.

Management : for FDP_UIT.1, FDP_UIT.2, and FDP_UIT.3

263 There are no management activities foreseen for this component.

Audit : for FDP_UIT.1

264 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

- a) Minimal: The identity of any user or subject using the data exchange mechanisms.
- b) Basic: The identity of any user or subject attempting to use the user data exchange mechanisms, but who is unauthorised to do so.
- c) Basic: A reference to the names or other indexing information useful in identifying the user data which was transmitted or received. This could include security attributes associated with the user data.
- d) Basic: Any identified attempts to block transmission of user data.
- e) Detailed: The types and/or effects of any detected modifications of transmitted user data.

D R A F T

Audit : for FDP_UIT.2 and FDP_UIT.3

265 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

- a) Minimal: The identity of any user or subject using the data exchange mechanisms.
- b) Minimal: Successful recovery from errors including they type of error that was detected.
- c) Basic: The identity of any user or subject attempting to use the user data exchange mechanisms, but who is unauthorised to do so.
- d) Basic: A reference to the names or other indexing information useful in identifying the user data which was transmitted or received. This could include security attributes associated with the user data.
- e) Basic: Any identified attempts to block transmission of user data.
- f) Detailed: The types and/or effects of any detected modifications of transmitted user data.

FDP_UIT.1 Data Exchange Integrity

Hierarchical to: no other components.

FDP_UIT.1.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] to be able to [selection: *transmit, receive*] user data in a manner protected from undetectable [selection: *modification, deletion, insertion, replay*] errors.

FDP_UIT.1.2 The TSF shall be able to determine on receipt of user data, whether [selection: *modification, deletion, insertion, or replay*] has occurred.

Dependencies : [FTP_ITC.1 Inter-TSF Trusted Channel, or
FTP_TRP.1 Trusted Path]
[FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]

FDP_UIT.2 Source Data Exchange Recovery

Hierarchical to: no other components.

FDP_UIT.2.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] to be able to recover from [assignment: *list of recoverable errors*] with the help of the source Trusted IT Product.

D R A F T

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]
FTP_ITC.1 Inter-TSF Trusted Channel

FDP UIT.3 Destination Data Exchange Recovery

Hierarchical to: FDP UIT.2

FDP UIT.3.1 The TSF shall enforce the [assignment: *access control SFP and/or information flow control SFP*] to be able to recover from [assignment: *list of recoverable errors*] **without any help from the source Trusted IT Product.**

Dependencies : [FDP_ACC.1 Subset Access Control, and/or
FDP_IFC.1 Subset Information Flow Control]
FTP_ITC.1 Inter-TSF Trusted Channel

D R A F T

Class FIA

Identification and Authentication

- 266 Families in this class address the requirements for functions to establish and verify a claimed user identity.
- 267 Identification and Authentication is required to ensure that users are associated with the proper Security Attributes (e.g. identity, groups, roles, security or integrity levels).
- 268 The unambiguous identification of authorised 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 authorised user. Other classes of requirements (e.g. User Data Protection, Security Audit) are dependent upon correct identification and authentication of users in order to be effective.

DRAFT

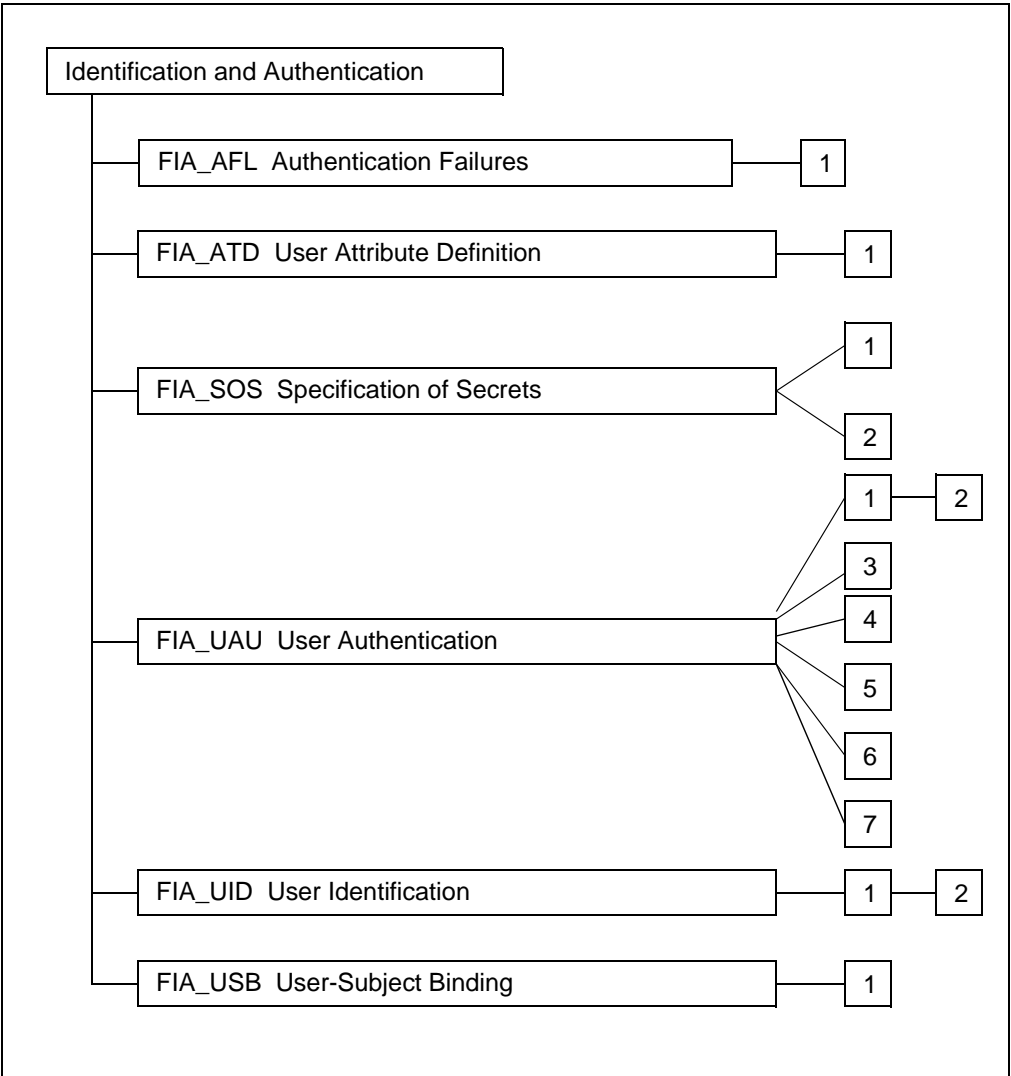


Figure 2.10 - Identification and Authentication class decomposition

D R A F T

FIA_AFL Authentication Failures

Family behaviour

269 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.

Component levelling

FIA_AFL Authentication Failures

1

270 FIA_AFL.1 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 (e.g. workstation) from which the attempts were made until an administrator-defined condition occurs.

Management : for FIA_AFL.1

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

- a) management of the threshold for unsuccessful authentication attempts;
- b) management of actions to be taken in the event of an authentication failure.

Audit : for FIA_AFL.1

272 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: the reaching of the threshold for the unsuccessful authentication attempts and the actions (e.g. dis-abling of a terminal) taken and the subsequent, if appropriate, restoration to the normal state (e.g. re-enabling of a terminal).

FIA_AFL.1 Basic Authentication Failure Handling

273

Hierarchical to: no other components.

FIA_AFL.1.1 The TSF shall detect when [selection: an authorised administrator configurable number of , [assignment: *number*]] unsuccessful authentication attempts occur related to [assignment: list of authentication events].

D R A F T

FIA_AFL.1.2 After the defined number of unsuccessful authentication attempts has been detected, the TSF shall [assignment: list of actions].

Dependencies : **FIA_UAU.1** Timing of authentication

D R A F T

FIA_ATD User Attribute Definition

Family behaviour

274 All authorised users may have a set of security attributes, other than the user's identity, that is used to enforce the TSP. This family defines the requirements for associating user security attributes with users as needed to support the TSP.

Component levelling

FIA_ATD User Attribute Definition

1

275 FIA_ATD.1 User Attribute Definition, allows user security attributes for each user to be maintained individually.

Management : FIA_ATD.1

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

- a) if so indicated in the assignment, the authorised administrator might be able to define additional security attributes for users.

Audit : FIA_ATD.1

277 There are no actions identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

FIA_ATD.1 User Attribute Definition

Hierarchical to: no other components.

FIA_ATD.1.1 The TSF shall maintain [assignment: list of security attributes] belonging to individual users.

Dependencies : No dependencies.

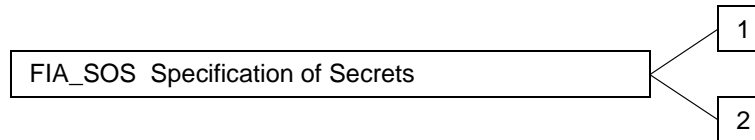
D R A F T

FIA_SOS Specification of Secrets

Family behaviour

278 This family defines requirements for mechanisms that enforce defined quality metrics on provided secrets and generate secrets to satisfy the defined metric.

Component levelling



279 FIA_SOS.1 Verification of Secrets requires the TSF to verify that secrets meet defined quality metrics.

280 FIA_SOS.2 TSF Generation of Secrets requires the TSF to be able to generate secrets that meet defined quality metrics.

Management : FIA_SOS.1

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

- a) the management of the metric used to verify the secrets.

Management : FIA_SOS.2

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

- a) the management of the metric used to generate the secrets.

Audit :

283 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Rejection by the TSF of any tested secret.
- b) Basic: Rejection or acceptance by the TSF of any tested secret.
- c) Detailed: Identification of any changes to the defined quality metrics.

FIA_SOS.1 Verification of Secrets

Hierarchical to: no other components.

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet [assignment: *a defined quality metric*].

D R A F T

Dependencies : No dependencies.

FIA_SOS.2 TSF Generation of Secrets

Hierarchical to: no other components.

FIA_SOS.2.1 The TSF shall provide a mechanism to generate secrets that meet [assignment: *a defined quality metric*].

FIA_SOS.2.2 The TSF shall be able to enforce the use of TSF generated secrets for [assignment: *list of TSF functions*].

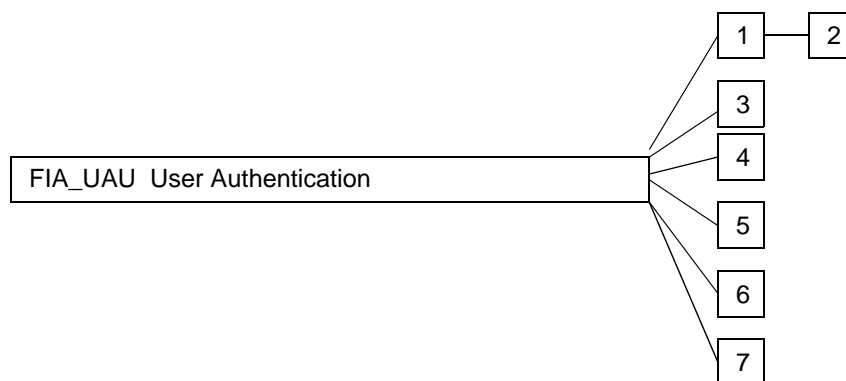
Dependencies : No dependencies.

D R A F T**FIA_UAU User Authentication**

Family behaviour

284 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 must be based.

Component levelling



285 FIA_UAU.1 Timing of authentication, allows a user to perform certain actions prior to the authentication of the user's identity.

286 FIA_UAU.2 User authentication before any action, requires that users authenticate themselves before any action will be allowed by the TSF.

287 FIA_UAU.3 Unforgeable Authentication, requires the authentication mechanism to be able to detect and prevent the use of authentication data that has been forged or copied.

288 FIA_UAU.4 Single-use Authentication Mechanisms, requires an authentication mechanism that operates with single-use authentication data.

289 FIA_UAU.5 Multiple Authentication Mechanisms, requires that different authentication mechanisms be provided and used to authenticate user identities for specific events.

290 FIA_UAU.6 Re-authenticating, requires the ability to specify events under which the user need to be re-authenticated.

291 FIA_UAU.7 Protected authentication feedback, require that only limited feedback information is provided to the user during the authentication.

Management : FIA_UAU.1

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

- a) management of the authentication data by an administrator;

D R A F T

- b) management of the authentication data by the associated user;
- c) managing the list of actions that can be taken before the user is authenticated;

Management : FIA_UAU.2

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

- a) management of the authentication data by an administrator;
- b) management of the authentication data by the user associated with this data.

Management : FIA_UAU.3, FIA_UAU.4 and FIA_UAU.7

294 There are no management activities foreseen.

Management : FIA_UAU.5

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

- a) the management of authentication mechanisms;
- b) the management of the rules for authentication.

Management : FIA_UAU.6

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

- a) if an authorised administrator could request re-authentication, the management includes a re-authentication request.

