THALES

XOMAIL VERSION 14.2

SECURITY TARGET

Edition: **10-public**

28-Apr-09

Author: **CHTE**

Resp.: **ØYJ**

Appr.: **BEKO**

All pages in this document shall have the same edition number

© THALES Norway AS

Foreword

This document forms the basis for conducting a security evaluation of the XOmail Military Messaging System, in accordance with the Common Criteria for Information Technology Security Evaluation. The document describes the operating environment and IT product requirements, as well as security functionality implemented in the IT product.

The document was prepared on behalf of

Forsvarets Logistikkorganisasjon IKT,
Postmottak, NO-2617 Lillehammer, Norway

By:

THALES Norway AS
PO Box 6611 Etterstad, NO-0609 Oslo, Norway
Risløkkv. 2, NO-0580 Oslo

Telephone: (+47) 22 63 83 00
Telefax:: (+47) 22 63 79 44
E-Mail: firmapost@no.thalesgroup.com
Internet: http://www.thalesgroup.no

Copyright notice:

This work is licensed under the
Creative Commons Attribution-NoDerivs-NonCommercial License
(which allows redistribution of the work). To view a copy of this license, visit
http://creativecommons.org/licenses/by-nd-nc/3.0/ or send a letter to Creative
Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

THALES Norway AS has made every attempt to ensure that the information in this document is correct and complete. However, THALES Norway AS assumes no liability for errors, or for any damage that may result from the use of this document or any product that it accompanies.

Comments on this document may be sent to THALES Norway AS by either postal mail or email. Please include document name, document edition and a reference within the document to which the comment apply.

TABLE OF CONTENTS

1. SCOPE OF DOCUMENT 9
2. RELATED DOCUMENTS 10
3. TERMINOLOGY 11
   3.1 Abbreviations and acronyms 11
   3.2 Document organisation 13
4. GENERAL 14
   4.1 ST introduction 14
      4.1.1 ST identification 14
      4.1.2 Supported Operating Systems 14
      4.1.3 ST overview 14
      4.1.4 CC conformance claim 15
   4.2 TOE description 15
      4.2.1 TOE identification 15
      4.2.2 Main characteristics 15
      4.2.3 System overview 18
      4.2.4 XOmail overview 20
         4.2.4.1 Administrator access control 21
         4.2.4.2 API framework 21
         4.2.4.3 MIP Gateway 22
         4.2.4.4 Maritime Gateway 23
         4.2.4.5 DMP gateway 24
      4.2.5 OS responsibilities 24
         4.2.5.1 Authentication mechanism 24
   4.3 TOE security environment 24
      4.3.1 Assumptions 24
      4.3.2 Assets 25
      4.3.3 Threat agents 26
      4.3.4 Threats 26
         4.3.4.1 Threats met by the TOE 26
         4.3.4.2 Threats met by the TOE environment 29
      4.3.5 Organisational security policy 31
   4.4 Security objectives 34
      4.4.1 Security objectives for the TOE 34
      4.4.2 IT Security objectives for the TOE environment 36
      4.4.3 Non-IT Security objectives for the TOE environment 37
5. IT SECURITY REQUIREMENTS 38
   5.1 TOE security requirements 38
      5.1.1 TOE security functional requirements 38
         5.1.1.1 Class FAU: Security audit 38
         5.1.1.2 Class FDP: User data protection 42
         5.1.1.3 Class FIA: Identification and authentication 45
         5.1.1.4 Class FMT: Security management 47
         5.1.1.5 Class FPT: Protection of the TSF 48
         5.1.1.6 Class FTA: TOE access 50
      5.1.2 TOE security assurance requirements 50
         5.1.2.1 Class ACM: Configuration management 51
         5.1.2.2 Class ADO: Delivery and operation 52
         5.1.2.3 Class ADV: Development 53
         5.1.2.4 Class AGD: Guidance document 56
         5.1.2.5 Class ALC: Life cycle support 58
         5.1.2.6 Class ATE: Tests 59
         5.1.2.7 Class AVA: Vulnerability assessment 61

© THALES Norway AS

5.2 Security requirements for the IT environment 62
5.2.1 IT environment security functional requirements 62
5.2.1.1 Class FAU: Security audit 62
5.2.1.2 Class FIA: Identification and authentication 64
5.2.1.3 Class FDP: User Data Protection 65
5.2.1.4 Class FPT: Protection of the TSF 65
5.2.1.5 Class FTA: TOE Access 65
6. TOE SUMMARY SPECIFICATION 66
6.1 TOE security functions 66
6.1.1 SF.AUDIT 66
6.1.2 SF.AUTHENTICATION 67
6.1.3 SF.AUTO_LOGOUT 67
6.1.4 SF.CLASSIFICATION_TAG 67
6.1.5 SF.CLEAR 68
6.1.6 SF.COMMAND_ACCESS 68
6.1.7 SF.COMMUNICATION_SECURITY 68
6.1.8 SF.DAC 69
6.1.9 SF.DB_SELF_TEST 69
6.1.10 SF.EXECUTION_DOMAINS 69
6.1.11 SF.LABEL_TRANSFORM 69
6.1.12 SF.LABELLING 70
6.1.13 SF.LOCK 70
6.1.14 SF.MAC 70
6.1.15 SF.ROLES 70
6.1.16 SF.SECURE_STATE_RECOVERY 71
6.1.17 SF.SUBNET_RESTRICTION 71
6.1.18 SF.VALIDATE 71
6.2 Assurance measures 72
7. PP CLAIMS 73
8. RATIONALE 74
8.1 Security objectives rationale 74
8.1.1 TT.ADM_ERROR 76
8.1.2 TT.AUDIT_FAILURE 77
8.1.3 TT.COM_INTEGRITY 77
8.1.4 TT.DOS 77
8.1.5 TT.INSERTION 78
8.1.6 TT.MASQUERADE 78
8.1.7 TT.MONITORING 79
8.1.8 TT.REPLAY 79
8.1.9 TT.UNATTENDED 80
8.1.10 TT.UNAUTH_ACCESS 80
8.1.11 TE.AUDIT_FAILURE 82
8.1.12 TE.DELIVERY 82
8.1.13 TE.DOS 83
8.1.14 TE.IMPROPER_INST 83
8.1.15 TE.POOR_DESIGN 83
8.1.16 TE.POOR_IMPL 84
8.1.17 TE.UNATTENDED 84
8.1.18 A.ADM_TRAINING 85
8.1.19 A.AUDIT_REVIEW 85
8.1.20 A.CONFIDENCE 85
8.1.21 A.INVALIDATE 85
8.1.22 A.NETWORK 85
8.1.23 A.NOTIFY 86
8.1.24 A.PHYSICAL 86

© THALES Norway AS

8.1.25 A.PHYSICAL_LOC 86
8.1.26 A.OS 86
8.1.27 A.USR_TRAINING 87
8.1.28 P.ACCOUNTING 87
8.1.29 P.CLASSIFICATION 87
8.1.30 P.CLEAR 87
8.1.31 P.DAC 87
8.1.32 P.INTEGRITY 87
8.1.33 P.INTERFACE_CONTROL 88
8.1.34 P.MAC 88
8.1.35 P.MARKING 88
8.1.36 P.PROTECTION 88
8.2 Security requirements rationale 88
8.2.1 Requirements are appropriate 88
8.2.1.1 O.ACCESS_HIST 90
8.2.1.2 O.AUDIT 90
8.2.1.3 O.AUTO_LOGOUT 91
8.2.1.4 O.CMD_ACL 91
8.2.1.5 O.CMD_LOG 92
8.2.1.6 O.DAC 92
8.2.1.7 O.ID_AUTH 93
8.2.1.8 O.LABELLING 93
8.2.1.9 O.LOCK 93
8.2.1.10 O.MAC 94
8.2.1.11 O.MAC_INTEGRITY 94
8.2.1.12 O.MANAGE 94
8.2.1.13 O.MESSAGING 95
8.2.1.14 O.RECOVER 95
8.2.1.15 O.REF_MONITOR 96
8.2.1.16 O.RESOURCE_SHARE 96
8.2.1.17 O.REUSE 96
8.2.1.18 O.ROLE_MNG 96
8.2.1.19 O.ROLES 96
8.2.1.20 O.SELF_TEST 97
8.2.1.21 OE.ACCOUNTABLE 97
8.2.1.22 OE.AUDIT 97
8.2.1.23 OE.ID_AUTH 98
8.2.1.24 OE.NETWORK 98
8.2.1.25 OE.TRAF_SEPARATION 98
8.2.1.26 NOE.ADM_TRUST 98
8.2.1.27 NOE.INSTALL 99
8.2.1.28 NOE.PHYSICAL 99
8.2.2 Functional security requirements dependencies 99
8.2.3 Security assurance requirements dependencies 100
8.3 TOE summary specification rationale 101
8.3.1 TOE security functional requirements satisfaction 101
8.3.1.1 FAU_ARP.1 103
8.3.1.2 FAU_GEN.1(1) 103
8.3.1.3 FAU_GEN.2(1) 103
8.3.1.4 FAU_SAA.1 103
8.3.1.5 FAU_SAR.1(1) 103
8.3.1.6 FAU_SAR.2(1) 104
8.3.1.7 FAU_STG.1(1) 104
8.3.1.8 FAU_STG.3(1) 104
8.3.1.9 FAU_STG.4(1) 104
8.3.1.10 FDP_ACC.1 104

© THALES Norway AS

8.3.1.11 FDP_ACF.1 105
8.3.1.12 FDP_ETC.2 105
8.3.1.13 FDP_IFC.2 105
8.3.1.14 FDP_IFF.2 105
8.3.1.15 FDP_ITC.2 106
8.3.1.16 FDP_RIP.2 106
8.3.1.17 FIA_AFL.1 106
8.3.1.18 FIA_ATD.1 106
8.3.1.19 FIA_UAU.2 106
8.3.1.20 FIA_UAU.5 107
8.3.1.21 FIA_UID.2(1) 107
8.3.1.22 FIA_USB.1 107
8.3.1.23 FMT_MSA.1 107
8.3.1.24 FMT_MSA.3 108
8.3.1.25 FMT_MTD.1 108
8.3.1.26 FMT_SMF.1 108
8.3.1.27 FMT_SMR.1 108
8.3.1.28 FPT_FLS.1 108
8.3.1.29 FPT_RCV.1 108
8.3.1.30 FPT_RCV.2 109
8.3.1.31 FPT_RCV.4 109
8.3.1.32 FPT_RVM.1 109
8.3.1.33 FPT_SEP.3 109
8.3.1.34 FPT_TDC.1 109
8.3.1.35 FPT_TST.1 110
8.3.1.36 FTA_SSL.3 110
8.3.1.37 FTA_TSE.1(1) 110
8.4 PP rationale 110

© THALES Norway AS

LIST OF TABLES

Table 2-1: Related documents 10
Table 3-1: Abbreviations and Acronyms 12
Table 3-2: Terminology....13
Table 3-3: Document to CC mapping....13
Table 4-1: TOE Supported operating systems....14
Table 4-2: Supported operating systems outside the TOE 14
Table 5-1: IT Security Requirements notation....38
Table 5-2: Auditable Events 41
Table 5-3: EAL4....51
Table 5-4: Auditable Events 64
Table 6-1: Assurance measures 72
Table 8-1: TOE threats coverage 75
Table 8-2: Assumptions coverage....75
Table 8-3: Policies coverage 76
Table 8-4: Security objectives satisfaction 89
Table 8-5: Environment security objectives satisfaction 90
Table 8-6: Functional requirements dependency check 100
Table 8-7: Assurance requirements dependency check....101
Table 8-8: Functional requirements satisfaction....102

© THALES Norway AS

LIST OF FIGURES

Figure 4-1: XOmail overview 18
Figure 4-2: MMHS servers in a Partitioned Operation Mode system 18
Figure 4-3: MMHS servers in an MLS environment 19
Figure 4-4: XOmail API framework 22
Figure 4-5: XOmail Maritime Gateway 22

© THALES Norway AS

1. SCOPE OF DOCUMENT

This document is a Security Target for XOmail. The Security Target forms the basis for product security evaluation according to the Common Criteria (CC). The document contains a list of threats met by XOmail and its environment, security objectives for XOmail, security function requirements, and assurance requirements for XOmail. Rationale is provided for each of the identified security objectives, requirements and functions identified. Assurance level for the XOmail equals the EAL4 defined in CC.

© THALES Norway AS

2. RELATED DOCUMENTS

[1]	712 27734 AXAA EO	XOmail Installation and Configuration Guide
[2]	739 20529 ABAA EO	XOmail User's Guide
[3]	739 20561 ABAA EO	XOmail Administrator's Guide
[4]	ESD-TR-75-306	Bell & La Padula: Secure Computer Systems: *Unified Exposition and Multics Interpretation*.
[5]	FOR 2001-07-01 nr 744	Forskrift om informasjonssikkerhet. (Norwegian directives on information security).
[6]	CCMB-2005-08-001	Common Criteria for Information Technology Security Evaluation, August 2005, Version 2.3, Part 1 (also known as part 1 of the ISO/IEC 15408 Evaluation Criteria).
[7]	CCMB-2005-08-002	Common Criteria for Information Technology Security Evaluation, August 2005, Version 2.3, Part 2 (also known as part 2 of the ISO/IEC 15408 Evaluation Criteria).
[8]	CCMB-2005-08-003	Common Criteria for Information Technology Security Evaluation, August 2005, Version 2.3, Part 3 (also known as part 3 of the ISO/IEC 15408 Evaluation Criteria).
[9]	C-M(2002)49	Security Within the North Atlantic Treaty Organisation (NATO), 17. June 2002.
[10]	LOV 1998-03-20 nr 10	Lov om forebyggende sikkerhetstjeneste (Norwegian Security Act)
[11]	STANAG 4406 ed. 1	Military Message Handling System (NATO C3 Board), March 1999
[12]	RFC 2156	MIME Internet X.400 Enhanced Relay
[13]	Open Group (X/Open)	API to Electronic Mail (X.400), Issue 3, May 1996
[14]	Open Group (X/Open)	OSI-Abstract-Data Manipulation API (XOM), Issue 3, May 1996
[15]	Open Group (X/Open)	Message Store API (XMS), June 1993
[16]	Open Group (X/Open)	API to Directory Services (XDS), Issue 3, May 1996

Table 2-1: Related documents

© THALES Norway AS

3. TERMINOLOGY

3.1 Abbreviations and acronyms

ACL	Access Control List
ACP	Allied Communication Publication
ASN.1	Abstract Syntax Notation number One
B&L	Bell & La Padula Security model
CAPP	Controlled Access Protection Profile
CC	Common Criteria Version 2.3 (ISO/IEC 15408 Evaluation Criteria)[6,7, 8]
CCIS	Command Control Information System
CM	Configuration Management
CMS	Cryptographic Message Syntax
CORBA	Common Object Request Broker Architecture
COTS	Commercial Off The Shelf
CUI	Character-based User Interface
C4ISR	Command, Control, Communications, Computers, Intelligence, Surveillance and Reconnaissance
DAC	Discretionary Access Control
DAP	Directory Access Protocol
DOS	Denial Of Service
DMP	Direct Message Profile
DSP	Directory System Protocol
EAL	Evaluation Assurance Level
FSM	Finite State Machine
FSMI	Finite State Machine Interpreter
HCL	Hierarchical Classification Level (e.g. RESTRICTED)
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
JAPI	Journal API (XOmail interface)
LAN	Local Area Network
MAC	Mandatory Access Control
MAPI	Messaging API
MCI	MIP Common Interface (NATO standard)
MEM	Message Exchange Mechanism (The message exchange part of MCI)
MHS	Message Handling System
MIP	Multilateral Interoperability Programme
ML	Multi Level
MMHS	Military Message Handling System
MMHS server	A server (hardware and OS) running the XOmail Server software
MTA	Message Transfer Agent
NATO	North Atlantic Treaty Organization
NHC	Non-Hierarchical Category (e.g. CLEAR)
OS	Operating System
PCT	Protecting Content Type
PP	Protection Profile
SA	Security Administrator
SFP	Security Function Policy
SIC	Subject Indicator Code
SL	Single Level
SOF	Strength Of Function
SP	Security Policy
SS	Secure Storage

© THALES Norway AS

ST	Security Target
STANAG	NATO Standardization Agreement
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Functions
TSP	TOE Security Policy
WAN	Wide Area Network
XOmail	The XOmail MMHS software product, including Server, MailClient, TSClient and AdminClient
XOmail Server	The XOmail Server software

Table 3-1: Abbreviations and Acronyms

Admin Main Object	The main configurable objects of the system. These objects are visible in the top level of the administration client's navigation tree for a given Message Server.
Admin Object	Configurable objects of the system. These objects are identified with leaves in the static parts of the administration client's navigation tree.
Administrator	The least privileged administrator role. Can administer all parts of the TOE, except for the Network Management parts, security parameters and Security Administrator restricted User Templates. Administrator access can be further limited using Command Access parameters in the User Template.
Clearance	Each user is assigned a clearance indicating the maximum security classification of information the user is allowed to access.
Command Access	ACLs for administrative commands. Access to each command can be controlled on a per User Template basis.
Multi Level object	Object that is able to handle multiple pieces of information with different security labels.
Network Administrator	Administrator with extended privileges compared to the Administrator role. This role has access to administration of Network Management parameters. Other than this, the same limitations are valid as for Administrator.
Non-resident processes	Software processes that run temporarily during TOE operation.
Normal	The ordinary user role. Users belonging to this role have no administrative access.
OS Root	The OS user named "root". For a Solaris install, this coincides with the OS super user, while for other platforms this is a constructed user as described in XOmail Installation and Configuration Guide[1].
Primary Security Administrator	The most privileged administrator role. Can administer all parts of the TOE. Command access limitations will not be applied.
Resident processes	Software processes that are started during TOE start-up and remains running until TOE shutdown.
Security Administrator	The most privileged administrator role. Can administer all parts of the TOE. Command access limitations can however be applied.
Security Attribute	CC definition: Characteristics of subjects, users, objects, information, and/or resources that are used for the enforcement of the TSP. XOmail context: Subject and object HCLs, NHCs and SPs.
Single Level object	Object that is able to handle information at a single security label equal to its own security label.
Social engineering	The use of persuasion and/or deception to gain access to information systems.
Target of Evaluation	An IT product or system and its associated guidance documentation that is the subject of an evaluation.

© THALES Norway AS

Template	Upon creation all System Units are based on a template. The new System Unit remains associated with a template during its whole lifetime, and some attributes will remain pure template attributes. Templates must be created for Users, Departments, Message Servers and Directory Servers.
Trusted object	Object that is allowed to override security policies.
Trusted subject	Subject that is allowed to override security policies.

Table 3-2: Terminology

3.2 Document organisation

The organisation of this document is similar to the organisation proposed in CC part 1[6]. The main components are ordered as specified in the CC standard[6], while the chapter numbering differs. The mapping from chapter numbers in this document to ST components identified by CC can be found in Table 3-3.

Chapter reference within this document	ST component as identified in CC part 1 [6]
4.1	B.2.2
4.2	B.2.3
4.3	B.2.4
4.4	B.2.5
5	B.2.6
6	B.2.7
7	B.2.8
8	B.2.10

Table 3-3: Document to CC mapping

The reason for this document ordering is conformance with existing requirement specifications for the product. Chapter 4 contains background material and cursory descriptions of the requirements. The actual requirements are contained in chapter 5. Other required ST components are put in separate subsequent chapters.

© THALES Norway AS

4. GENERAL

4.1 ST introduction

4.1.1 ST identification

ST title: XOmail version 14.2 Security Target

ST version: See front page.

ST date: See front page.

TOE name: XOmail

TOE version: XOmail 14.2.4, product id 712 27734 AXAA 73

4.1.2 Supported Operating Systems

The following operating systems are supported by the TOE:

Application	Operating systems
XOmail Server	Windows Server 2003 SP2, including R2, Standard, Enterprise, and Datacenter

Table 4-1: TOE Supported operating systems

The Windows versions listed above have been CC-certified EAL4 with CAPP. Refer to the publicly available certification reports for additional information. Previous versions of Windows XP and Server 2003 have also been certified. These are outdated and should not be used.

XOmail also runs on the operating systems shown in Table 4-2. Support for these operating systems is not part of the TOE.

Application	Operating systems
XOmail Server	Solaris 8, 9, and 10 (Standard x86-based PC and Sun UltraSPARC) Windows 2000 with SP4 Professional, Server, and Advanced Server

Table 4-2: Supported operating systems outside the TOE

Solaris versions 8, 9 and 10 are available CC-certified EAL4 with CAPP. Refer to the publicly available certification reports for additional information.

4.1.3 ST overview

This ST describes the IT security requirements for XOmail, a software military message handling system (MMHS) built according to ITU-T X.400 with military extensions according to STANAG 4406.

© THALES Norway AS

4.1.4 CC conformance claim

The XOmail Security Target has been developed using the Common Criteria (CC) Version 2.3 (ISO/IEC 15408 Evaluation Criteria for Information Technology Security; Part 1: Introduction and general model, Part 2: Security functional requirements, and Part 3: Security assurance requirements).

XOmail has been developed to include components as defined in Common Criteria Part 2. The XOmail development conforms to the EAL4 assurance level as identified in Common Criteria Part 3.

4.2 TOE description

4.2.1 TOE identification

The TOE is the XOmail MMHS server component, with the functionality as defined in Section 4.2.2. XOmail is a COTS product tailored to information handling and transfer in modern military C4ISR solutions and large organizations.