Audit : FIA_UAU.1

297 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Unsuccessful use of the authentication mechanism.
- b) Basic: All use of the authentication mechanism.
- c) Detailed: All TSF mediated actions performed before authentication of the user.

Audit : FIA_UAU.2

298 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Unsuccessful use of the authentication mechanism.
- b) Basic: All use of the authentication mechanism.

D R A F T

Audit : FIA_UAU.3

299 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Detection of fraudulent authentication data.
- b) Basic: All immediate measures taken and results of checks on the fraudulent data.

Audit : FIA_UAU.4

300 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Attempts to reuse authentication data.

Audit : FIA_UAU.5

301 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: The final decision on authentication.
- b) Basic: The result of each activated mechanism together with the final decision.

Audit : FIA_UAU.6

302 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Failure of reauthentication.
- b) Basic: All reauthentication attempts.

Audit : FIA_UAU.7

303 There are no auditable events foreseen.

FIA_UAU.1 Timing of authentication

Hierarchical to: no other components.

FIA_UAU.1.1 The TSF shall allow [assignment: *list of TSF mediated actions*] on behalf of the user to be performed before the user is authenticated.

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.

Dependencies : **FIA_UID.1 Timing of Identification**

D R A F T

FIA_UAU.2 User authentication before any action

Hierarchical to: FIA_UAU.1 Timing of authentication

FIA_UAU.2.1 The TSF shall require each user to be successfully authenticated before allowing **any TSF-mediated actions** on behalf of that user.

Dependencies : FIA_UID.1 Timing of Identification

FIA_UAU.3 Unforgeable Authentication

Hierarchical to: no other components.

FIA_UAU.3.1 The TSF shall detect and prevent use of authentication data that has been forged by any user of the TSF.

FIA_UAU.3.2 The TSF shall detect and prevent use of authentication data that has been copied from any other user of the TSF.

Dependencies : No dependencies.

FIA_UAU.4 Single-use Authentication Mechanisms

Hierarchical to: no other components.

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to [assignment: *identified authentication mechanism(s)*].

Dependencies : No dependencies.

FIA_UAU.5 Multiple Authentication Mechanisms

Hierarchical to: no other components.

FIA_UAU.5.1 The TSF shall provide [assignment: list of multiple authentication mechanisms] to support user authentication.

FIA_UAU.5.2 The TSF shall authenticate any user's claimed identity according to the [assignment: rules describing how the multiple authentication mechanisms provide authentication].

Dependencies : No dependencies.

D R A F T

FIA_UAU.6 Re-authenticating

Hierarchical to: no other components.

FIA_UAU.6.1 The TSF shall re-authenticate the user under the conditions [assignment: list of conditions under which re-authentication is required].

Dependencies : No dependencies.

FIA_UAU.7 Protected authentication feedback

Hierarchical to: no other components.

FIA_UAU.7.1 The TSF shall only provide [Assignment: list of feedback] to the user while the authentication is in progress.

Dependencies : FIA_UAU.1 Timing of authentication

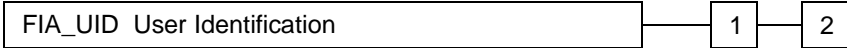
D R A F T

FIA_UID User Identification

Family behaviour

304 This family defines the conditions under which users shall be required to identify themselves before performing any other actions that are to be mediated by the TSF and which require user identification.

Component levelling



FIA_UID User Identification

1

2

305 FIA_UID.1 Timing of Identification, allows users to perform certain actions before being identified by the TSF.

306 FIA_UID.2 User Identification before any action, require that users identify themselves before any action will be allowed by the TSF.

Management : FIA_UID.1

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

- a) the management of the user identities;
- b) if an authorised administrator can change the actions allowed before identification, the managing of the action lists.

Management : FIA_UID.2

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

- a) the management of the user identities.

Audit : FIA_UID.1 and FIA_UID.2

309 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Unsuccessful use of the user identification mechanism, including the user identity provided.
- b) Basic: All use of the user identification mechanism, including the user identity provided.

D R A F T

FIA_UID.1 Timing of Identification

Hierarchical to: no other components.

FIA_UID.1.1 The TSF shall allow [assignment: *list of TSF-mediated actions*] on behalf of the user to be performed before the user is identified.

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.

Dependencies : No dependencies.

FIA_UID.2 User Identification before any action

Hierarchical to: FIA_UID.1 Timing of Identification

FIA_UID.2.1 The TSF shall require each user to identify itself before allowing **any TSF-mediated actions** on behalf of that user.

Dependencies : No dependencies.

D R A F T

FIA_USB User-Subject Binding

Family behaviour

- 310 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.

Component levelling

FIA_USB User-Subject Binding

1

- 311 FIA_USB.1 User-Subject Binding requires the maintenance of an association between the user's security attributes and a subject acting on the user's behalf.

Management :

- 312 The following actions could be considered for the management functions in FMT:
- a) an authorised administrator can define default subject security attributes.

Audit :

- 313 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:
- a) Minimal: Unsuccessful binding of user security attributes to a subject (e.g. creation of a subject).
 - b) Basic: Success and failure of binding of user security attributes to a subject (e.g. success and failure to create a subject).

FIA_USB.1 User-Subject Binding

Hierarchical to: no other components.

- FIA_USB.1.1 The TSF shall associate the appropriate user security attributes with subjects acting on behalf of that user.**

Dependencies : **FIA_ATD.1 User Attribute Definition**

D R A F T

D R A F T

Class FMT

Security Management

- 314 This class is intended to specify the management of the several aspects of the TSF. The aspects consist of the attributes (security attributes), TSF data and functions. The different roles with respect to management and their interaction, such as separation of capability, can be specified.
- 315 This class has several objectives:
- a) management of TSF data, which include for example banners;
 - b) management of security attributes, which include for example the Access Control Lists, Capability Lists etc.;
 - c) management of functions of the TSF, which includes for example the selection of functions, and rules or conditions influencing the behaviour of the TSF function;
 - d) definition of security roles.

D R A F T

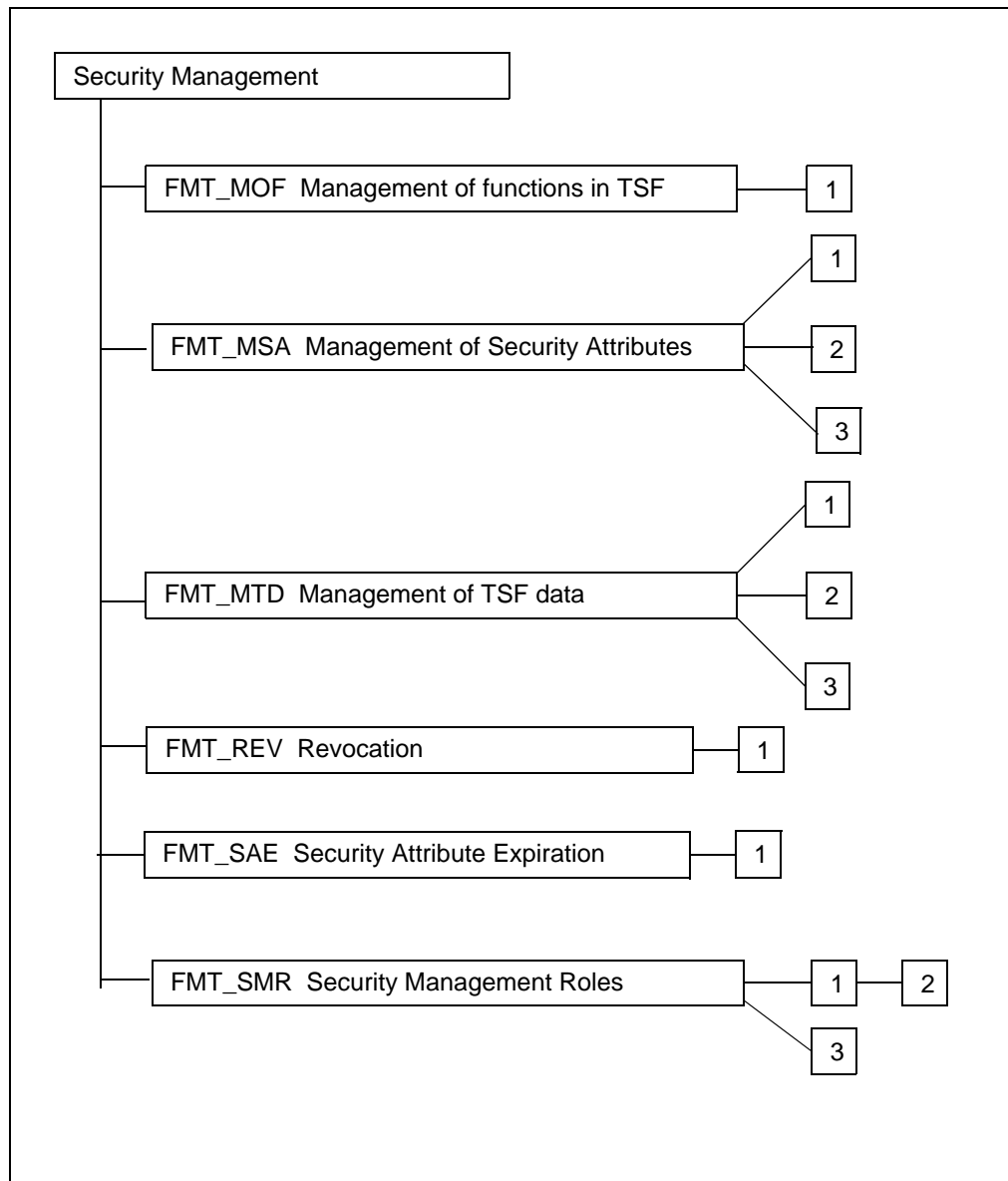


Figure 2.11 - Security Management class decomposition

D R A F T

New FMT Component	Implements old components	Name
FMT-MSA.1	FDP_ACI.1 FDP_ACI.2, FDP_ACI.3, FDP_ACI.4, FDP_ACI.5, FDP_SAM.1, FDP_SAM.2, FDP_SAQ.2, FRU_PRS.3, FRU_RSA.3, FIA_ADA, FIA_ATA	Management of security attributes
FMT-MSA.2	FDP_ACI.4, FDP_ACI.5, FDP_ACI.1, FDP_SAM.3	safe security attributes
FMT-MSA.3	FDP_ACI.4	static initialisation
FMT-MTD.1	FAU_MGT.1	Management of TSF Data
FMT-MTD.2	FAU_MGT.2, FAU_MGT.3, FPT_TSM	Management of controls on TSF Data
FMT-MTD.3		safe TSF Data
FMT-MOF.1	<Any with rules or conditions>	Management of security functions behaviour
FMT-SMF.1	FPT_TSA, FPT_TSM	Security roles
FMT-SMF.2		Restriction of Security roles
FMT-SMF.3		Explicit taking a role.
Refinement of one of the above	FAU_MGT.4	

Table 2.1 - Mapping from CC 1.0 to 2.0 draft for security management requirements

Editor Note: THIS TABLE IS FOR REVIEW PURPOSES ONLY AND WILL NOT BE INCLUDED IN VERSION 2.0 FINAL).

D R A F T

FMT_MOF Management of functions in TSF

Family behaviour

316 This family allows authorised users control over the management of functions in the TSF. Examples of functions in the TSF include the audit functions and the multiple authentication functions.

Component levelling

FMT_MOF Management of functions in TSF

1

317 FMT_MOF.1 Management of security functions behaviour allows the authorised users (roles) to manage the behaviour of functions in the TSF which use rules or have specified conditions that may be modifiable.

Management : FMT_MOF.1

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

- a) managing the group of roles/users that can interact with the functions in the TSF;

Audit : FMT_MOF.1

319 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Basic: All modifications in the behaviour of the functions in the TSF.

FMT_MOF.1 Management of security functions behaviour

Component rationale and application notes

Hierarchical to: no other components.

FMT_MOF.1.1 The TSF shall restrict the ability to [selection: *determine, disable, enable, modify*] the behaviour of the functions [assignment: *list of functions and potential modification*] to [assignment: *the authorised identified roles*].

Dependencies : **FMT_SMR.1 Security roles**

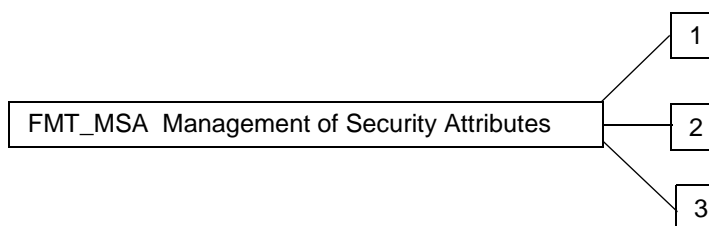
D R A F T

FMT_MSA Management of Security Attributes

Family behaviour

320 This family allows authorised users control over the management of user attributes. This management includes capabilities for viewing and modifying of security attributes.

Component levelling



321 FMT_MSA.1 Management of security attributes allows the authorised administrator or other designated persons to modify the values of specified security attributes.