The TOE consists of the XOmail Server, which is the building block for the messaging infrastructure. The TOE provides a number of APIs, which allow third-party applications to use the messaging infrastructure.

The TOE is accessed by user and administration clients, which provides user interfaces for messaging and day-to-day management. The clients are not part of the TOE.

In this document, an *MMHS server* is a computer running the XOmail Server. The MMHS Server may additionally run any of the XOmail clients or API applications, but this is usually not the case. The operating system and the computer hardware/firmware is not a part of the TOE.

A client computer is a computer running the XOmail MailClient, the XOmail TSClient and/or the XOmail AdminClient. The clients, the operating system and the computer hardware/firmware is not a part of the TOE.

4.2.2 Main characteristics

Main characteristics of the TOE:

- Military messaging system built according to ITU-T X.400 with STANAG 4406 Ed. 1 military extensions.
- Integrated Multi-Level Security.
- Supports ACP 127 to allow automatic flow to/from traditional teleprinter networks. The implementation is compliant with “ACP 127 NATO SUPP-3 (A)” and “STANAG-4406 Annex D”.
- Organisation-to-organisation, as well as person-to-person, messaging.
- Includes a limited ACP133 Directory Service and the ability to interact with an external master Directory Service.

© THALES Norway AS

- Maintainable, portable and extendable software for dedicated Messaging services. Customer extensible through industry standard Application Program Interfaces.
- Interface for an administration client capable of local and remote administration and supervision.
- Message distribution mechanism that is able to distribute messages to both local and global addressees. This is in addition to standard auto forward functionality.
- SIC handling.
- Messages handled according to message priorities, including mapping to transport layer priorities.
- X.25 protocol support
  The X.25 protocol is used to interface X.25-based networks or radio equipment.
- MIP Gateway
  The MIP gateway interfaces networks based on the NATO MIP Message Exchange Mechanism. See also Section 4.2.4.3.
- Maritime Gateway
  Support for broadcast and ship-shore services using ACP 127 procedures and formats. See also Section 4.2.4.4.
- DMP gateway
  DMP is an extremely low overhead protocol for use in tactical communications. The protocol is defined in STANAG 4406 Annex E. See also Section 4.2.4.5.
- Perl extensions
  The Perl extensions support automated installation. This module is only used for installation, not operational use.

In addition to the TOE, XOmail provides the clients listed below. The clients are used to access the TOE.

- XOmail Mail Client
  End-user messaging client
- XOmail Admin Client
  Administrative interface
- XOmail Transit Storage Client
  Traffic operator interface

The functions below are also provided by XOmail, but shall not be used with the TOE in a certified configuration. The functions are contained in packages which are not installed on the XOmail Server or XOmail MailClient unless explicitly selected during installation. The XOmail Outlook Add-In is installed separately.

- Security Services (Server)
  The Security Services module provides support for the Spanish legacy SICOMEDE MMHS security services and PCT signatures (STANAG 4406 legacy server-server signatures).

© THALES Norway AS

- P_mul gateway (Server)
  XOmail provides experimental support for the P_mul protocol defined in STANAG 4406 Annex E. The P_mul protocol is designed for use in tactical environments with limited bandwidth.

- Internet Protocols (Server)
  The two following modules provide support for client access using internet protocols. The current version provides a limited implementation of the POP3 and IMAP4 standards.

  - POP3 client access
    This module is not part of the TOE.
    The POP3 module is provided for testing only.

  - IMAP4 client access
    This module is not part of the TOE.

    The IMAP4 server allows Microsoft Outlook to be used to access XOmail messages. The XOmail Outlook Add-In must be used. This product is not part of the TOE.

- Central Archive (Server)
  The Central Archive module provides the XOmail Server with the ability to support organizational archive policies. Messages are stored in one or several centralized external archives using third-party software. XOmail provides an interface to the archives, allowing storage, retrieval and search functions.

- Client Security Services
  The XOmail MailClient is able to use information stored on smart cards for logon and digital message signatures.

  XOmail relies on third-party software for signing and hashing of messages, while verification of signed messages is handled by XOmail.

- XOmail Outlook Add-In
  The XOmail Outlook Add-In is a plug-in to Microsoft Outlook. The plug-in allows Microsoft Outlook to be used to access messages in XOmail. The add-in requires the IMAP4 module above.

Figure 4-1 provides an overview of the XOmail MMHS.

© THALES Norway AS

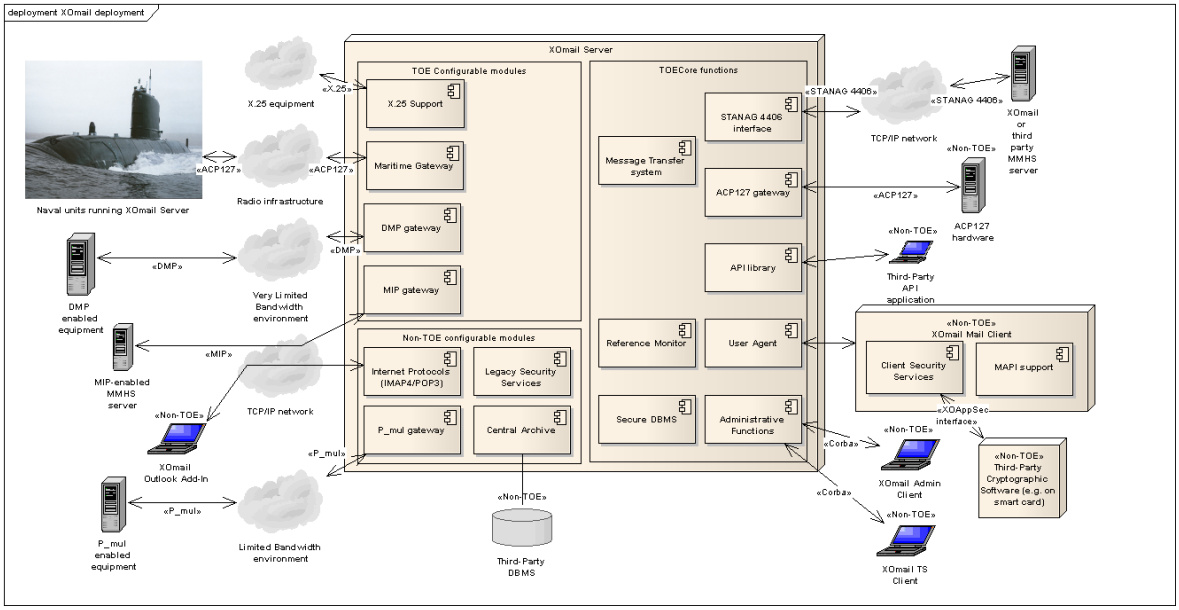

Figure 4-1: XOmail overview

4.2.3 System overview

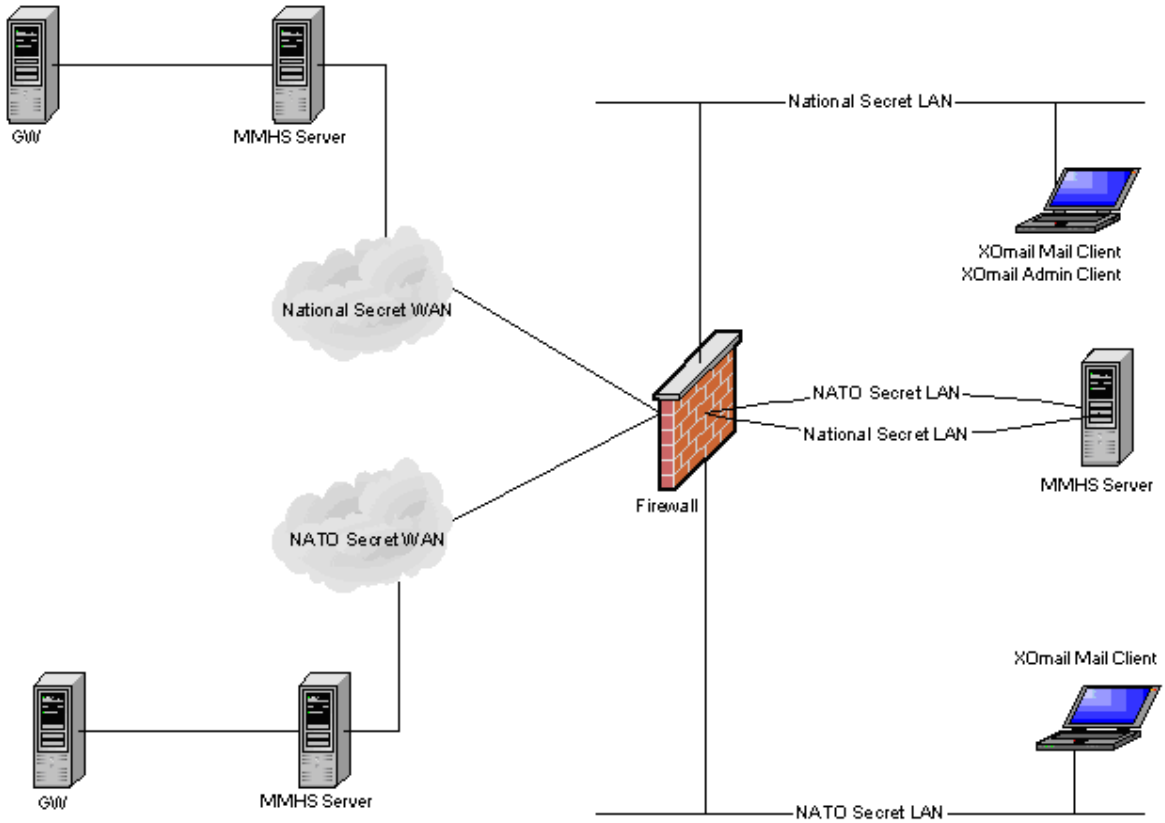

Figure 4-2: MMHS servers in a Partitioned Operation Mode system

© THALES Norway AS

Figure 4-2 shows a general overview of the Partitioned Operation-Mode.

The MMHS server running XOmail Server may in principle also run XOmail MailClient, the XOmail TSClient and/or the XOmail AdminClient. This will not normally be the case.

The XOmail Server may serve client logons from both the National Secret LAN and the NATO Secret LAN. In addition it is able to communicate with the MMHS Servers located in the WANs. XOmail Server may also exchange messages with third party MMHS servers that are present. XOmail Server is able to separate information classified in the different security policies.

In the Figure 4-2 scenario, XOmail Mail Clients have been allowed to connect from both the National Secret LAN and the NATO Secret LAN. Some users may connect only from the National Secret LAN, while others may connect from both. Administrators are only allowed to connect from the National Secret LAN.

Figure 4-3 shows a generalized version of Figure 4-2, with MMHS servers operating in a multi level security environment. This scenario shows the XOmail Server in two different roles, as security gateway (the server marked MLS), and as individual MMHS servers. The security gateway ensures that only messages labelled *restricted* or lower may pass between the two networks. Messages marked *secret,* or higher, are blocked

- when sent from the *secret* network, and
- when sent from the *restricted* network.

The first rule preserves confidentiality, while the second rule ensures that *secret* messages cannot be injected into the *secret* LAN from the less trusted LAN. These rules are configurable.

Messages may be marked with additional labels, such as *national eyes only*. A *national restricted* LAN may exchange messages with a *NATO restricted* LAN, using the XOmail Server to ensure that *national eyes only* messages do not propagate outside the national network.

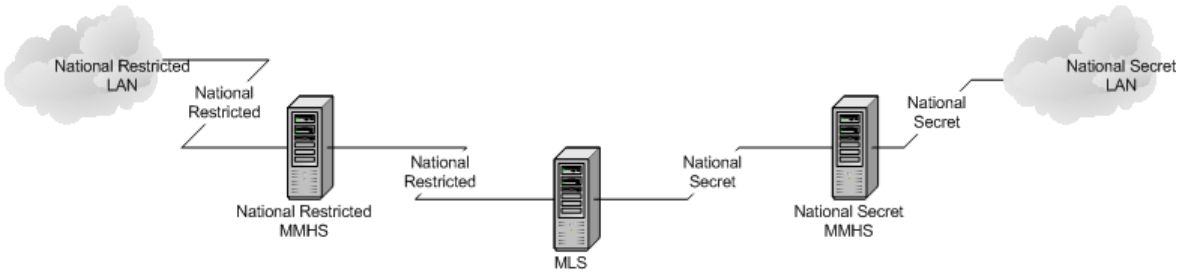

Figure 4-3: MMHS servers in an MLS environment

The protection of the LANs to which the MMHS servers and clients are connected is out of scope of this Security Target. It is assumed that the confidentiality of information is protected using approved IP-crypto, or any other approved means of protection.

© THALES Norway AS

4.2.4 XOmail overview

Logons to be handled by the XOmail Server are Administration Client logons, Transit Storage Client logons and Mail Client logons. It is possible to use configurations so that logons can be performed from a specific host only, from a specific subnet, or from any host. Logon restrictions may be applied to administrative client logons only, messaging client logons only, or both.

The MailClient is considered a *dumb client* that:

a) Is available for all personnel that have user accounts (given that the account has not been locked).

b) Presents only the data that the server allows it to present. The server ensures this by not sending data that the client is not allowed to present (ensures both MAC and DAC). The clients also present the label that is associated with the information.

c) Sends requests to the server in order to store, retrieve, delete or create data. The server verifies that the operation is allowed for that user in his/her current environment.

The MailClient supports labelling of information upon presentation, but the label is always retrieved from the Server, or from the user that creates the data. To ensure information confidentiality, all traffic between the Server and the clients should be protected by some means that ensures confidentiality (e.g. using IP-crypto).

The TSClient is a *thin client* that:

- Is available for all personnel that have appropriate administrative rights (given that the user account has not been locked). See 4.2.4.1 for details on how administrative rights are assigned.
- Presents only the information that the server allows it to present. The server ensures this by not sending data that the client is not allowed to present (ensures both MAC and DAC).
- Presents both unclassified and classified information. The server always controls the classification as it associates a label with all information that is sent to the TSClient.
- Supports “command access” and “administrator roles” by hiding unavailable commands.
- Sends requests to the server in order to retrieve, delete or create data. The server verifies that the operation is allowed for that user in his/her current environment.

To ensure information confidentiality, all traffic between the Server and the clients should be protected by some means to ensure confidentiality (e.g. using IP-crypto). The traffic to and from the TSClient is not considered to be administrative traffic, and does not need to be separated from other message traffic.

The AdminClient is a *thin client* that:

- Is available for all personnel that have appropriate administrative rights (given that the user account has not been locked). See 4.2.4.1 for details on how administrative rights are assigned.
- Presents only the XOmail configuration data that the server allows it to present. The server ensures this by not sending data that the client is not allowed to present.
- Does not present message data.
- Supports “command access” and “administrator roles” by hiding unavailable commands.

© THALES Norway AS

- Sends requests to the server in order to store, retrieve, delete or create data. The server verifies that the operation is allowed for that user in his/her current environment.

To ensure information confidentiality, all traffic between the Server and the clients should be protected by some mean that ensures confidentiality (e.g. using IP-crypto).

The Terminal Interface:

XOmail Server provides the legacy text-based administration tool TI, which is used for a few specialized tasks. TI is integrated with the XOmail Server and requires console access to the computer the XOmail Server runs on.

4.2.4.1 Administrator access control

There are four administrator roles in XOmail: Administrator role, Network Administrator role, Security Administrator and Primary Security Administrator role. All administrators, regardless of role, have access to the AdminClient and the TSClient.

An administrator's User Template defines a set of commands available to all administrators based on that template. This command access grants and denies access to commands available in every command group, e.g. it is possible to grant access to Open in the Department Template command group, while denying access to New, Save and Delete.

The maximum command access that is possible to configure for any given User Template is determined from the administrator role that is set for the User Template. E.g. it will only be possible to grant access to network management tasks to Network Administrators and Security Administrators.

Only Security Administrators and the Primary Security Administrator are allowed to store changes in security parameters, such as user clearance.

A user with access to TI has access to all commands in TI. Thus, a user with access to TI is equivalent to a Security Administrator. To use TI, a user must be a member of the OS group for XOmail privileged users. The group is managed by the operating system. DAC is enforced by the operating system.

MAC is enforced on TI users by the XOmail Server in the same way as all other XOmail users. This means that a user must be added as a XOmail user to gain access to classified information. OS users with access to TI, but not added as XOmail users, will not be able to access classified information, e.g. logs.

4.2.4.2 API framework

XOmail provides a number of standard APIs defined by X/Open for accessing the messaging infrastructure [13-16], e.g. API to Electronic Mail (MA-API) and X/Open Message Store API (MS-API).

Additionally, XOmail provides the XOmail Simplified API (XOsapi), based on MA-API. XOsapi offers a simplified, easy-to-use interface to the send and the receive functionality of the XOmail Server.

On computers not running XOmail Server, the XOmail RemoteAccess is installed to provide a low level communication stack which handles communication towards a remote XOmail Server.

© THALES Norway AS

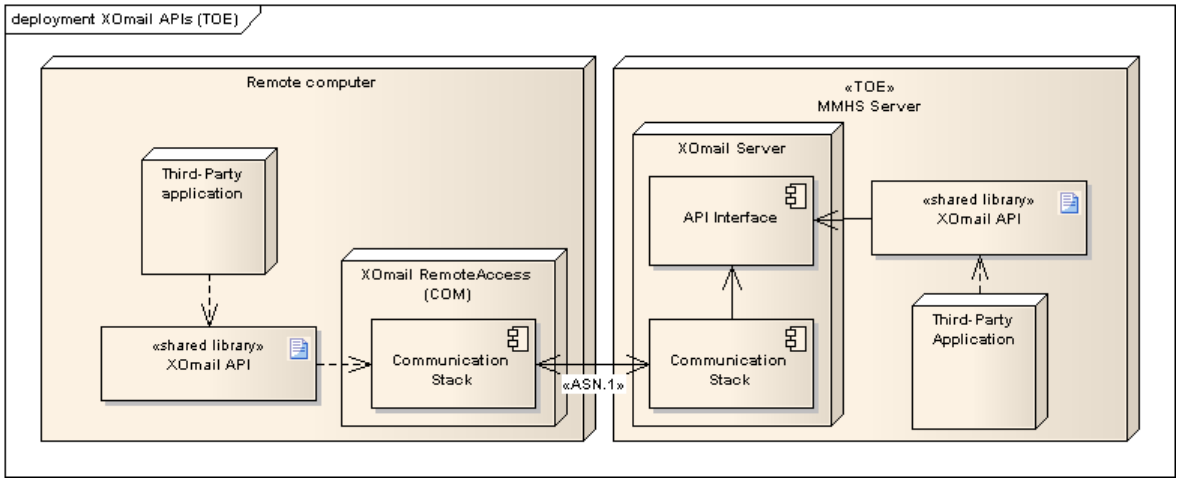

Figure 4-4: XOmail API framework

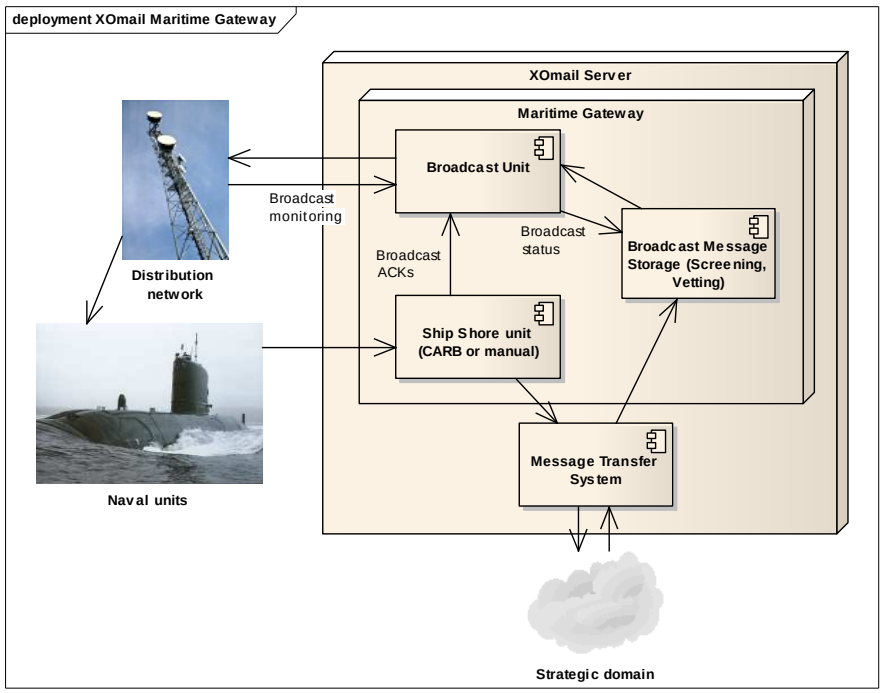

Figure 4-5: XOmail Maritime Gateway

4.2.4.3 MIP Gateway

The MIP gateway allows messages to be sent and received through the SMTP based NATO MIP Message Exchange Mechanism. MIP messages are received through an SMTP server which accepts messages on the Internet Mail format with MIP extensions. The messages are converted to the STANAG 4406 format according to the MIXER guidelines [12].

The MIP gateway transmits messages to other MIP servers using SMTP.

The module can be selected during installation of the XOmail Server. If the server is not going to communicate with remote servers using the MIP protocol, this module does not need to be installed.

© THALES Norway AS

4.2.4.4 Maritime Gateway

The Maritime Gateway relays messages between domains connected via maritime communications and procedures. The Maritime Gateway module provides additional system unit types for ACP127 communication. The module uses the base ACP127 channel support for low-level communication. An overview of the Maritime Gateway is shown in Figure 4-5 above.