322 FMT_MSA.2 Safe security attributes ensures that the security attributes are valid with respect to the secure state.

323 FMT_MSA.3 Static Attribute Initialisation ensures that the default values of security attributes are either permissive or restrictive in nature.

Management : FMT_MSA.1

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

- a) managing the group of roles/users that can interact with the security attributes.

Management : FMT_MSA.2

325 There are no additional management activities foreseen for this component.

Management : FMT_MSA.3

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

- a) managing the group of roles/users that can specify initial values;
- b) managing the permissive and restrictive setting of default values for a given access control SFP.

D R A F T

Audit : FMT_MSA.1

327 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Basic: All modifications of the values of security attributes.

Audit : FMT_MSA.2

328 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: All offered and rejected values for a security attribute;
- b) Detailed: All offered and accepted safe values for a security attribute.

Audit : FMT_MSA.3

329 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Basic: Modifications of the default setting of permissive or restrictive rules.
- b) Basic: All modifications of the initial values of security attributes.

FMT_MSA.1 Management of security attributes

Hierarchical to: no other components.

FMT_MSA.1.1 The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] to restrict the ability to [selection: *change_default, read, modify, delete*] the values of the security attributes [assignment: *list of security attributes*] to [assignment: *the authorised identified roles*].

Dependencies : [FDP_ACC.1 Subset Access Control or
FDP_IFC.1 Subset Information Flow Control]
FMT_SMR.1 Security roles

FMT_MSA.2 Safe security attributes

Hierarchical to: no other components.

FMT_MSA.2.1 The TSF shall ensure that only safe values are accepted for security attributes.

D R A F T

Dependencies : ADV_SPM.1 Informal TOE security policy model
[FDP_ACC.1 Subset Access Control or
FDP_IFC.1 Subset Information Flow Control]
FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

FMT_MSA.3 Static Attribute Initialisation

Hierarchical to: no other components.

FMT_MSA.3.1 The TSF shall enforce the [assignment: *access control SFP, information flow control SFP*] to provide [selection: *restrictive, permissive, other property*] default values for object security attributes that are used to enforce the *SFP*.

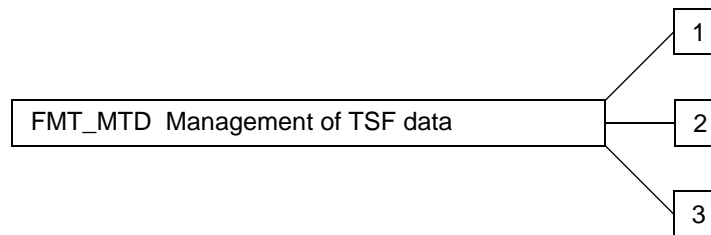
FMT_MSA.3.2 The TSF shall allow the [assignment: *the authorised identified roles*] to specify alternate initial values to override the default values when an object is created.

Dependencies : ADV_SPM.1 Informal TOE security policy model
FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

D R A F T

FMT_MTD Management of TSF data**Family behaviour**

330 This family allows authorised users control over the management of TSF data. Examples of TSF data include audit information, clock, system configuration and other TSF configuration parameters.

Component levelling

331 FMT_MTD.1 Management of TSF data allows the authorised administrator or other designated persons to manage TSF data.

332 FMT_MTD.2 Management of limits on TSF data specifies the action to be taken if limits on TSF data are reached or exceeded.

333 FMT_MTD.3 Safe TSF data ensures that the TSF data are valid with respect to the secure state.

Management : FMT_MTD.1

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

- a) managing the group of roles/users that can interact with the TSF data.

Management : FMT_MTD.2

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

- a) managing the group of roles/users that can interact with the limits on the TSF data.

Management : FMT_MTD.3

336 There are no additional management activities foreseen for this component.

Audit : FMT_MTD.1

337 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

D R A F T

- a) Basic: All modifications to the values of TSF data.

Audit : FMT_MTD.2

338 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Basic: All modifications to the limits on TSF data;
- b) Basic: All modifications in the actions to be taken in case of violation of the limits.

Audit : FMT_MTD.3

339 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: All rejected values of TSF data.

FMT_MTD.1 Management of TSF data

Hierarchical to: no other components.

FMT_MTD.1.1 The TSF shall restrict the ability to [selection: *change_default, read, modify, delete, clear*] the [assignment: *list of TSF data*] to [assignment: *the authorised identified roles*].

Dependencies : FMT_SMR.1 Security roles

FMT_MTD.2 Management of limits on TSF data

Hierarchical to: no other components.

FMT_MTD.2.1 The TSF shall restrict the specification of the limits for [assignment: *list of TSF data*] to [assignment: *the authorised identified roles*].

FMT_MTD.2.2 The TSF shall perform [assignment: *actions to be taken*] in case the TSF data is at, or exceeds the indicated limits.

Dependencies : FMT_MTD.1 Management of TSF data
FMT_SMR.1 Security roles

FMT_MTD.3 Safe TSF data

Hierarchical to: no other components.

FMT_MTD.3.1 The TSF shall ensure that only safe values are accepted for TSF data.

D R A F T

Dependencies : **ADV_SPM.1 Informal TOE security policy model**
FMT_MTD.1 Management of TSF data

D R A F T

FMT_REV Revocation

Family behaviour

340 This family addresses revocation of security attributes for a variety of entities within a TOE.

Component levelling

FMT_REV Revocation

1

341 FMT_REV.1 Revocation, provides for revocation of security attributes to be enforced at some point in time.

Management : FMT_REV.1

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

- a) managing the group of roles/users that can invoke revocation of security attributes;
- b) managing the lists of users, subjects, objects and other resources for which revocation is possible;
- c) managing the rules which constitutes the revocation.

Audit : FMT_REV.1

343 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Basic: unsuccessful revocation of security attributes.
- b) Minimal: All attempts to revoke security attributes.

FMT_REV.1 Revocation

Hierarchical to: no other components.

FMT_REV.1.1 The TSF shall restrict the ability to revoke security attributes associated with the [selection: *users, subjects, objects, other additional resources*] within the TSC to [assignment: *the authorised identified roles*].

FMT_REV.1.2 The TSF shall enforce revocation [assignment: *specification of revocation rules*].

Dependencies : **FMT_SMR.1 Security roles**

D R A F T

FMT_SAE Security Attribute Expiration

Family behaviour

344 This family addresses the capability to enforce time limits for the validity of security attributes.

Component levelling

FMT_SAE Security Attribute Expiration

1

345 This family consists of only one component, FMT_SAE.1 Time-Limited Authorisation, which requires the ability for the authorised administrator to specify an expiration time on specified security attributes.

Management : FMT_SAE.1

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

- a) managing the list of security attributes for which expiration is to be supported;
- b) the actions to be taken if the expiration time has passed.

Audit : FMT_SAE.1

347 The following actions should be audited if FAU Security Audit is included in the PP/ST:

- a) Basic: Specification of the expiration time for an attribute
- b) Basic: Action taken due to attribute expiration

FMT_SAE.1 Time-Limited Authorisation

Hierarchical to: no other components.

FMT_SAE.1.1 The TSF shall restrict the ability to specify an expiration time for [assignment: *list of security attributes for which expiration is to be supported*] to [assignment: *the authorised identified roles*].

FMT_SAE.1.2 For each of these security attributes, the TSF shall be able to [assignment: *list of actions to be taken for each security attribute*] after the expiration time for the indicated security attribute has passed.

Dependencies : FMT_SMR.1 Security roles

FPT_STM.1 Reliable Time Stamps

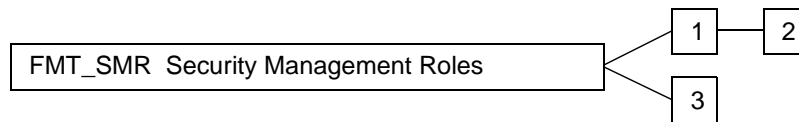
D R A F T

FMT_SMR Security Management Roles

Family behaviour

348 This family is intended to control the assignment of different roles to users. The capabilities of these roles with respect to security management is described in the other families in this class.

Component levelling



349 FMT_SMR.1 Security roles specifies the roles with respect to security that the TSF recognises.

350 FMT_SMR.2 Restrictions on security roles specifies that in addition to the specification of the roles there are rules that control the relationship between the roles.

351 FMT_SMR.3 Assuming roles requires that an explicit request is given to the TSF to assume a role.

Management : FMT_SMR.1

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

- a) managing the group of users that are part of a role.

Management : FMT_SMR.2

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

- a) managing the group of users that are part of a role;
- b) managing the conditions that the roles must satisfy.

Management : FMT_SMR.3

354 There are no additional management activities foreseen for this component.

Audit : FMT_SMR.1

355 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

D R A F T

- a) minimal: modifications to the group of users that are part of a role;
- b) detailed: every use of the rights of a role.

Audit : FMT_SMR.2

356 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) minimal: modifications to the group of users that are part of a role;
- b) minimal: unsuccessful attempts to use a role due to the given conditions on the roles.
- c) detailed: every use of the rights of a role.

Audit : FMT_SMR.3

357 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) minimal: explicit request to assume a role.

FMT_SMR.1 Security roles

Hierarchical to: no other components.

FMT_SMR.1.1 The TSF shall maintain the roles [assignment: *the authorised identified roles*].

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

Dependencies : FIA_UID.1 Timing of Identification

FMT_SMR.2 Restrictions on security roles

Hierarchical to: **FMT_SMR.1 Security roles**

FMT_SMR.2.1 The TSF shall maintain the roles [assignment: *the authorised identified roles*].

FMT_SMR.2.2 The TSF shall be able to associate users with roles.

FMT_SMR.2.3 The TSF shall ensure that [assignment: *conditions for the different roles*] are satisfied.

Dependencies : No dependencies.

D R A F T

FMT_SMR.3 Assuming roles

Hierarchical to: no other components.

FMT_SMR.3.1 The TSF shall require an explicit request to be made in order for an authorised user to assume the roles [assignment: *the role*].

Dependencies : **FMT_SMR.1** Security roles

D R A F T

D R A F T

Class FPR

Privacy

358 This class contains privacy requirements. These requirements provide a user
protection against discovery and misuse of his identity by other users.

359 This class is based on the current available knowledge about privacy techniques.
Since research in this area is still ongoing, in the future these components might
need expansion or revision.

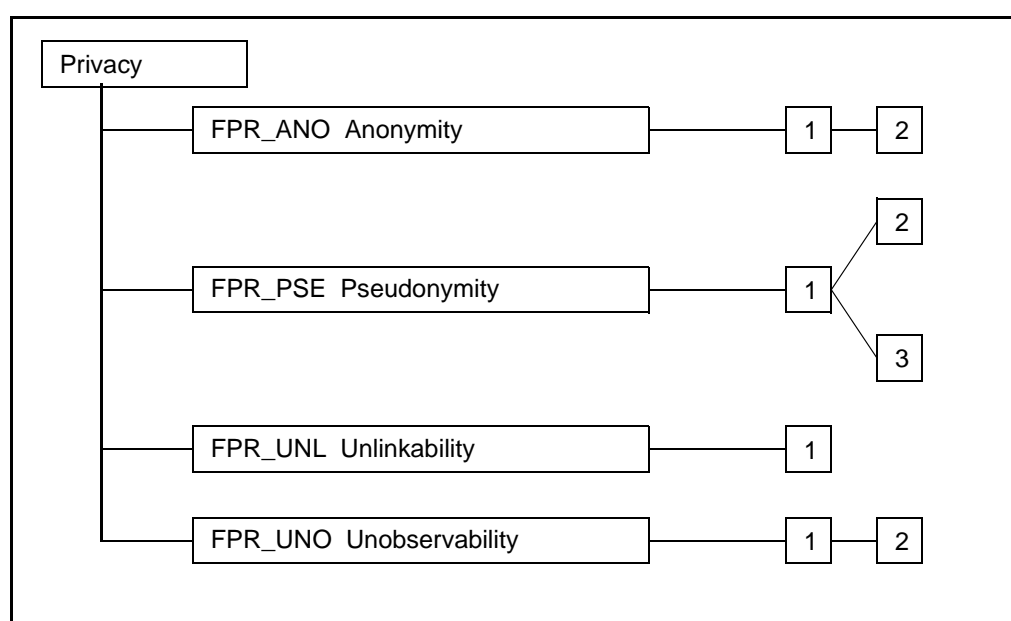


Figure 2.12 - Privacy class decomposition

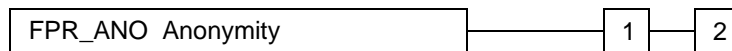
D R A F T

FPR_ANO Anonymity

Family behaviour

360 This family ensures that a user may use a resource or service without disclosing the user's identity. The requirements for Anonymity provide protection of the user identity. Anonymity is not intended to protect the subject identity.

Component levelling



361 FPR_ANO.1 Anonymity requires that other users or subjects are unable to determine the identity of a user bound to a subject or operation.