A number of Broadcast Units and Ship-Shore Units can be defined on an XOmail Maritime Gateway.

The Broadcast Unit handles the transmission (broadcasting) of messages to ships. The Ship-Shore Unit handles incoming messages from ships. Thus, two-way connectivity requires at least one Broadcast and one Ship-Shore Unit to be defined.

Broadcast Unit:

The system offers two filtering functions for reducing broadcast load:

- **Screening**
  All messages are subject to automatic screening. Expired messages are discarded and suspected duplicates are halted.

- **Vetting**
  Messages may be stopped for manual review to determine the relevance in the current situation.

Other special Broadcast Unit functions are:

- **Broadcast Schedules**
  Ships may maintain reduced radio watch. A broadcast schedule can be established to ensure that messages are only sent during watch hours. Broadcast schedules can also be used for time-sharing a broadcaster, by allowing certain types of traffic (e.g. NATO only) during certain periods of the day.

- **Automatic Re-runs**
  To ensure that all called stations will receive a message, transmission re-runs can be defined.

- **Traffic Lists generation**
  Broadcast Units log all broadcasted messages in Traffic Lists. The ship side uses these Traffic Lists to determine whether all broadcast messages are received or not.

Ship-Shore Unit:

A Ship-Shore Unit handles incoming messages from ships. In addition, a Ship-Shore Unit provides functionality for automatic and manual aerial switching as well as reception quality assessment and control. A number of Ship-Shore Units can be defined on one XOmail Maritime Gateway.

The ship uses Working Channels for sending messages to the Maritime Gateway. A channel link must be established before a message can be sent. Channel establishment is either performed automatically using CARB procedures, or manually by an operator. In addition, an Aerial Select Channel can be used for directing/controlling the antenna used for receiving messages from ships.

The Maritime Gateway module can be selected during installation of the XOmail Server. The Maritime Gateway does not need to be installed unless the broadcast/ship-shore functionality is required. The base ACP 127 functionality is not affected by this module.

© THALES Norway AS

4.2.4.5 DMP gateway

The DMP gateway allows messages to be sent and received through the DMP protocol defined in STANAG 4406 Annex E. DMP is a low overhead messaging protocol for use in low bandwidth scenarios.

The module is selected during installation of the XOmail Server. If the server is not going to communicate with remote servers using the DMP protocol, this module does not need to be installed.

4.2.5 OS responsibilities

4.2.5.1 Authentication mechanism

The TOE utilizes the authentication mechanisms provided by the OS. The responsibility of the TOE is to ensure that authentication is performed before any other operation. The OS is responsible for performing the actual authentication and provide secure storage of the authentication tokens.

4.3 TOE security environment

In the following sub-chapters the TOE security environment will be described. First, a list of assumptions regarding the TOE use and operating environment is given. It is important to note that the operating environment is responsible for parts of the TOE security functionality. This is further described in the list of assumptions. Following the list of assumptions is a list of assets and threat agents. These lists are necessary for later identifying the threats. The assets in the system are objects for which the TOE shall ensure confidentiality, integrity and availability, whereas threat agents are the subjects that may perform actions later identified as threats.

Based on the operating environment, the list of threat agents and the TOE itself (including assets), a list of threats can be identified. The identified threats are listed in 4.3.4. All threats are listed with a description of the threat, the involved threat agents, the affected assets and a description of unwanted outcome from a possible attack. In 4.3.5, the organisational security policy with which the TOE must comply is described.

4.3.1 Assumptions

A.ADM_TRAINING	All administrators know how to administrate the TOE in a secure manner. Administration of the TOE in a secure manner means that each administrator must know all consequences of all administrative tasks that are performed. Furthermore, each administrator knows all his/her responsibilities with regards to TOE administration. Administrator training shall be based on XOmail Administrator's Guide[3].
A.AUDIT_REVIEW	Administrator personnel review audit logs on a regular basis.
A.CONFIDENCE	Administrators or developers will not intentionally compromise the TOE security.
A.INVALIDATE	Proper disposal of authentication data and associated privileges is performed after access is removed (job termination, change in responsibility). Proper

© THALES Norway AS

	disposal means that the authentication data cannot be used to authenticate towards any part of the TOE.
A.NETWORK	The network connections used between separate parts of the TOE and for external communication are protected from unauthorized disclosure and modification.
A.NOTIFY	Administrators and users notify the proper authority of any security issues that impact their systems. This will minimize the potential for loss or compromise of data.
A.PHYSICAL	The hardware on which XOmail runs is protected from unauthorized physical modification.
A.PHYSICAL_LOC	The hardware on which XOmail runs is located where only authorized personnel have access.
A.OS	The TOE runs on a CAPP evaluated OS with EAL4 or higher. The OS protects the TOE from unauthorized modifications and provide vital security mechanisms like auditing.
A.USR_TRAINING	All users know how to use the TOE in a secure manner. Use of the TOE in a secure manner means that each user must know the consequences of all tasks that are performed. It is also assumed that the user training ensures that the user always labels information correctly. User training shall be based on XOmail User's Guide[2].

4.3.2 Assets

AS.AUDIT	The log data gathered by the TOE and OS that are part of the audit trail.
AS.AUTH_DATA	Data provided by a subject to perform authentication.
AS.CLASSIFIED_INFO	Information classified according to a specific security policy. The information is assigned a hierarchical classification level (HCL) and optionally a set of non-hierarchical categories (NHC).
AS.CONFIG_EXT	TOE configuration data that is stored within the control of the OS. This includes configuration files for which the OS controls the access.
AS.CONFIG_SS	TOE configuration data that is stored in the SS, except that covered by AS.SEC_DATA.
AS.PROPER_OP	Proper operation of the TOE is considered an asset as it is important for authorized users to get access to the system as needed.
AS.SEC_DATA	TOE security data. This includes - clearance settings for users and departments, - clearance settings and other security settings for unit descriptors for units that the TOE may communicate with, - labelling of structures within SS, - ACLs for structures within SS,

© THALES Norway AS

4.3.3 Threat agents

TA.ADM	Authenticated authorized administrators of XOmail. These threat agents may unintentionally perform unauthorized actions. Administrators that have not authenticated, or perform actions from applications other than the AdminClient or the TSClient are considered TA.INTERNAL.
TA.DEVELOPER	The XOmail developer. These threat agents may unintentionally compromise the TOE security.
TA.EXTERNAL	Personnel with no authorized access to the TOE environment. These threat agents may try to access classified information and may have “unlimited” resources supporting them.
TA.INTERNAL	Personnel with authorized access to the TOE environment. These threat agents may try to perform unauthorized actions. Furthermore, such threat agents may be specially trained to perform the unauthorized actions and may have “unlimited” resources supporting them.
TA.SYSTEM_ERROR	Hardware or software failures or transmission errors may cause information to be modified by accident.
TA.USER	Authenticated authorized users of the TOE. These threat agents may intentionally or unintentionally perform unauthorized actions. Users that have not authenticated, or perform actions from applications other than the MailClient are considered TA.INTERNAL.

4.3.4 Threats

The threats are divided into two groups based on who meets them. The first group, named TT, is comprised by the threats met by the TOE itself. The second group, named TE, is comprised by the threats met by the TOE environment.

4.3.4.1 Threats met by the TOE

TT.ADM_ERROR	Improper administration may result in override of specific security policy.
Threat agents	TA.ADM.
Asset	AS.CLASSIFIED_INFO, AS.CONFIG_SS, AS.SEC_DATA.
Unwanted outcome	Unauthorized personnel get access to, change, or delete classified information because of unintentional improper administration of the TOE.

© THALES Norway AS

TT.AUDIT_FAILURE	An attacker may cause audit records to be lost or modified. Attackers may also cause audit overflow, so that important audit records seemingly disappear.
Threat agents	TA.INTERNAL, TA.EXTERNAL
Asset	AS.AUDIT
Unwanted outcome	The TOE is unable to store audit data or provide necessary audit data to the IT environment, or the audit becomes useless because of the inability to separate important audit records from other records. The latter is the case if the audit is overflowed.

TT.COM_INTEGRITY	The integrity of transmitted information may be compromised due to deliberate or accidental modification.
Threat agents	TA.INTERNAL, TA.EXTERNAL, TA.SYSTEM_ERROR
Asset	AS.CLASSIFIED_INFO
Unwanted outcome	Classified information is modified.

TT.DOS	An attacker prevents authorized users from accessing system resources via a resource exhaustion denial of service attack.
Threat agents	TA.USER, TA.INTERNAL, TA.EXTERNAL
Asset	AS.PROPER_OP
Unwanted outcome	Authorized users do not get access to necessary information that they have clearance for.

TT.INSERTION	An attacker introduces information that appears to come from a trusted entity.
Threat agents	TA.INTERNAL, TA.EXTERNAL
Asset	AS.CLASSIFIED_INFO
Unwanted outcome	Information is inserted with a fraudulent originator or classification.

© THALES Norway AS

TT.MASQUERADE	An attacker tries to masquerade as a trusted entity in order to by mistake be trusted with classified information.
Threat agents	TA.INTERNAL, TA.EXTERNAL
Asset	AS.CLASSIFIED_INFO, AS.CONFIG_SS, AS.SEC_DATA
Unwanted outcome	Classified information is sent/made available to an entity that is not authorized for the data.

TT.MONITORING	An attacker monitors activities and actions performed on classified information. Such activities and actions include authentication and creating, viewing, modifying and deleting classified information. The monitoring activities can be performed at multiple levels, like screen monitoring or network monitoring.
Threat agents	TA.INTERNAL, TA.EXTERNAL
Asset	AS.CLASSIFIED_INFO, AS.AUTH_DATA
Unwanted outcome	Based on the monitoring activities, attackers can make assumptions of the type of information sent, and possibly even the content of the information sent. Statistical methods can be used to make such assumptions. Also determining normal traffic load, and possible divergence from this can give attackers valuable information. Possibly, the attacker can draw conclusions that would be considered classified information based on traffical patterns.

TT.REPLAY	A malicious process or user gains access by replaying authentication data.
Threat agents	TA.INTERNAL, TA.EXTERNAL
Asset	AS.CLASSIFIED_INFO, AS.CONFIG_SS, AS.SEC_DATA
Unwanted outcome	Unauthorized personnel get access to classified information by replaying the authentication data provided by an authorized user or administrator.

TT.UNATTENDED	A malicious user may gain unauthorized access to an unattended session.
Threat agents	TA.ADM, TA.USER and TA.INTERNAL
Asset	AS.CLASSIFIED_INFO, AS.CONFIG_SS, AS.SEC_DATA
Unwanted outcome	Unauthorized personnel get access to the specified assets.

© THALES Norway AS

TT.UNAUTH_ACCESS Unauthorized access to identified assets may occur. Methods of attack covered by this threat are brute force attacks, session hijacking, authentication data cracking, privilege escalation and social engineering.