362 FPR_ANO.2 TSF Anonymity enhances the requirements of FPR_ANO.1 by ensuring that the TSF does not ask for the user identity.

Management :

363 There are no management activities foreseen for these components.

Audit :

364 The following actions shall be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: The invocation of the anonymity mechanism.

FPR_ANO.1 Anonymity

Hierarchical to: no other components.

FPR_ANO.1.1 The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to determine the user identity bound to [assignment: *list of subjects and/or operations*].

Dependencies : No dependencies.

D R A F T

FPR_ANO.2 TSF Anonymity

Hierarchical to: FPR_ANO.1

FPR_ANO.2.1 The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to determine the user identity bound to [assignment: *list of subjects and/or operations*].

FPR_ANO.2.2 **The TSF shall not solicit any reference to the user identity in order to initiate actions on behalf of [assignment: *list of subjects*] or subjects requesting [assignment: *list of operations*].**

Dependencies : No dependencies.

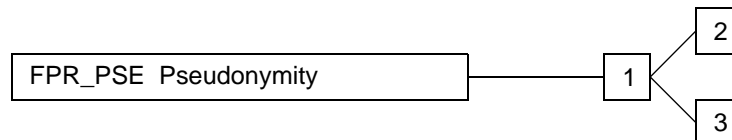
D R A F T

FPR_PSE Pseudonymity

Family behaviour

365 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.

Component levelling



366 FPR_PSE.1 Pseudonymity 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 this user is still accountable for its actions.

367 FPR_PSE.2 Reversible Pseudonymity requires the TSF to provide a capability to determine the original user identity based on a provided alias.

368 FPR_PSE.3 Alias Pseudonymity requires the TSF to follow certain construction rules for the alias to the user identity.

Management :

369 There are no management activities foreseen for these components.

Audit :

370 The following actions shall be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: The subject /user which requested resolution of the user identity should be audited.

FPR_PSE.1 Pseudonymity

Hierarchical to: no other components.

FPR_PSE.1.1 The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to determine the user identity bound to [assignment: *list of subjects and/or operations and/or objects*].

FPR_PSE.1.2 The TSF shall be able to provide [assignment: *number of aliases*] aliases of the user identity to [assignment: *list of subjects*].

D R A F T

FPR_PSE.1.3 The TSF shall [selection: *determine an alias for a user, accept the alias from the user*] and verify that it conforms to the [assignment: *alias metric*].

Dependencies : No dependencies.

FPR_PSE.2 Reversible Pseudonymity

Hierarchical to: FPR_PSE.1

FPR_PSE.2.1 The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to determine the user identity bound to [assignment: *list of subjects and/or operations and/or objects*].

FPR_PSE.2.2 The TSF shall be able to provide [assignment: *number of aliases*] aliases of the user identity to [assignment: *list of subjects*].

FPR_PSE.2.3 The TSF shall [selection: *determine an alias for a user, accept the alias from the user*] and verify that it conforms to the [assignment: *alias metric*].

FPR_PSE.2.4 The TSF shall provide [selection: *an authorised administrator, [assignment: list of trusted subjects]*] a capability to determine the user identity based on the provided alias only under the [assignment: *list of conditions*].

Dependencies : **FIA_UID.1 Timing of Identification**

FPR_PSE.3 Alias Pseudonymity

Hierarchical to: FPR_PSE.1

FPR_PSE.3.1 The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to determine the user identity bound to [assignment: *list of subjects and/or operations and/or objects*].

FPR_PSE.3.2 The TSF shall be able to provide [assignment: *number of aliases*] aliases of the user identity to [assignment: *list of subjects*].

FPR_PSE.3.3 The TSF shall [selection: *determine an alias for a user, accept the alias from the user*] and verify that it conforms to the [assignment: *alias metric*].

FPR_PSE.3.4 The TSF shall provide an alias to the user identity which shall be identical to an alias provided previously under the following [assignment: *list of conditions*] and unrelated to previously provided aliases otherwise.

Dependencies : No dependencies.

D R A F T

FPR_UNL Unlinkability

Family behaviour

371 This family ensures that a user may make multiple uses of resources or services without others being able to link these uses together.

Component levelling

FPR_UNL Unlinkability

1

372 FPR_UNL.1 Unlinkability requires that users and/or subjects are unable to determine whether the same user caused certain specific operations in the system.

Management :

373 There are no management activities foreseen for this component.

Audit :

374 The following actions shall be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: The invocation of the unlinkability mechanism.

FPR_UNL.1 Unlinkability

Hierarchical to: no other components.

FPR_UNL.1.1 The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to determine whether [assignment: *list of operations*] [selection: *‘were caused by the same user’, ‘are related as follows* [assignment: *list of relations*’]].

Dependencies : No dependencies.

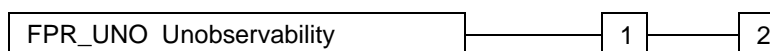
D R A F T

FPR_UNO Unobservability

Family behaviour:

375 This family ensures that a subject may use a resource or service without others, especially third parties, being able to observe that the resource or service is being used.

Component levelling



376 FPR_UNO.1 Unobservability requires that users and/or subjects cannot determine whether an object is being used.

377 FPR_UNO.2 Authorised Administrator Observability requires the TSF to provide the authorised administrator with a capability to observe the usage of resources and/or services.

Management :

378 There are no management activities foreseen for these components.

Audit :

379 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: The invocation of the unobservability mechanism.

FPR_UNO.1 Unobservability

Hierarchical to: no other components.

FPR_UNO.1.1 The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to observe the operation [assignment: *list of operations*] on [assignment: *list of objects*] by another user or subject.

Dependencies : No dependencies.

D R A F T

FPR_UNO.2 Authorised Administrator Observability

Hierarchical to: FPR_UNO.1

FPR_UNO.2.1 The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to observe the operation [assignment: *list of operations*] on [assignment: *list of objects*] by another user or subject.

FPR_UNO.2.2 **The TSF shall provide an authorised administrator with the capability to observe the usage of resources and/or services.**

Dependencies : No dependencies.

Class FPT

Protection of the TOE Security Functions

380 This class contains families of functional requirements that relate to the integrity and management of the mechanisms that provide the TSF (independent of TSP-specifics) and to the integrity of TSF data (independent of the specific contents of the TSP data). In some sense, families in this class may appear to duplicate components in the FDP (User Data Protection) class; they may even be implemented using the same mechanisms. However, FDP focuses on user data protection, while FPT focuses on TSF data protection. In fact, components from the FPT class are necessary even in the absence of any user data protection, to provide confidence in the enforcement of other policies (such as accountability) that may be specified in the PP/ST.

381 From the point of view of this class, there are three significant portions for the TSF:

- a) The TSF's *abstract machine*, which is the virtual or physical machine upon which the specific TSF software under evaluation executes.
- b) The TSF's *software*, which executes on the abstract machine and implements the mechanisms that enforce the TSP.
- c) The TSF's *data*, which are the administrative databases that guide the enforcement of the TSP.

D R A F T

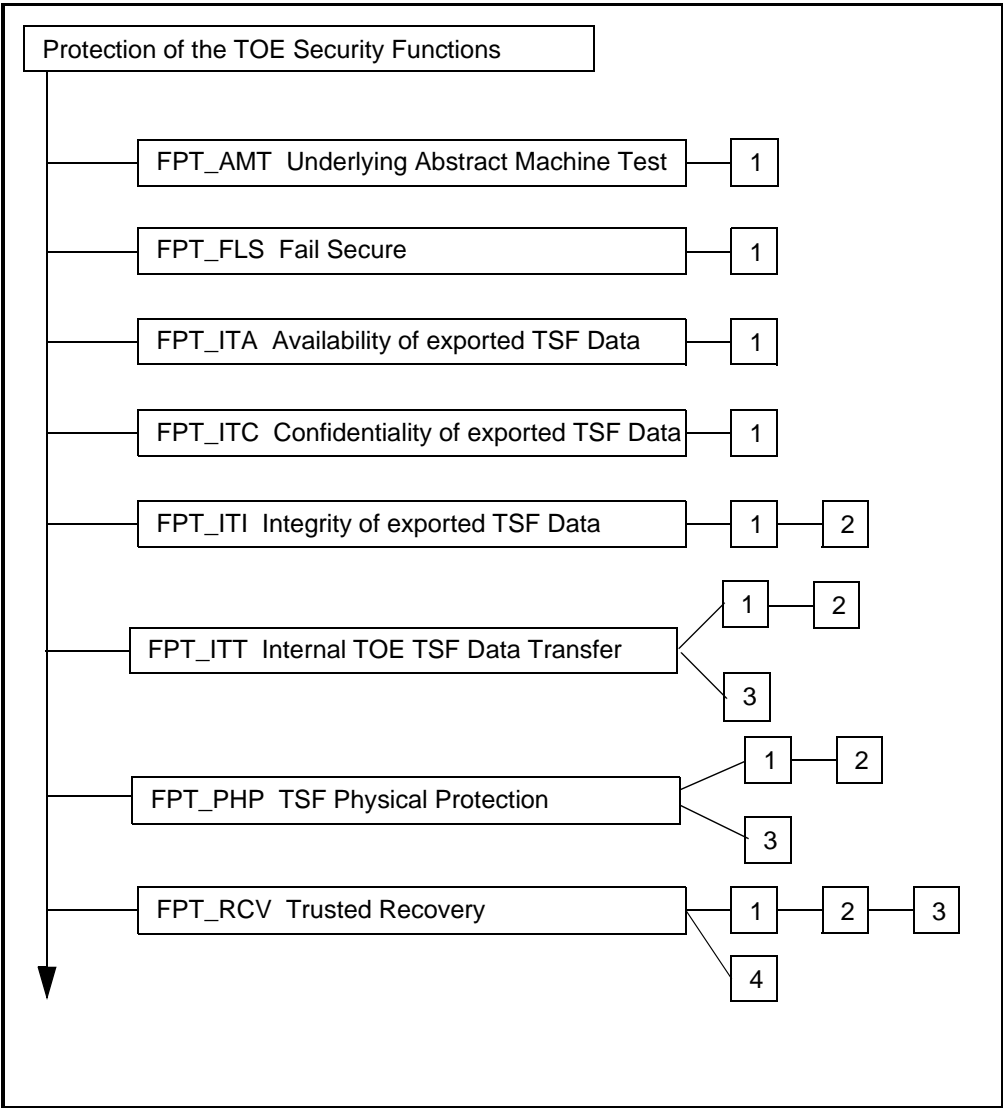


Figure 2.13 - Protection of the TOE Security Functions class decomposition

D R A F T

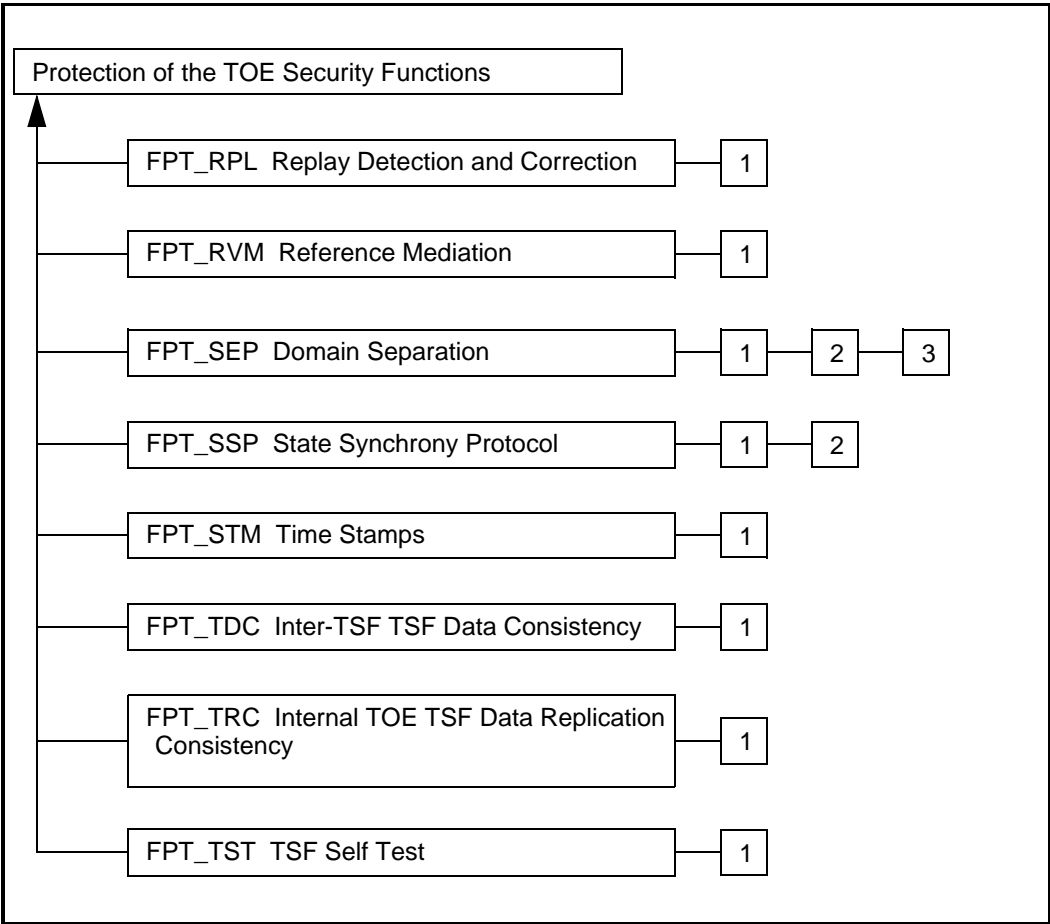


Figure 2.14 - Protection of the TOE Security Functions class decomposition (Cont.)

D R A F T

FPT_AMT Underlying Abstract Machine Test

Family behaviour

382 This family defines requirements for the TSF to perform testing to demonstrate the security assumptions made about the underlying abstract machine upon which the TSF relies. This “abstract” machine could be a hardware/firmware platform, or it could be some known and assessed hardware/software combination acting as a virtual machine.

Component levelling

FPT_AMT Underlying Abstract Machine Test

1

383 FPT_AMT.1 Abstract Machine Testing, provides for testing of the underlying abstract machine.

Management : FPT_AMT.1

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

- a) management of the conditions under which abstract machine test occurs, such as during initial start-up, regular interval, or under specified conditions;
- b) management of the time interval if appropriate.

Audit : FPT_AMT.1

385 The following actions should be audited if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Basic: Execution of the tests of the underlying machine and the results of the tests.

FPT_AMT.1 Abstract Machine Testing

Hierarchical to: no other components.

FPT_AMT.1.1 The TSF shall run a suite of tests [selection: *during initial start-up, periodically during normal operation, at the request of the authorised administrator, other conditions*] to demonstrate the correct operation of the security functions provided by the abstract machine which underlies the TSF.

Dependencies : No dependencies.

D R A F T

FPT_FLS Fail Secure

Family behaviour

386 The requirements of this family ensure that the TOE will not violate its TSP in the event of identified categories of failures in the TSF.

Component levelling

FPT_FLS Fail Secure

1

387 This family consists of only one component, FPT_FLS.1 Failure with Preservation of Secure State, which requires that the TSF maintain a secure state in the face of the identified failures.

Management : FPT_FLS.1

388 There are no management activities foreseen.

Audit : FPT_FLS.1

389 The following actions should be audited if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Basic: Failure of the TSF.

FPT_FLS.1 Failure with Preservation of Secure State

Hierarchical to: no other components.

FPT_FLS.1.1 The TSF shall preserve a secure state when [assignment: *list of types of TSF failures*] occur.

Dependencies : **ADV_SPM.1 Informal TOE security policy model**

D R A F T

FPT_ITA Availability of exported TSF Data

Family behaviour

390 This family defines the rules for the prevention of loss of availability of TSF data moving between the TSF and a remote trusted IT product. This data could, for example, be TSF critical data such as passwords, keys, audit data, or TSF executable code.

Component levelling

FPT_ITA Availability of exported TSF Data

1

391 This family consists of only one component, FPT_ITA.1 Inter-TSF Availability Within a Defined Availability Metric, which requires that the TSF ensure, to an identified degree of probability, that TSF data made available between TSFs can be obtained by the receiving TSF.

Management : FPT_ITA.1

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

- a) management of the list of types of TSF data that must be available to a remote trusted IT product.

Audit : FPT_ITA.1

393 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: the absence of TSF data when required by a TOE.

FPT_ITA.1 Inter-TSF Availability Within a Defined Availability Metric

Hierarchical to: no other components.

FPT_ITA.1.1 The TSF shall ensure the availability of [assignment: *list of types of TSF data*] provided to a remote trusted IT product within [assignment: *a defined availability metric*] given the conditions [assignment: *conditions to ensure availability*].

Dependencies : No dependencies.

D R A F T

FPT_ITC Confidentiality of exported TSF Data

Family behaviour

394 This family defines the rules for the protection from unauthorised disclosure of TSF data moving between the TSF and a remote trusted IT product. This data could, for example, be TSF critical data such as passwords, keys, audit data, or TSF executable code.

Component levelling

FPT_ITC Confidentiality of exported TSF Data	1
--	---

395 This family consists of only one component, FPT_ITC.1 Inter-TSF Confidentiality During Transmission, which requires that the TSF ensure that data transmitted between TSFs is protected from disclosure while in transit.

Management : FPT_ITC.1

396 There are no management activities foreseen.

Audit : FPT_ITC.1

397 There are no actions identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

FPT_ITC.1 Inter-TSF Confidentiality During Transmission

Hierarchical to: no other components.

FPT_ITC.1.1 The TSF shall protect all TSF data transmitted from the TSF to a remote trusted IT product from unauthorised disclosure.

Dependencies : No dependencies.

D R A F T

FPT_ITI Integrity of exported TSF Data

Family behaviour

398 This family defines the rules for the protection, from unauthorised modification, of TSF data moving between the TSF and a remote trusted IT product. This data could, for example, be TSF critical data such as passwords, keys, audit data, or TSF executable code.

Component levelling



399 FPT_ITI.1 Inter-TSF Detection of Modification, provides the ability to detect modification of TSF data when it is transmitted to a remote trusted IT product, under the assumption that the remote trusted IT product is cognisant of the mechanism used.

400 FPT_ITI.2 Inter-TSF Detection and Correction of Modification, provides the ability for the remote trusted IT product not only to detect modification, but to correct modified TSF data under the assumption that the remote trusted IT product is cognisant of the mechanism used.

Management : FPT_ITI.1

401 There are no management activities foreseen.

Management : FPT_ITI.2

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

- a) management of the types of TSF data that the TSF should try to correct if modified in transit.

Audit : FPT_ITI.1

403 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: the detection of modification of transmitted TSF data.

Audit : FPT_ITI.2

404 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: the detection of modification of transmitted TSF data;

D R A F T

- b) Basic: the correction of modified transmitted TSF data.

FPT_ITI.1 Inter-TSF Detection of Modification

Hierarchical to: no other components.

FPT_ITI.1.1 The TSF shall provide the capability to detect modification within [assignment: *a defined modification metric*] of all TSF data transmitted between the TSF and a remote trusted IT product.

FPT_ITI.1.2 The TSF shall verify the integrity of all TSF data transmitted between the TSF and a remote trusted IT product and perform [assignment: *action to be taken*] in case modifications are detected.

Dependencies : No dependencies.

FPT_ITI.2 Inter-TSF Detection and Correction of Modification

Hierarchical to: FPT_ITI.1

FPT_ITI.2.1 The TSF shall provide the capability to detect modification within [assignment: *a defined modification metric*] of all TSF data transmitted between the TSF and a remote trusted IT product.

FPT_ITI.2.2 The TSF shall verify the integrity of all TSF data transmitted between the TSF and a remote trusted IT product and perform [assignment: *action to be taken*] in case modifications are detected.

FPT_ITI.2.3 The TSF shall provide the capability to correct [assignment: *type of modification*] of all TSF data transmitted between the TSF and a remote trusted IT product.

Dependencies : No dependencies.

D R A F T

FPT_ITT Internal TOE TSF Data Transfer

Family behaviour

405 This family provides requirements that address protection of TSF data when it is transferred between parts of a TOE across an internal channel.

Component levelling



406 FPT_ITT.1 Basic Internal TSF Data Transfer Protection, requires that TSF data be protected when transmitted between parts of the TOE.

407 FPT_ITT.2 TSF Data Transfer Separation, requires that the TSF separate user data from TSF data during transmission.

408 FPT_ITT.3 TSF Data Integrity Monitoring, requires that the TSF data transmitted between parts of the TOE is monitored for identified integrity errors.

Management : FPT_ITT.1

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

- a) management of the types of modification the TSF should protect against;
- b) management of the mechanism used to provide the protection of the data in transit between different parts of the TSF.

Management : FPT_ITT.2

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

- a) management of the types of modification the TSF should protect against;
- b) management of the mechanism used to provide the protection of the data in transit between different parts of the TSF;
- c) management of the separation mechanism.

Management : FPT_ITT.3

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

- a) management of the types of modification the TSF should protect against;

D R A F T

- b) management of the mechanism used to provide the protection of the data in transit between different parts of the TSF;
- c) management of the types of modification of TSF data the TSF should try to detect;
- d) management of the actions that will be taken.

Audit : FPT_ITT.1, FPT_ITT.2

412 There are no actions identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

Audit : FPT_ITT.3

413 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: the detection of modification of TSF data.

FPT_ITT.1 Basic Internal TSF Data Transfer Protection

Hierarchical to: no other components.

FPT_ITT.1.1 The TSF shall protect TSF data from [selection: *disclosure, modification*] when it is transmitted between physically-separated parts of the TOE.

Dependencies : No dependencies.

FPT_ITT.2 TSF Data Transfer Separation

Hierarchical to: FPT_ITT.1

FPT_ITT.2.1 The TSF shall protect TSF data from [selection: *disclosure, modification*] when it is transmitted between physically-separated parts of the TOE.

FPT_ITT.2.2 The TSF shall separate user data from TSF data when such data is transmitted between physically-separated parts of the TOE.

Dependencies : No dependencies.

D R A F T

FPT_ITT.3 TSF Data Integrity Monitoring

Hierarchical to: no other components.

FPT_ITT.3.1 The TSF shall be able to detect [selection: *modification of data, substitution of data, re-ordering of data, deletion of data, other integrity errors*] for TSF data transmitted between physically-separated parts of the TOE.

FPT_ITT.3.2 Upon detection of a data integrity error, the TSF shall [assignment: *specify the action to be taken*].

Dependencies : FPT_ITT.1 Basic Internal TSF Data Transfer Protection

D R A F T

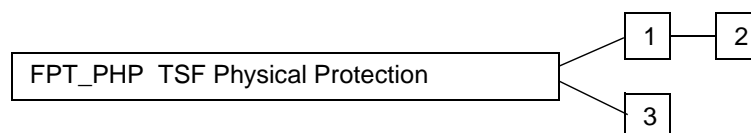
FPT_PHP TSF Physical Protection

Family behaviour

414 TSF physical protection components refer to restrictions on unauthorised physical access to the TSF, and to the deterrence of, and resistance to, unauthorised physical modification, or substitution of the TSF.

415 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 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 family also provides requirements regarding how the TSF shall respond to physical tampering attempts.

Component levelling



416 FPT_PHP.1 Passive Detection of Physical Attack, provides for features that indicate when a TSF device or element is subject to tampering. However, notification of a tampering attack is not automatic; an authorised administrator must invoke a security administrative function or perform manual inspection to determining if tampering has occurred.

417 FPT_PHP.2 Notification of Physical Attack, provides for automatic notification of tampering attacks for an identified subset of physical penetrations.

418 FPT_PHP.3 Resistance to Physical Attack, provides for features that prevent or resist physical tampering with TSF devices and elements.

Management : FPT_PHP.1, FPT_PHP.3

419 There are no management activities foreseen.

Management : FPT_PHP.2

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

- a) management of the user or role that gets informed about intrusions;
- b) management of the list of devices that should inform the indicated user or role about the intrusion.

D R A F T

Audit : FPT_PHP.1

421 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: if detection by IT means, detection of intrusion.

Audit : FPT_PHP.2,

422 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: detection of intrusion.

Audit : FPT_PHP.3

423 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) None.

FPT_PHP.1 Passive Detection of Physical Attack

Hierarchical to: no other components.

FPT_PHP.1.1 The TOE shall provide unambiguous detection of physical tampering that might compromise the TSF.

FPT_PHP.1.2 The TSF shall provide the capability to determine whether physical tampering with the TSF's devices and elements has occurred.

Dependencies : **FMT_MOF.1 Management of security functions behaviour**

FPT_PHP.2 Notification of Physical Attack

Hierarchical to: FPT_PHP.1

FPT_PHP.2.1 The TOE shall provide unambiguous detection of physical tampering that might compromise the TSF.

FPT_PHP.2.2 The TSF shall provide the capability to determine whether physical tampering with the TSF's devices and elements has occurred.

FPT_PHP.2.3 For [assignment: *list of devices/elements for which active detection is required*], the TSF shall monitor the devices and elements and notify [assignment: *a designated user or role*] when physical tampering with the TSF's devices and elements has occurred.

D R A F T

Dependencies : FMT_MOF.1 Management of security functions behaviour

FPT_PHP.3 Resistance to Physical Attack

Hierarchical to: no other components.

FPT_PHP.3.1 The TOE shall resist identified physical tampering attacks to the [assignment: *list of devices/elements, physical tampering attack scenarios, work factors for which resistance to attack is required*]

FPT_PHP.3.2 The TOE shall respond automatically to physical attacks to [assignment: *list of devices/elements, physical tampering attack scenarios for which automatic response to attack is required*] in such a way as to ensure that the TSP is not violated

Dependencies : No dependencies.