Threat agents TA.ADM, TA.USER, TA.INTERNAL and TA.EXTERNAL

Asset AS.CLASSIFIED_INFO, AS.CONFIG_SS, AS.SEC_DATA

Unwanted outcome Unauthorized personnel get access to classified information.

4.3.4.2 Threats met by the TOE environment

TE.AUDIT_FAILURE An attacker may cause audit records to be lost or modified.

Threat agents TA.INTERNAL, TA.EXTERNAL

Asset AS.AUDIT

Unwanted outcome Audit records are prevented from being recorded, or an attacker is able to alter the information before it is placed in the audit trail.

TE.DELIVERY An attacker may try to replace parts (or the complete) TOE with a malicious version.

Threat agents TA.INTERNAL, TA.EXTERNAL.

Asset AS.AUDIT, AS.CLASSIFIED_INFO, AS.CONFIG_EXT, AS.CONFIG_SS, AS.PROPER_OP, AS.SEC_DATA

Unwanted outcome Unauthorized personnel get unauthorized access to classified information because a compromised version of the TOE is used to maintain the assets.

TE.DOS An attacker block authorized users from system resources via a resource exhaustion denial of service attack.

Threat agents TA.ADM, TA.USER, TA.INTERNAL, TA.EXTERNAL.

Asset AS.PROPER_OP.

Unwanted outcome Authorized users do not get access to necessary information that they have clearance for.

© THALES Norway AS

TE.IMPROPER_INST	The TOE is installed and/or configured in a manner that undermines security.
Threat agents	TA.ADM, TA.USER, TA.INTERNAL, TA.EXTERNAL.
Asset	AS.CLASSIFIED_INFO, AS.CONFIG_EXT, AS.CONFIG_SS
Unwanted outcome	The TOE is installed in a manner that eases the process of gaining unauthorized access to classified information.

TE.POOR_DESIGN	Unintentional or intentional errors in design of the XOmail may occur. Such design flaws includes: inability to adequately separate information based on SP, HCL or NHC and inability to associate correct security attributes with the users.
Threat agents	TA.DEVELOPER
Asset	AS.AUDIT, AS.AUTH_DATA, AS.CONFIG_EXT, AS.CONFIG_SS, AS.CLASSIFIED_INFO, AS.SEC_DATA.
Unwanted outcome	The developer has failed in designing the TOE in a secure manner thereby undermining its ability to protect assets against attacks.

TE.POOR_IMP	The developer has failed in implementing the TOE in a secure manner, failed in implementing the TOE according to the design, or deliberately planted backdoors, Trojans or similar.
Threat agents	TA.DEVELOPER
Asset	AS.AUDIT, AS.AUTH_DATA, AS.CONFIG_EXT, AS.CONFIG_SS, AS.CLASSIFIED_INFO, AS.SEC_DATA.
Unwanted outcome	Attackers get access to classified information, audit data or configuration data. Attackers may also attempt to successfully conceal unauthorized access and other attacks.

TE.UNATTENDED	A malicious user may gain unauthorized access to an unattended session.
Threat agents	TA.ADM, TA.USER and TA.INTERNAL
Asset	AS.AUDIT, AS.CLASSIFIED_INFO, AS.CONFIG_EXT.
Unwanted outcome	Unauthorized personnel get access to the specified assets.

© THALES Norway AS

4.3.5 Organisational security policy

The TOE is compliant with the applicable parts of:

Norwegian security policy	Norwegian Security Act [10] with supplementary Norwegian Information Security Regulations [5].
NATO security policy	C-M(2002)49 Security Within the North Atlantic Treaty Organisation (NATO) [9]

The following is a summary of security policy statements from the Norwegian security policy and NATO security policy which are related to the TOE and/or TOE environment.

P.ACCOUNTING	The objective of the policy for accounting is to provide sufficient information to be able to investigate a deliberate or accidental compromise of accountable information and assess the damage arising from the compromise.
Norwegian security policy	Norwegian Information Security Regulations [5]: §4-11, §4-12, §4-14, §4-16.
NATO security policy	NATO security policy [9]: • Enclosure B: §16 • Enclosure E: §§14-16, §§17-23 • Enclosure F: §11, §12.i

P.CLASSIFICATION	The objective of the policy for classification of information is to determine who is responsible for security classification of information, which security classifications to be used, handling of re-classification, and time-limitations of classifications.
Norwegian security policy	Norwegian Security Act [10]: §11 Norwegian Information Security Regulations [5]: §2-1, §2-3, §2-9-§2-13,
NATO security policy	NATO security policy [9]: • Enclosure B: §§16-19 • Enclosure E: §3, §5

© THALES Norway AS

P.CLEAR		Under exceptional operational circumstances, classified information may be transmitted in clear text provided each occasion is properly authorised.
	NATO security policy	NATO security policy [9]: • Enclosure F: §21
P.DAC		A discretionary access control policy based on identity and need-to-know of the user, process and/or groups to which they belong, shall be enforced.
	Norwegian security policy	Norwegian Information Security Regulations [5]: §3-3, §5-2
	NATO security policy	NATO security policy [9]: • Enclosure B: §9.b, §11, §16, §20 • Enclosure C • Enclosure F : §12.a, §12.b
P.INTEGRITY		Classified information shall be protected against alteration and introduction of false information.
	Norwegian security policy	Norwegian Information Security Regulations [5]: §5-5
	NATO security policy	NATO security policy [9]: • Enclosure B: §2 • Enclosure F: §§3-4, §12.c, §12.d

© THALES Norway AS

P.INTERFACE_CONTROL Different information systems can be connected on the following conditions:

1. It shall only be used services and protocols, and administration of these, which are necessary to fulfil the functional requirements of the system.
2. Each of the connected information systems shall have a protection against other information systems, and the security in each of the systems shall be based on mechanisms in that system.
3. Security measures shall be implemented on different levels in the system, to avoid that the protection is based on only one component.
4. Access according to need-to-know shall be implemented mutually between the connected systems.

Norwegian security policy: Norwegian Information Security Regulations [5]: §5.4

NATO security policy: NATO security policy [9]:

- Enclosure F: §12.f

P.MAC A mandatory access control policy based on hierarchical classification levels and non-hierarchical categories shall be enforced. Information shall not be allowed to flow from a higher security level to a lower security level or between non-comparable security levels.

All individuals, civilian and military, who require access to, or whose duties or functions may afford access to information classified CONFIDENTIAL / KONFIDENSIELT or above, shall be appropriately cleared and briefed before such access is authorised.

Norwegian security policy: Norwegian Security Act [10]: §§19-26

Norwegian Information Security Regulations [5]: §3-3, §5-2,

NATO security policy: NATO security policy [9]:

- Enclosure A: Article 3
- Enclosure B : §9.b, §11, §12
- Enclosure C.
- Enclosure F: §7

© THALES Norway AS

P.MARKING The objective of the policy for marking of classified information is to determine who is responsible for the marking, and details on how the marking shall be done.

Norwegian security policy: Norwegian Security Act [10]: §§11-16

Norwegian Information Security Regulations [5]: §2-2 - §2-7, §4-1 - §4-3, §4-7, §4-8

NATO security policy: NATO security policy [9]:

- Enclosure B: §19
- Enclosure E: §3, §7, §§10-12, §13

P.PROTECTION Classified information, stored or transmitted, shall be protected against loss of confidentiality, integrity or availability. A balanced set of security measures (physical, personnel, security of information and INFOSEC) shall be implemented.

Protective measures and procedures to prevent, detect, and recover from the loss or compromise of information shall be enforced.

Norwegian security policy: Norwegian Information Security Regulations [5]: §5-5, §5-6, §6-1

NATO security policy: NATO security policy [9]:

- Enclosure B: §2, §16, §22
- Enclosure D.
- Enclosure E: §§1-2, §5
- Enclosure F : §§3-4, §8, §12.f, §12.h, §16-20

4.4 Security objectives

4.4.1 Security objectives for the TOE

O.ACCESS_HIST The TOE maintains information related to previous attempts for a user to establish a session. This information is displayable to authorized administrators.

O.AUDIT	The TOE uses its internal secure database (SS), as well as OS audit mechanisms for recording any security related information. When OS audit mechanisms are used, the TOE provides the OS with all necessary information, including the identity of the user that caused the event. The audit shall contain information that is sufficient to reconstruct sequences of events, i.e. the event shall include the event time, the event type, who was responsible for the event, and the outcome of the event.
O.AUTO_LOGOUT	The TOE provides an automatic logout mechanism for the MailClient and the CUI. The mechanism terminates client sessions after a configurable period of inactivity.
O.CMD_ACL	The TOE provides means for restricting access to administrative commands for each user or group of users. This can be used to restrict the administrative right given to the role that the users belong to.
O.CMD_LOG	The TOE provides means for recording administrative commands.
O.DAC	The TOE ensures Discretionary Access Control by controlling access to resources based on the identity of users and groups of users.
O.ID_AUTH	A user is identified and authenticated before given access to classified information. Authentication mechanisms include verification of the provided username and password.
O.LABELLING	The TOE ensures that information is labelled with the correct human-readable label when exported out of TSC.
O.LOCK	The TOE provides a locking mechanism that makes it possible to prevent users from logging on, even if they have a valid account. The locking mechanism works on accounts, and can be activated both manually and automatically. Automatic locking is activated when the user has provided a number of invalid authentication tokens.
O.MANAGE	The TOE allows administrators to effectively, accurately and securely manage the TOE and its security functions.
O.MAC	The TOE ensures Mandatory Access Control by controlling access to resources based on security clearance of users and resources.
O.MAC_INTEGRITY	The TOE allows authorized security administrators to specify the security clearance of users and resources.
O.MESSAGING	The TOE provides secure messaging functions. This implies that messages with incorrect security marks will be rejected, while correctly formatted messages are accepted. The TOE will furthermore be able to convert security marks to and from all supported security mark representations.
O.RECOVER	The TOE will ensure preservation of a secure state in the event of a secure component failure. Upon restart after an abnormal termination, the state may not be a secure state, and the TOE shall use the current state to recover to a secure state.

© THALES Norway AS

O.REF_MONITOR	A reference monitor ensures that a SFP implemented by a TSF cannot be by-passed.
O.RESOURCE_SHARE	The TOE ensures that no entity can block other authorized entities from accessing resources.
O.REUSE	The TOE ensures secure reuse of resources. Secure reuse implies that it is not possible to retrieve information stored during a previous use of the resource by other subjects or by the same subject at a different security label.
O.ROLE_MNG	When template “Permit” flag is set, only administrators with the same, or a more privileged level can associate users with that template. However, for templates where the Permit flag is not set, only *Security Administrators* can associate other users with that template, thereby prohibiting use of that template for non-“Security Administrators”.
O.ROLES	The TOE assigns each user to a specific role. The TOE will define roles named *Normal*, *Administrator, Network Administrator, Security Administrator* and *Primary Security Adminstrator*. The role *Normal* has no administrative rights, while the role *Administrator* and *Network Administrator* have administrative rights except security settings. The roles *Security Administrator* and *Primary Security Adminstrator* has all administrative rights. For all roles it is possible to limit the privileges, except for the user assigned *Primary Security Adminstrator* role which will always have all privileges. Furthermore, all defined users are associated with one of the defined roles above. All roles can be blocked for log-in except for the *Primary Security Administrator*.
O.SELF_TEST	The TOE database performs a self-test during start-up. The system will not be operable before the database consistency check passes.

4.4.2 IT Security objectives for the TOE environment

OE.ACCOUNTABLE	Those responsible for the TOE will ensure that the product is configured such that only the group of users for which the system was accredited may access the system, and furthermore that each individual user is assigned a unique user identification.
OE.AUDIT	The OS will perform auditing as specified in CAPP. The audit shall contain information that is sufficient to reconstruct sequences of events, i.e. the event shall include the event time, the event type, who was responsible for the event, and the outcome of the event.
OE.ID_AUTH	A user is identified and authenticated before given access to the OS that the TOE runs on.
OE.NETWORK	Those responsible for the TOE will ensure that networks that are used for communication between separate parts of the TOE and for external communication are protected. The protection is implemented by means of data encryption or physical protection.
OE.TRAF_SEPARATION	The TOE environment will ensure that TOE administrative network traffic can be separated from other TOE network traffic.

© THALES Norway AS

4.4.3 Non-IT Security objectives for the TOE environment

NOE.ADM_TRUST	Those responsible for the TOE will ensure that administrators of the system are trustworthy.
NOE.INSTALL	Those responsible for the TOE will ensure that the TOE is installed, managed and operated according to the TOE guidance documentation. This requires users and administrators to be properly trained.
NOE.PHYSICAL	Those responsible for the TOE will ensure that security relevant components of the TOE are protected from physical attack that might compromise the IT-security.

© THALES Norway AS

5. IT SECURITY REQUIREMENTS

The IT security requirements are standard CC requirements, with minor adaptations. Any deviations from the CC standard requirements have been marked with *blue and italic text*.

The IT security requirements are fulfilled by either the TOE or the IT environment. Some security requirements need to be covered both by the TOE or the IT environment, or are represented in both domains. These requirements may be iterated. Some requirements may need to be iterated to cover different variations on the same subject. The notations for these are shown in Table 5-1.

FAU_GEN.1	Standard component, not iterated.
FAU_GEN.1.1	Element 1 of standard component FAU_GEN.1, not iterated.
FAU_GEN.1(1)	First iteration of standard component FAU_GEN.1.
FAU_GEN.1(1).1	Element 1 of iterated component FAU_GEN.1(1).
FAU_GEN.1(2)	Second iteration of standard component FAU_GEN.1.

Table 5-1: IT Security Requirements notation

5.1 TOE security requirements

5.1.1 TOE security functional requirements

The following subchapters present the security functional requirements for the TOE. Among the assurance requirements, AVA_SOF.1 is included. Therefore a SOF claim must be made for security functions realised using probabilistic or permutational mechanisms. The SOF claim made for TOE security functions is SOF-Medium. Further details each of the security functions can be found in 6.1.

5.1.1.1 Class FAU: Security audit

FAU_ARP.1
Security alarms.

FAU_ARP.1.1
The TSF shall take *the following actions* upon detection of a potential security violation:

a) Warm start on selected events

b) Cold start on selected events

FAU_GEN.1(1)
Audit data generation.

FAU_GEN.1(1).1
The TOE shall be able to generate audit records of the following auditable events:

a) Start-up and shutdown of the audit functions;

© THALES Norway AS

b) All auditable events for the *basic* level of audit*, as listed in* Table 5-2.

c) Message operations on the offline journal are not logged.

FAU_GEN.1(1).2
The TOE shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event;

b) For each audit event type, based on the auditable event definitions of the functional components included in the ST, *message identifier for selected audit event types.*

FAU_GEN.2(1)
User identity association

FAU_GEN.2(1).1
The TOE shall be able to associate each auditable event with the identity of the user that caused the event.

TOE Requirement	Event	Note
FAU_ARP.1	Actions taken due to imminent security violations	
FAU_GEN.1(1)	N/A	
FAU_GEN.2(1)	N/A	
FAU_SAA.1	An alarm is raised when the number of alarm events for a given alarm type occurs more than once within the given period.	
FAU_SAR.1(1)	Reading of information from the audit records.	
FAU_SAR.2(1)	Unsuccessful attempts to read information from the records.	
FAU_STG.1(1)	N/A	
FAU_STG.3(1)	An alarm is raised when the available disk space falls below a given limit.	
FAU_STG.4(1)	N/A	When the disk is full, it is not possible to store auditable events.
FDP_ACC.1	N/A	
FDP_ACF.1	All requests to perform an operation on an object covered by the SFP.	
FDP_ETC.2	All attempts to export information.	

© THALES Norway AS

TOE Requirement	Event	Note
FDP_IFC.2	N/A	
FDP_IFF.2	All decisions on requests for information flow.	
FDP_ITC.2	All attempts to import user data, including any security attributes.	
FDP_RIP.2	N/A	
FIA_AFL.1	The reaching of the threshold for the unsuccessful authentication attempts and the actions (e.g. disabling of a terminal) taken and the subsequent, if appropriate, restoration to the normal state (e.g. re-enabling of a terminal).	
FIA_ATD.1	N/A	
FIA_UAU.2	All use of the authentication mechanism.	
FIA_UAU.5	The result of each activated mechanism together with the final decision.	
FIA_UID.2(1)	All use of the user identification mechanism, including the user identity provided.	Unknown user names are not logged, in order to protect passwords from inadvertent disclosure.
FIA_USB.1	Success and failure of binding of user security attributes to a subject (e.g. success or failure to create a subject).	This operation is performed in conjunction with the login process, and is not logged separately.
FMT_MSA.1	All modifications of the values of security attributes.	
FMT_MSA.3	a) Modifications of the default setting of permissive or restrictive rules; b) All modifications of the initial values of security attributes.	
FMT_MTD.1	All modifications to the values of TSF data.	
FMT_SMF.1	Use of the management functions.	
FMT_SMR.1	Modifications to the group of users that are part of a role.	
FPT_FLS.1	Failure of the TSF.	
FPT_RCV.1	a) The fact that a failure or service discontinuity occurred.	Auditing may not be available in this event (audit storage is full). Startup is guaranteed to be logged via FPT_RCV.2.

© THALES Norway AS

TOE Requirement	Event	Note
FPT_RCV.2	a) The fact that a failure or service discontinuity occurred; b) Resumption of the regular operation; c) Type of failure or service discontinuity	
FPT_RCV.4	a) If possible, the impossibility to return to a secure state after failure of a security function; b) If possible, the detection of a failure of a security function.	
FPT_RVM.1	N/A	
FPT_SEP.3	N/A	
FPT_TDC.1	a) Use of the TSF data consistency mechanisms. b) Identification of which TSF data have been interpreted. c) Detection of modified TSF data.	This requirement defines "TSF data" as "object security labels" a) The mechanisms cannot be bypassed. Specific logging is not considered to be informative. b) The TSF data is invariant (security labels), and are always included in log data. c) Any invalid security labels are specifically handled. The IT environment protective measures are used to ensure TSF data integrity.
FPT_TST.1	Execution of the TSF self tests and the results of the tests.	
FTA_SSL.3	Termination of an interactive session by the session locking mechanism.	
FTA_TSE.1(1)	All attempts at establishment of a user session.	

Table 5-2: Auditable Events

FAU_SAA.1
Potential violation analysis

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.

© THALES Norway AS

FAU_SAA.1.2
The TSF shall enforce the following rules for monitoring audited events:

a) Accumulation or combination of *alarms* known to indicate a potential security violation;

b) None

FAU_SAR.1(1)
Audit review

FAU_SAR.1(1).1
The TSF shall provide *authorized users* with the capability to read *all audit information* from the *TOE* audit records.

FAU_SAR.1(1).2
The TSF shall provide the audit records in a manner suitable for the authorized user to interpret the information.

FAU_SAR.2(1)
Restricted audit review

FAU_SAR.2(1).1
The TSF shall prohibit all users read-access to the audit records except those users that have been granted explicit read-access.

FAU_STG.1(1)
Protected audit trail storage

FAU_STG.1(1).1
The TSF shall protect the stored audit records from unauthorized deletion.

FAU_STG.1(1).2
The TSF shall be able to *prevent* unauthorized modifications to the audit records in the audit trail.

FAU_STG.3(1)
Action in case of possible audit data loss

FAU_STG.3(1).1
The TSF shall *generate an alarm* if the audit trail exceeds *80% of available disk space or a configurable limit*.

FAU_STG.4(1)
Prevention of audit data loss

FAU_STG.4(1).1
The TSF shall *shut down the system* if the audit trail is full.

5.1.1.2 Class FDP: User data protection

FDP_ACC.1
Subset access control

FDP_ACC.1.1
The TSF shall enforce the *DAC* on *database objects accessed and operations performed by subjects.*

© THALES Norway AS

FDP_ACF.1
Access control functions

FDP_ACF.1.1
The TSF shall enforce the *DAC* to objects based on the following: *subject identifier, object ownership*.

FDP_ACF.1.2
The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *Use of Command ACLs and Storage ACL.*

FDP_ACF.1.3
The TSF shall explicitly authorize access of subjects to objects based on the following additional rules:

a) *XOMAIL_SA*
If the subject is connected as the Primary Security Administrator user, all XOmail administrative access is allowed.

Note: This applies only to the Primary Security Administrator user, not to all users with SA role.

b) *OS_ROOT*
If the subject is the OS Root, user access to all database objects is allowed.

FDP_ACF.1.4
The TSF shall explicitly deny access of subjects to objects based on the *following additional rules: none*.

FDP_ETC.2
Export of user data with security attributes

FDP_ETC.2.1
The TSF shall enforce the *DAC and the MAC* when exporting user data, controlled under the SFP(s), 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 the following rules when user data is exported from the TSC:

c) *Convert security label from internal representation into the representation required by the export medium.*

FDP_IFC.2
Complete information flow control

FDP_IFC.2.1
The TSF shall enforce the *MAC as stated in B&L [4] on all subjects, all information*, and all operations that cause that information to flow to and from non-trusted subjects covered by the SFP.

FDP_IFC.2.2
The TSF shall ensure that all operations that cause any information in the TSC to flow to and from any subject in the TSC are covered by an information flow control SFP.

© THALES Norway AS

FDP_IFF.2
Hierarchical security attributes

FDP_IFF.2.1
The TSF shall enforce the *MAC as stated in B&L [4]* based on the following types of subject and information security attributes: *subject security clearance (max HCL, NHC, SP), object hierarchical classification level (HCL), object non-hierarchical categories (NHC) and object security policy (SP).*

FDP_IFF.2.2
The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules, based on the ordering relationships between security attributes hold:

a) Read operation: Subject clearance must dominate object label (S >= O).

b) Write operation: Object label must dominate subject clearance (O >= S).

c) RW operation: Subject clearance must be equal to object label (S == O).