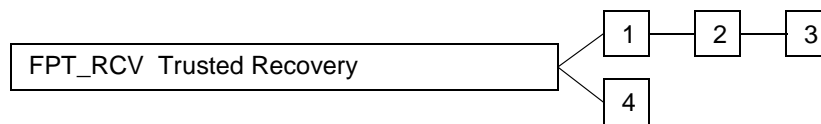
D R A F T

FPT_RCV Trusted Recovery

Family behaviour

424 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. Satisfying the requirements of this family establishes that the initial and recovered states of the TSF satisfy the requirements. This family is important because the start-up state of the TSF determines the protection of subsequent states.

Component levelling



425 FPT_RCV.1 Manual Recovery, allows a TOE to provide only mechanisms that involve human intervention to return to a secure state.

426 FPT_RCV.2 Automated Recovery, provides, for at least one type of service discontinuity, recovery to a secure state without human intervention; recovery for other discontinuities may require human intervention.

427 FPT_RCV.3 Automated Recovery without Undue Loss, also provides for automated recovery, but strengthens the requirements by disallowing undue loss of protected objects.

428 FPT_RCV.4 Function Recovery, provides for recovery at the level of particular SFs, ensuring either successful completion or rollback of TSF data to a secure state.

Management : FPT_RCV.1, FPT_RCV.4

429 There are no management activities foreseen.

Management : FPT_RCV.2, FPT_RCV.3

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

- a) management of the list of failures/service discontinuities that will be handled through the automatic procedures.

Audit : FPT_RCV.1, FPT_RCV.2, FPT_RCV.3

431 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: the fact that a failure or service discontinuity occurred;

D R A F T

- b) Minimal: resumption of the regular operation;
- c) Basic: type of failure or service discontinuity.

Audit : FPT_RCV.4

432 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: if possible, the impossibility to return to a secure state after failure of a security function;
- b) Basic: if possible, the detection of a failure of a security function.

FPT_RCV.1 Manual Recovery

Hierarchical to: no other components.

FPT_RCV.1.1 After a failure or service discontinuity, the TSF shall enter a maintenance mode where the ability to return the TOE to a secure state is provided.

FPT_RCV.1.2 The TSF shall provide the authorised administrator with the capability to restore the TSF data to a consistent and secure state.

Dependencies : **FPT_TST.1 TSF Testing**

FMT_SMR.1 Security roles

AGD_ADM.1 Administrator guidance

ADV_SPM.1 Informal TOE security policy model

FPT_RCV.2 Automated Recovery

Hierarchical to: FPT_RCV.1

FPT_RCV.2.1 When automated recovery from a failure or service discontinuity is not possible, the TSF shall enter a maintenance mode where the ability to return the TOE to a secure state is provided.

FPT_RCV.2.2 The TSF shall provide the authorised administrator with the capability to restore the TSF data to a consistent and secure state.

FPT_RCV.2.3 For [assignment: *list of failures/service discontinuities*], the TSF shall ensure the return of the TOE to a secure state using automated procedures.

D R A F T

Dependencies : FMT_SMR.1 Security roles
FPT_TST.1 TSF Testing
AGD_ADM.1 Administrator guidance
ADV_SPM.1 Informal TOE security policy model

FPT_RCV.3 Automated Recovery without Undue Loss

Hierarchical to: FPT_RCV.2

- FPT_RCV.3.1** When automated recovery from a failure or service discontinuity is not possible, the TSF shall enter a maintenance mode where the ability to return the TOE to a secure state is provided.
- FPT_RCV.3.2** The TSF shall provide the authorised administrator with the capability to restore the TSF data to a consistent and secure state.
- FPT_RCV.3.3** For [assignment: *list of failures/service discontinuities*], the TSF shall ensure the return of the TOE to a secure state using automated procedures.
- FPT_RCV.3.4** **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: *quantification*] for loss of TSF data or objects within the TSC.**
- FPT_RCV.3.5** **The TSF shall provide the capability to determine the objects that were or were not capable of being recovered.**

Dependencies : FMT_SMR.1 Security roles
FPT_TST.1 TSF Testing
AGD_ADM.1 Administrator guidance
ADV_SPM.1 Informal TOE security policy model

FPT_RCV.4 Function Recovery

Hierarchical to: no other components.

- FPT_RCV.4.1** **The TSF shall ensure that [assignment: *list of SFs and failure scenarios*] have the property that the SF either completes successfully, or for the indicated failure scenarios, recovers to a consistent and secure state.**

Dependencies : **ADV_SPM.1 Informal TOE security policy model**

D R A F T

FPT_RPL Replay Detection and Correction

Family behaviour

433 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.

Component levelling

FPT_RPL Replay Detection and Correction

1

434 The family consists of only one component, FPT_RPL.1 Replay Detection, which requires that the TSF shall be able to detect the replay of identified entities (e.g. messages, service requests, service responses).

Management : FPT_RPL.1

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

- a) management of the list of identified entities for which replay shall be detected;
- b) management of the list of actions that need to be taken in case of relay.

Audit : FPT_RPL.1

436 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Basic: Detected replay attacks.
- b) Detailed: Action to be taken based on the specific actions.

FPT_RPL.1 Replay Detection

Hierarchical to: no other components.

FPT_RPL.1.1 The TSF shall detect replay for [assignment: *list of identified entities*].

FPT_RPL.1.2 The TSF shall perform [assignment: *list of specific actions*] when replay is detected.

Dependencies : No dependencies.

D R A F T

FPT_RVM Reference Mediation

Family behaviour

437 The requirements of this family address the “always invoked” aspect of a traditional reference monitor. The goal of these components is to ensure, with respect to a given SFP, that all actions requiring policy enforcement are validated by the TSF against the SFP. If the portion of the TSF that enforces the SFP also meets the requirements of appropriate components from FPT_SEP (Domain Separation) and ADV_INT (TSF internals), then that portion of the TSF provides a “reference monitor” for that SFP.

438 A TSF that implements a SFP provides effective protection against unauthorised operation if and only if all enforceable actions (e.g. accesses to objects) requested by untrusted subjects with respect to any or all of that SFP are validated by the TSF before succeeding. If an action that could be enforceable by the TSF, is incorrectly enforced or incorrectly bypassed, the overall enforcement of the SFP could be compromised. Subjects could then bypass the SFP in a variety of unauthorised ways (e.g. circumvent access checks for some subjects or objects, bypass checks for objects whose protection was assumed by applications, retain access rights beyond their intended lifetime, bypass auditing of audited actions, or bypass authentication). Note that some subjects, the so called “trusted subjects” with respect to a specific SFP, might be trusted to enforce the SFP by themselves, and bypass the mediation of the SFP.

Component levelling

FPT_RVM Reference Mediation

1

439 This family consists of only one component, FPT_RVM.1 Non-Bypassability of the TSP, which requires non-bypassability for all SFPs in the TSP.

Management : FPT_RVM.1

440 There are no management activities foreseen.

Audit : FPT_RVM.1

441 There are no actions identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

FPT_RVM.1 Non-Bypassability of the TSP

Hierarchical to: no other components.

FPT_RVM.1.1 The TSF shall ensure that TSP enforcement functions are invoked and succeed before assignment operation within the TSC is allowed to proceed.

D R A F T

Dependencies : No dependencies.

D R A F T

FPT_SEP Domain Separation

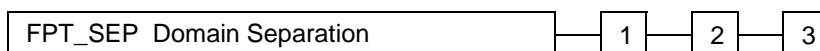
Family behaviour

442 The components of this family ensure that at least one security domain is available for the TSF's own execution and that the TSF is protected from external interference and tampering (e.g. by modification of TSF code or data structures) by untrusted subjects. Satisfying the requirements of this family makes the TSF self-protecting, meaning that an untrusted subject cannot modify or damage the TSF.

443 This family requires the following:

- a) The resources of the TSF's security domain ("protected domain") and those of subjects and unconstrained entities external to the domain are separated such that the entities external to the protected domain cannot observe or modify TSF data or TSF code internal to the protected domain.
- b) The transfers between domains are controlled such that arbitrary entry to, or return from, the protected domain is not possible.
- c) The user or application parameters passed to the protected domain by addresses are validated with respect to the protected domain's address space, and those passed by value are validated with respect to the values expected by the protected domain.
- d) The security domains of subjects are distinct except for controlled sharing via the TSF.

Component levelling



444 FPT_SEP.1 TSF Domain Separation, provides a distinct protected domain for the TSF and provides separation between subjects within the TSC.

445 FPT_SEP.2 SFP Domain Separation, requires that the TSF be further subdivided, with distinct domain(s) for an identified set of SFPs that act as reference monitors for their policies, and a domain for the remainder of the TSF, as well as domains for the non-TSF portions of the TOE.

446 FPT_SEP.3 Complete Reference Monitor, requires that there be distinct domain(s) for TSP enforcement, a domain for the remainder of the TSF, as well as domains for the non-TSF portions of the TOE.

Management : FPT_SEP.1, FPT_SEP.2, FPT_SEP.3

447 There are no management activities foreseen.

D R A F T

Audit : FPT_SEP.1, FPT_SEP.2, FPT_SEP.3

448 There are no actions identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

FPT_SEP.1 TSF Domain Separation

Hierarchical to: no other components.

FPT_SEP.1.1 The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

FPT_SEP.1.2 The TSF shall enforce separation between the security domains of subjects in the TSC.

Dependencies : No dependencies.

FPT_SEP.2 SFP Domain Separation

Hierarchical to: FPT_SEP.1

FPT_SEP.2.1 The unisolated portion of the TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

FPT_SEP.2.2 The TSF shall enforce separation between the security domains of subjects in the TSC.

FPT_SEP.2.3 The TSF shall maintain the part of the TSF related to [assignment: *list of access control and/or information flow control SFPs*] in a security domain for their own execution that protects them from interference and tampering by the remainder of the TSF and by subjects untrusted with respect to those SFPs.

Dependencies : No dependencies.

FPT_SEP.3 Complete Reference Monitor

Hierarchical to: FPT_SEP.2

FPT_SEP.3.1 The unisolated portion of the TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

FPT_SEP.3.2 The TSF shall enforce separation between the security domains of subjects in the TSC.

FPT_SEP.3.3 The TSF shall maintain the part of the TSF that enforces the access control and/or information flow control SFPs in a security domain for its own execution that

D R A F T

protects **them** from interference and tampering by the remainder of the TSF and by subjects untrusted with respect to **the TSP**.

Dependencies : No dependencies.

D R A F T

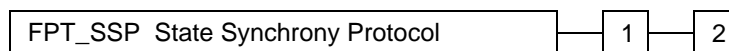
FPT_SSP State Synchrony Protocol

Family behaviour

449 Distributed systems may give rise to greater complexity than monolithic systems through the potential for differences in state between parts of the system, and through delays in communication. In most cases synchronisation 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.

450 FPT_SSP establishes the requirement for certain critical security functions of the TSF to use this trusted protocol. FPT_SSP ensures that two distributed parts of the TOE (e.g. hosts) have synchronised their states after a security-relevant action.

Component levelling



451 FPT_SSP.1 Simple Trusted Acknowledgement requires only a simple acknowledgment by the data recipient.

452 FPT_SSP.2 Mutual Trusted Acknowledgement requires mutual acknowledgment of the data exchange.

Management : FPT_SSP.1, FPT_SSP.2

453 There are no management activities foreseen.

Audit : FPT_SSP.1, FPT_SSP.2

454 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: failure to receive an acknowledgement when expected.

FPT_SSP.1 Simple Trusted Acknowledgement

Hierarchical to: no other components.

FPT_SSP.1.1 The TSF shall acknowledge, when requested by another part of the TSF, the receipt of an unmodified TSF data transmission.

Dependencies : **FPT_ITT.1 Basic Internal TSF Data Transfer Protection**

D R A F T

FPT_SSP.2 Mutual Trusted Acknowledgement

Hierarchical to: FPT_SSP.1

FPT_SSP.2.1 The TSF shall acknowledge, when requested by another part of the TSF, the receipt of an unmodified TSF data transmission.

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.**

Dependencies : FPT_ITT.1 Basic Internal TSF Data Transfer Protection

D R A F T

FPT_STM Time Stamps

Family behaviour

454 This family addresses requirements for a reliable time stamp function within a TOE.

Component levelling

FPT_STM Time Stamps

1

454 This family consists of only one component, FPT_STM.1 Reliable Time Stamps, which requires that the TSF provide reliable time stamps for TSF functions.

Management : FPT_STM.1

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

a) management of the time.

Audit : FPT_STM.1

456 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: changes to the time;

b) Detailed: providing a timestamp.

FPT_STM.1 Reliable Time Stamps

Hierarchical to: no other components.

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps for its own use.

D R A F T

FPT_TDC Inter-TSF TSF Data Consistency

Family behaviour

456 In a distributed or composite system environment, a TOE may need to exchange TSF data (e.g. the SFP-attributes associated with data, audit information, identification information) 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.

Component levelling

FPT_TDC Inter-TSF TSF Data Consistency

1

457 FPT_TDC.1 Inter-TSF Basic TSF Data Consistency requires that the TSF provide mechanisms to ensure consistency of attributes between TSFs.

Management : FPT_TDC.1

458 There are no management activities foreseen.

Audit : FPT_TDC.1

459 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Successful use of TSF data consistency mechanisms.
- b) Basic: assignment use of the TSF data consistency mechanisms.
- c) Basic: Identification of which TSF data have been interpreted.
- d) Basic: Detection of modified TSF data.

FPT_TDC.1 Inter-TSF Basic TSF Data Consistency

Hierarchical to: no other components.

FPT_TDC.1.1 The TSF shall enforce the consistent interpretation of [assignment: *list of TSF data types*] between this TSF and another trusted IT product.

FPT_TDC.1.2 The TSF shall use [assignment: *list of interpretation rules to be applied by the TSF*] when interpreting the TSF data from another trusted IT product.

Dependencies : No dependencies.

D R A F T

FPT_TRC Internal TOE TSF Data Replication Consistency

Family behaviour

460 The requirements of this family are needed to ensure the consistency of TSF data when such data is replicated internal to the TOE. Such data may become inconsistent if the internal channel between parts of the TOE becomes inoperative. If the TOE is internally structured as a network and parts of the TOE network connections are broken, this may for instance occur when parts become disabled.

Component levelling

FPT_TRC Internal TOE TSF Data Replication Consistency

1

461 This family consists of only one component, FPT_TRC.1 Internal TSF Consistency, which requires that the TSF ensure the consistency of TSF data that is replicated in multiple locations.

Management : FPT_TRC.1

462 There are no management activities foreseen.

Audit : FPT_TRC.1

463 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: restoring consistency upon reconnection.
- b) Basic: Detected inconsistency between TSF data.

FPT_TRC.1 Internal TSF Consistency

Hierarchical to: no other components.

FPT_TRC.1.1 The TSF shall ensure that TSF data is consistent when replicated between parts of the TOE.

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: *list of SFs dependent on TSF data replication consistency*].

Dependencies : **FPT_ITT.1 Basic Internal TSF Data Transfer Protection**

D R A F T

FPT_TST TSF Self Test**Family behaviour**

464 The family defines the requirements for the self-testing of the TSF with respect to some expected correct operation. Examples are calls 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 authorised administrator, 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.

465 The requirements of this family are also needed to detect the corruption of TSF data and executable code by various failures that do not necessarily stop the TOE's operation (which would be handled by other families). These checks must be performed because these failures may 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.

Component levelling

FPT_TST TSF Self Test

1

466 FPT_TST.1 TSF Testing, provides the ability to test the TSF's correct operation. These tests may be performed at start-up, periodically, at the request of the authorised administrator, or when other conditions are met. It also provides the ability to verify the integrity of TSF data and executable code.

Management : FPT_TST.1

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

- a) management of the conditions under which abstract machine test occurs, such as during initial start-up, regular interval, or under specified conditions;
- b) management of the time interval if appropriate.

468 FPT_TST.1

469 No audit in addition to that required for security administrative functions, if the tests are invoked by the authorised administrator during normal operation.

D R A F T

FPT_TST.1 TSF Testing

Hierarchical to: no other components.

FPT_TST.1.1 The TSF shall run a suite of self tests [selection: *during initial start-up, periodically during normal operation, at the request of the authorised administrator, at the conditions [assignment: conditions at which self test should occur]*] to demonstrate the correct operation of the TSF.

FPT_TST.1.2 The TSF shall provide authorised administrators with the capability to verify the integrity of TSF data.

FPT_TST.1.3 The TSF shall provide authorised administrators with the capability to verify the integrity of stored TSF executable code.

Dependencies : **FPT_AMT.1 Abstract Machine Testing**

D R A F T

DRAFT

Class FRU

Resource Utilisation

470

This class provides three families which support the availability of required resources such as processing capability and/or storage capacity when needed. 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 monopolised by lower priority tasks. The family Resource Allocation provides limits on the use of available resources, therefore preventing users from monopolising the resources.

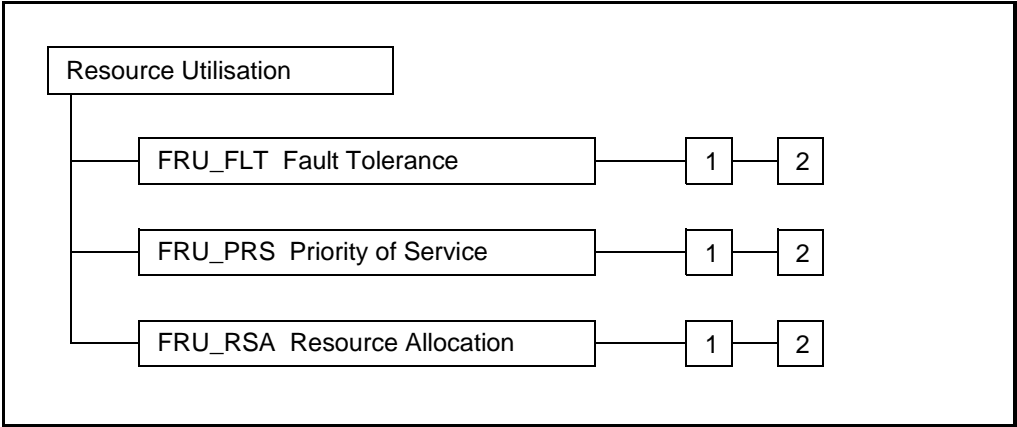


Figure 2.15 - Resource Utilisation class decomposition

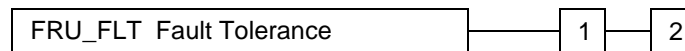
D R A F T

FRU_FLT Fault Tolerance

Family behaviour

471 The requirements of this family ensure that the TOE will maintain correct operation even in the event of failures.

Component levelling



472 FRU_FLT.1 Degraded Fault Tolerance requires the TOE to continue correct operation of identified capabilities in the event of identified failures.

473 FRU_FLT.2 Limited Fault Tolerance requires the TOE to continue correct operation of all capabilities in the event of identified failures.

Management : FRU_FLT.1, FRU_FLT.2

474 There are no management activities foreseen.

Audit : for FRU_FLT.1

475 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Any failure detected by the TSF.
- b) Basic: All TOE capabilities being discontinued due to a failure.

Audit : for FRU_FLT.2

476 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Any failure detected by the TSF.

FRU_FLT.1 Degraded Fault Tolerance

Hierarchical to: no other components.

FRU_FLT.1.1 The TSF shall ensure the operation of [assignment: *list of TOE capabilities*] that will be maintained when [assignment: *list of type of failures*] occur.

Dependencies : **FPT_FLS.1 Failure with Preservation of Secure State**

D R A F T

FRU_FLT.2 Limited Fault Tolerance

Hierarchical to: FRU_FLT.1

FRU_FLT.2.1 The TSF shall ensure the operation of **all the TOE's capabilities** when [assignment: *list of type of failures*] occur.

Dependencies : FPT_FLS.1 Failure with Preservation of Secure State

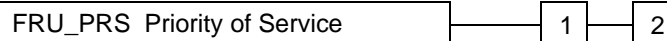
D R A F T

FRU_PRS Priority of Service

Family behaviour

477 The requirements of this family allow the TSF to control the use of resources within the TSC by users and subjects such that high priority activities within the TSC will always be accomplished without undue interference or delay caused by low priority activities.

Component levelling



478 FRU_PRS.1 Limited Priority of Service provides priorities for a subject's use of a subset of the resources within the TSC.

479 FRU_PRS.2 Full Priority of Service provides priorities for a subject's use of all of the resources within the TSC.

Management : for FRU_PRS.1, FRU_PRS.2

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

- a) assignment of priorities to each subject in the TSF;

Audit : for FRU_PRS.1, FRU_PRS.2

481 The following actions shall be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Minimal: Rejection of operation based on the use of priority within an allocation.
- b) Basic: All attempted uses of the allocation function which involves the priority of the service functions.

FRU_PRS.1 Limited Priority of Service

Hierarchical to: no other components.

FRU_PRS.1.1 The TSF shall assign a priority to each subject in the TSF.

FRU_PRS.1.2 The TSF shall ensure that each access to [assignment: *controlled resources*] shall be mediated on the basis of Priority of Service.

Dependencies : No dependencies.

D R A F T

FRU_PRS.2 Full Priority of Service

Hierarchical to: FRU_PRS.1

FRU_PRS.2.1 The TSF shall assign a priority to each subject in the TSF.

FRU_PRS.2.2 The TSF shall ensure that each access to **all shareable resources** shall be mediated on the basis of Priority of Service.

Dependencies : No dependencies.

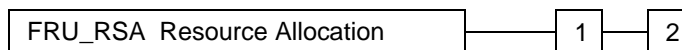
D R A F T

FRU_RSA Resource Allocation

Family behaviour

482 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 unauthorised monopolisation of resources.

Component levelling



483 FRU_RSA.1 Maximum Quotas provides requirements for quota mechanisms that ensure that users and subjects will not monopolize a controlled resource.

484 FRU_RSA.2 Minimum and Maximum Quotas provides requirements for quota mechanisms that ensure that users and subjects will always have at least a minimum of a specified resource and that they will not be able to monopolise a controlled resource.

Management : for FRU_RSA.1

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

- a) specifying maximum limits for a resource for groups and/or individual users and/or subjects by an administrator;

Management : for FRU_RSA.2

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

- a) specifying minimum and maximum limits for a resource for groups and/or individual users and/or subjects by an administrator;

Audit : for FRU_RSA.1 and FRU_RSA.2

487 The following actions shall be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Minimal: Rejection of allocation operation due to resource limits.
- b) Basic: All attempted uses of the resource allocation functions for resources that are under control of the TSF.

D R A F T

FRU_RSA.1 Maximum Quotas

Hierarchical to: no other components.

FRU_RSA.1.1 The TSF shall enforce quotas limiting the maximum quantity of [assignment: *controlled resources*] that [selection: *individual user, defined group of users, subjects*] can use [selection: *simultaneously, over a specified period of time*].

Dependencies : No dependencies.

FRU_RSA.2 Minimum and Maximum Quotas

Hierarchical to: FRU_RSA.1

FRU_RSA.2.1 The TSF shall enforce quotas limiting the maximum quantity of [assignment: *controlled resources*] that [selection: *individual user, defined group of users*] can use [selection: *simultaneously, over a specified period of time*].

FRU_RSA.2.2 The TSF shall ensure the provision of minimum quantity of each [assignment: *controlled resource*] that is available for [selection: *an individual user, defined group of users, subjects*] to use [selection: *simultaneously, over a specified period of time*]

Dependencies : No dependencies.

D R A F T

DRAFT

Class FTA

TOE Access

488 This family specifies functional requirements for controlling the establishment of a
user’s session.

489 Figure 2.16 shows the decomposition of this class into its constituent components.

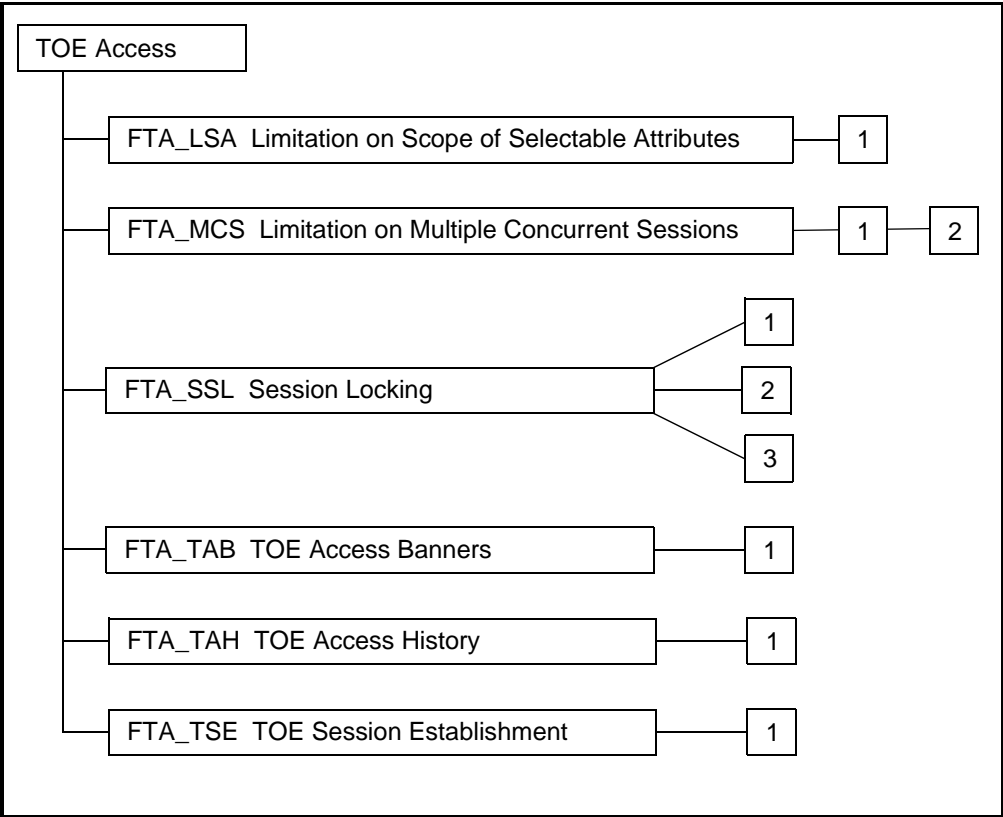


Figure 2.16 - TOE Access class decomposition

D R A F T

FTA_LSA Limitation on Scope of Selectable Attributes

Family behaviour

490 This family defines requirements to limit the scope of attributes that a user may select for a session.

Component levelling

FTA_LSA Limitation on Scope of Selectable Attributes

1

491 There is only one component in this family.

Management :

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

- a) management of the scope of the session security attributes by an administrator.

Audit :

493 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Minimal: All failed attempts at selecting a user attribute.
- b) Basic: All attempts at selecting a user attribute.
- c) Detailed: Capture of the values of each user security attribute.

FTA_LSA.1 Limitation on Scope of Selectable Attributes

Hierarchical to: no other components.

FTA_LSA.1.1 The TSF shall restrict the scope of the session security attributes [assignment: *session security attributes*], based on [assignment: *attributes*].

Dependencies : No dependencies.

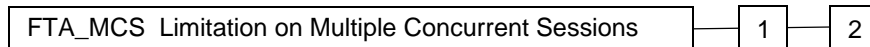
D R A F T

FTA_MCS Limitation on Multiple Concurrent Sessions

Family behaviour

494 This family defines requirements to place limits on the number of concurrent sessions that belong to the same user.

Component levelling



495 FTA_MCS.1 Basic Limitation on Multiple Concurrent Sessions provides limitations that apply to all users of the TSF.

496 FTA_MCS.2 Per User Attribute Limitation on Multiple Concurrent Sessions extends FTA_MCS.1 by requiring the ability to specify limitations on the number of concurrent sessions based on the related security attributes.

Management : FTA_MCS.1

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

- a) management of the maximum allowed number of concurrent user sessions by an administrator.

Management : FTA_MCS.2

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

- a) management of the rules that govern the maximum allowed number of concurrent user sessions by an administrator.

Audit :

499 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST

- a) Minimal: Rejection of a new session based on the limitation of multiple concurrent sessions.
- b) Detailed: Capture of the number of currently concurrent user sessions and the user security attribute(s).

D R A F T

FTA_MCS.1 Basic Limitation on Multiple Concurrent Sessions

Hierarchical to: no other components.

FTA_MCS.1.1 The TSF shall restrict the maximum number of concurrent sessions that belong to the same user.

FTA_MCS.1.2 The TSF shall enforce, by default, a limit of a single session per user.

Dependencies : FIA_UID.1 Timing of Identification

FTA_MCS.2 Per User Attribute Limitation on Multiple Concurrent Sessions

Hierarchical to: FTA_MCS.1

FTA_MCS.2.1 The TSF shall restrict the maximum number of concurrent sessions that belong to the same user **based on [assignment: *security attributes*] according to the rules [assignment: *rules for the number of maximum concurrent sessions*].**

FTA_MCS.2.2 The TSF shall enforce, by default, a limit of a single session per user.

Dependencies : FIA_UID.1 Timing of Identification

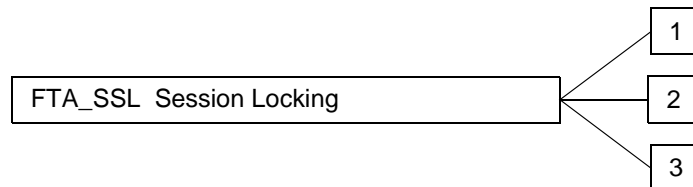
D R A F T

FTA_SSL Session Locking

Family behaviour

500 This family defines requirements for the TSF to provide the capability for TSF-initiated and user-initiated locking and unlocking of interactive sessions.

Component levelling



501 FTA_SSL.1 TSF-initiated Session Locking includes system initiated locking of an interactive session after a specified period of user inactivity.

502 FTA_SSL.2 User-initiated Locking provides capabilities for the user to lock and unlock the user's own interactive sessions.

503 FTA_SSL.3 TSF-initiated Termination provides requirements for the TSF to terminate the session after a period of user inactivity.

Management : FTA_SSL.1

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

- a) specification of the time of user inactivity after which lock-out occurs for an individual user;
- b) specification of the default time of user inactivity after which lock-out occurs.

Audit :

505 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Minimal: Locking of an interactive session by the session locking mechanism.
- b) Minimal: Termination of the interactive session by the TSF.
- c) Minimal: Successful unlocking of an interactive session.
- d) Basic: Any attempts at unlocking an interactive session.

D R A F T

FTA_SSL.1 TSF-initiated Session Locking

Hierarchical to: no other components.

FTA_SSL.1.1 The TSF shall lock an interactive session after [assignment: *time interval*] interval of user inactivity by:

- a) clearing or overwriting display devices, making the current contents unreadable;
- b) disabling any activity of the user's data access/display devices other than unlocking the session.

FTA_SSL.1.2 The TSF shall require [assignment: *events to occur*] prior to unlocking the session.

Dependencies : **FIA_UAU.1** Timing of authentication

FTA_SSL.2 User-initiated Locking

Hierarchical to: no other components.

FTA_SSL.2.1 The TSF shall allow user-initiated locking of the user's own interactive session, by:

- a) clearing or overwriting display devices, making the current contents unreadable;
- b) disabling any activity of the user's data access/display devices other than unlocking the session.

FTA_SSL.2.2 The TSF shall require [assignment: *events to occur*] prior to unlocking the session.

Dependencies : **FIA_UAU.1** Timing of authentication

FTA_SSL.3 TSF-initiated Termination

Hierarchical to: no other components.

FTA_SSL.3.1 The TSF shall terminate an interactive session after a [assignment: *time interval*] interval of user inactivity.

Dependencies : No dependencies.

D R A F T

FTA_TAB TOE Access Banners

Family behaviour

506 This family defines requirements to display a configurable advisory warning message to users regarding the appropriate use of the TOE.

Component levelling

FTA_TAB TOE Access Banners

1

507 FTA_TAB.1 Default TOE Access Banners provides the requirement for a TOE Access Banner. This banner is displayed prior to the establishment dialogue for a session.

Management :

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

a) maintenance of the banner by the authorised administrator.

Audit :

509 There are no actions identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

FTA_TAB.1 Default TOE Access Banners

Hierarchical to: no other components.

FTA_TAB.1.1 Before establishing a user session, the TSF shall display an advisory warning message regarding unauthorised use of the TOE.

Dependencies : No dependencies.

D R A F T

FTA_TAH TOE Access History

Family behaviour

- 510 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.

Component levelling

FTA_TAH TOE Access History

1

- 511 There is only one component in this family.

Management :

- 512 There are no management activities foreseen.

Audit :

- 513 There are no actions identified that should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST.

FTA_TAH.1 TOE Access History

Hierarchical to: no other components.

- FTA_TAH.1.1 Upon successful session establishment, the TSF shall display the [selection: *date, time, method, location*] of the last successful session establishment to the user.**

- FTA_TAH.1.2 Upon successful session establishment, the TSF shall display the [selection: *date, time, method, location*] of the last unsuccessful attempt to session establishment and the number of unsuccessful attempts since the last successful session establishment.**

- FTA_TAH.1.3 The TSF shall not erase the access history information from the user interface until it receives a direct user request to do so.**

Dependencies : No dependencies.

D R A F T

FTA_TSE TOE Session Establishment

Family behaviour

514 This family defines requirements to deny a user permission to establish a session with the TOE.

Component levelling

FTA_TSE TOE Session Establishment

1

515 There is only one component in this family.

Management : FTA_TAH.1

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

- a) management of the session establishment conditions by the authorised administrator.

Audit : FTA_TAH.1

517 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- a) Minimal: Successful use of the session establishment mechanism.
- b) Basic: All attempts at establishment of a user session.
- c) Detailed: Capture of the value of the selected access parameters (e.g. location of access, time of access).

FTA_TSE.1 TOE Session Establishment

Hierarchical to: no other components.

FTA_TSE.1.1 The TSF shall be able to deny session establishment based on [assignment: *attributes*].

Dependencies : No dependencies.

D R A F T

Class FTP

Trusted Path/Channels

517 Families in this class provide requirements for a trusted communication path between users and the TSF, and for a trusted communication channel between the TSF and other trusted IT products. Trusted paths and channels have the following general characteristics:

- The communications path is constructed using internal and external communications channels (as appropriate for the component) that isolate an identified subset of TSF data and commands from the remainder of the TSF and user data.
- Use of the communications path may 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)

517 In this paradigm, a ***trusted channel*** is a communication channel that may be initiated by either side of the channel, and provides non-repudiation characteristics with respect to the identity of the sides of the channel.

518 A ***trusted path*** provides a means for users to perform functions through an assured direct interaction with the TSF. Trusted path is usually desired for user actions such as initial identification and/or authentication, but may also be desired at other times during a user's session. Trusted path exchanges may 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.

519 Figure 2.17 shows the decomposition of this class into its constituent components.

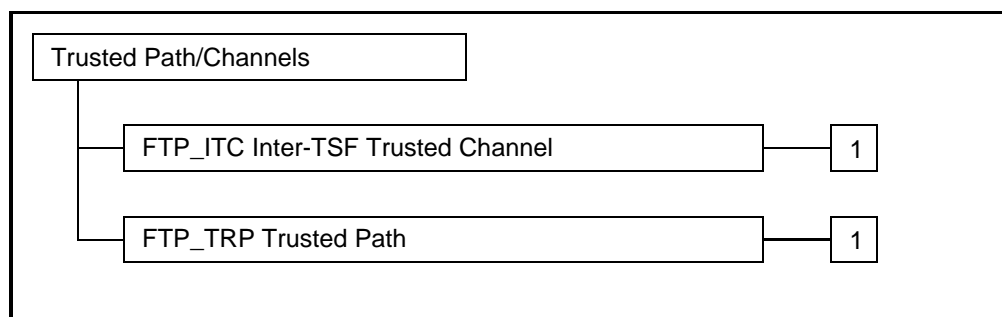


Figure 2.17 - Trusted Path / Channels Class decomposition

D R A F T

FTP_ITC Inter-TSF Trusted Channel

Family behaviour

520 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. This family should be included whenever there are requirements for the secure communication of user or TSF data between the TOE and other trusted IT products.

Component levelling

FTP_ITC Inter-TSF Trusted Channel

1

521 FTP_ITC.1 Inter-TSF Trusted Channel requires that the TSF provide a trusted communication channel between itself and another trusted IT product.

Management :

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

- a) Configuring the actions that require trusted channel, if supported.

Audit :

523 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Failure of the trusted channel functions.
- b) Minimal: Identification of the initiator and target of failed trusted channel functions.
- c) Basic: All attempted uses of the trusted channel functions.
- d) Basic: Identification of the initiator and target of all trusted channel functions.

FTP_ITC.1 Inter-TSF Trusted Channel

Hierarchical to: no other components.

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote 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.

D R A F T

FTP_ITC.1.2 **The TSF shall permit [selection: *only the TSF, the remote trusted IT product*] to initiate communication via the trusted channel.**

FTP_ITC.1.3 **The TSF shall initiate communication via the trusted channel for [assignment: *list of functions for which a trusted channel is required*].**

Dependencies : No dependencies.

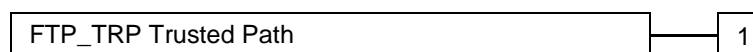
D R A F T

FTP_TRP Trusted Path

Family behaviour

524 This component 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.

Component levelling



525 FTP_TRP.1 Trusted Path requires that a trusted path between the TSF and a user be provided for a set of events defined by a PP/ST author. The user and/or the TSF may have the ability to initiate the trusted path.

Management :

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

- a) Configuring the actions that require trusted path, if supported.

Audit :

527 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

- a) Minimal: Failures of the trusted path functions.
- b) Minimal: Identification of the user associated with all trusted path failures, if available.
- c) Basic: All attempted uses of the trusted path functions.
- d) Basic: Identification of the user associated with all trusted path invocations, if available.

FTP_TRP.1 Trusted Path

Hierarchical to: no other components.

FTP_TRP.1.1 The TSF shall provide a communication path between itself and [selection: *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 modification or disclosure.

D R A F T

FTP_TRP.1.2 The TSF shall permit [selection: *only the TSF, local users, remote users*] to initiate communication via the trusted path.

FTP_TRP.1.3 The TSF shall require the use of the trusted path for [selection: *initial user authentication*, [assignment: *other services for which trusted path is required*]].

Dependencies : No dependencies.

D R A F T