FDP_IFF.2.3
The TSF shall enforce the *following additional flow control SFP rules: none*.

FDP_IFF.2.4
The TSF shall provide the following *additional SFP capabilities: none*.

FDP_IFF.2.5
The TSF shall explicitly authorise an information flow based on the following rules:

a) *The Non-Hierarchical category CLEAR shall allow communication of classified information via unsecured communication channels. Information shall be marked CLEAR (according to the interface protocol used), and the original label shall not be transmitted. On reception, CLEAR info shall be marked with a hierarchical security label corresponding to Confidential, and the non-hierarchical security category CLEAR.*

b) *Trusted subjects are allowed to bypass the MAC rules of FDP_IFF.2.2.*

FDP_IFF.2.6
The TSF shall explicitly deny information flow based on the following rules: *none*.

FDP_IFF.2.7
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 two security attributes are incomparable; and

b) There exists a ”least upper bound” in the set of security attributes, such that, given any two 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.

FDP_ITC.2
Import of user data with security attributes

FDP_ITC.2.1
The TSF shall enforce the *DAC and the MAC* when importing user data, controlled under the SFP,

© THALES Norway AS

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 of the user data.

FDP_ITC.2.5
The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC:

a) *If no label is present:*
ACP: The message shall be trapped.
Other channels: The message shall be set to the highest label of the channel.

b) *If the label is invalid:*
ACP: The message shall be trapped.
Other channels: The message shall be discarded and alarm shall be generated.

c) *If the channel is tagged as "System-High", the label shall be set to the highest allowed, and the original label shall be kept as an informative label.*

d) *If the label exceeds the bounds defined for the channel, the message shall be rejected, and an alarm shall be generated.*

e) *If Security Review & Release is activated for the channel, the authorized user shall specify the resulting security label.*

FDP_RIP.2
Full residual information protection

FDP_RIP.2.1
The TSF shall ensure that any previous information content of a resource is made unavailable upon the *allocation of the resource to* all objects.

5.1.1.3 Class FIA: Identification and authentication

FIA_AFL.1
Authentication failure handling

FIA_AFL.1.1
The TSF shall detect *a specified number (default: three) of successive unsuccessful* authentication attempts related to *client logons for all users except the Primary Security Administrator user*.

Note: This applies only to Primary Security Administrator user, not to all users with SA role.

FIA_AFL.1.2
When the defined number of unsuccessful authentication attempts has been met or surpassed, *the TSF shall lock the user account, e.g. the user shall not be able to authenticate until an administrator unlocks the user account.*

© THALES Norway AS

FIA_ATD.1
User attribute definition

FIA_ATD.1.1
The TSF shall maintain the following list of security attributes belonging to individual users:

a) *User identifier*
b) *User clearance*
c) *User command access*
d) *Administrator role.*
e) *Message storage access lists.*
f) *Command ACLs*

FIA_UAU.2
Timing of authentication

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

FIA_UAU.5
Multiple authentication mechanisms

FIA_UAU.5.1
The TSF shall provide *verification of username and password* to support user authentication.

FIA_UAU.5.2
The TSF shall authenticate any user's claimed identify according to the *following rules:*

a) *Use OS password verification mechanisms to verify the username and password.*
b) *Use Kerberos to authenticate user using a Kerberos ticket provided by the OS.*

FIA_UID.2(1)
Timing of identification

FIA_UID.2(1).1
The TSF shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

© THALES Norway AS

FIA_USB.1
User-subject binding

FIA_USB.1.1
The TSF shall associate the following user security attributes with subjects acting on behalf of that user: *User identifier, User Clearance, User Command Access.*

FIA_USB.1.2
The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users:

a) *Initial User identifier shall be identical to OS user identifier with the same username*

b) *Initial User Clearance shall be set from the User Template that the user creation is based on*

FIA_USB.1.3
The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users:

a) *Users shall be logged out when User Clearance has been changed*

b) *Users shall be logged out when User Command Access has been changed.*

5.1.1.4 Class FMT: Security management

FMT_MSA.1
Management of security attributes

FMT_MSA.1.1
The TSF shall enforce the *MAC and the DAC* to restrict the ability to *query, modify or delete* the security attributes:

a) Subject security clearance,

b) Subject administrator access levels,

c) Subject and object ACLs (e.g. command access),

d) Template Permit flag.

to *the Security Administrator role.*

The following exceptions exist:

a) The Administrator role may query and modify department ACLs if sufficient command access is available for the user.

b) A user may grant other users read access to objects owned by the user (the user's own storage).

FMT_MSA.3
Static attribute initialization

FMT_MSA.3.1
The TSF shall enforce the *MAC and DAC* to provide *configurable* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2
The TSF shall allow the *SA role* to specify alternative initial values to override the default values when

© THALES Norway AS

an object or information is created.

FMT_MTD.1
Management of TSF data

FMT_MTD.1.1
The TSF shall restrict the ability to *query, modify, and delete* the *system configuration* to *the Administrator, Network Administrator and Security Administrator roles*.

FMT_SMF.1
Specification of Management Functions

FMT_SMF.1.1
The TSF shall be capable of performing the following security management functions: *TOE security data maintenance*.

FMT_SMR.1
Security roles

FMT_SMR.1.1
The TSF shall maintain the roles *User, Administrator, Network Administrator and Security Administrator.*

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

5.1.1.5 Class FPT: Protection of the TSF

FPT_FLS.1
Fail secure

FPT_FLS.1.1
The TSF shall preserve a secure state when the following types of failures occur: *all*.

FPT_RCV.1
Manual Recovery

FPT_RCV.1.1
After *audit storage has been exhausted* the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

FPT_RCV.2
Automated recovery

FPT_RCV.2.1
When automated recovery from *abnormal termination* is not possible, the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

FPT_RCV.2.2
For *all failures except abnormal termination*, the TSF shall ensure the return of the TOE to a secure state using automated procedures.

© THALES Norway AS

FPT_RCV.4
Function recovery

FPT_RCV.4.1
The TSF shall ensure that *all SFs* have the property that the SF either completes successfully, or for the indicated failure scenarios, recovers to a consistent and secure state.

FPT_RVM.1
Non-bypassability of the TSP

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

FPT_SEP.3
Complete reference monitor

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 information flow control SFPs in a security domain for its own execution that protects them from interference and tampering by the remainder of the TSF and by subjects untrusted with respect to the TSP.

FPT_TDC.1
Inter-TSF basic TSF data consistency

FPT_TDC.1.1
The TSF shall provide the capability to consistently interpret *object security labels* when shared between the TSF and another trusted IT product.

FPT_TDC.1.2
The TSF shall use *the rules defined for the communication channel* when interpreting the TSF data from another trusted IT product.

Note 1: The interpretation will depend on which protocol is used by the originator.

FPT_TST.1
TSF testing

FPT_TST.1.1
The TSF shall run a suite of self-tests *during initial start-up* to demonstrate the correct operation of the TSF.

FPT_TST.1.2
The TSF shall provide authorized users with a capability to verify the integrity of *TSF data*.

FPT_TST.1.3
The TSF shall provide authorized users with the capability to verify the integrity of stored TSF executable code.

© THALES Norway AS

The strength of function FPT_TST.1 is SOF-Medium.

5.1.1.6 Class FTA: TOE access

FTA_SSL.3
TSF-initiated termination

FTA_SSL.3.1
The TSF shall terminate an interactive session after a *configurable period of user inactivity*.

FTA_TSE.1(1)
TOE session establishment

FTA_TSE.1(1).1
The TSF shall be able to deny session establishment based on *the following attributes:*
- the lock attribute for users and administrators,
- the ip address, subnet address or hostname of the client
- the authentication tokens provided

5.1.2 TOE security assurance requirements

The assurance requirements for this Security Target, taken from Part 3 of the CC, comprise the EAL4 level of assurance.

Assurance Class	Assurance components
ACM: Configuration management	ACM_AUT.1 Partial CM automation ACM_CAP.4 Generation support and acceptance procedures ACM_SCP.2 Problem tracking CM coverage
ADO: Delivery and operation	ADO_DEL.2 Detection of modification ADO_IGS.1 Installation, generation, and start-up procedures
ADV: Development	ADV_FSP.2 Fully defined external interfaces ADV_HLD.2 Security enforcing high-level design ADV_IMP.1 Subset of the implementation of the TSF ADV_LLD.1 Descriptive low-level design ADV_RCR.1 Informal correspondence demonstration ADV_SPM.1 Informal TOE security policy model
AGD: Guidance documents	AGD_ADM.1 Administrator guidance AGD_USR.1 User guidance
ALC: Life cycle support	ALC_DVS.1 Identification of security measures ALC_LCD.1 Developer defined life-cycle model ALC_TAT.1 Well-defined development tools
ATE: Tests	ATE_COV.2 Analysis of coverage ATE_DPT.1 Testing: high-level design ATE_FUN.1 Functional testing ATE_IND.2 Independent testing - sample

© THALES Norway AS

AVA: Vulnerability assessment	AVA_MSU.2 Validation of analysis AVA_SOF.1 Strength of TOE security function evaluation AVA_VLA.2 Independent vulnerability analysis

Table 5-3: EAL4

5.1.2.1 Class ACM: Configuration management

ACM_AUT.1
Partial CM automation

ACM_AUT.1.1D
The developer shall use a CM system.

ACM_AUT.1.2D
The developer shall provide a CM plan.

ACM_AUT.1.1.C
The CM system shall provide an automated means by which only authorized changes are made to the TOE implementation representation.

ACM_AUT.1.2C
The CM system shall provide an automated means to support the generation of the TOE.

ACM_AUT.1.3C
The CM plan shall describe the automated tools used in the CM system.

ACM_AUT.1.4C
The CM plan shall describe how the automated tools are used in the CM system.

ACM_AUT.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ACM_CAP.4
Generation support and acceptance procedures

ACM_CAP.4.1D
The developer shall provide a reference for the TOE.

ACM_CAP.4.2D
The developer shall use a CM system.

ACM_CAP.4.3D
The developer shall provide CM documentation.

ACM_CAP.4.1C
The reference for the TOE shall be unique to each version of the TOE.

ACM_CAP.4.2C
The TOE shall be labelled with its reference.

ACM_CAP.4.3C
The CM documentation shall include a configuration list, a CM plan, and an acceptance plan.

ACM_CAP.4.4C
The configuration list shall uniquely identify all configuration items that comprise the TOE.

© THALES Norway AS

ACM_CAP.4.5C
The configuration list shall describe the configuration items that comprise the TOE.

ACM_CAP.4.6C
The CM documentation shall describe the method used to uniquely identify the configuration items.

ACM_CAP.4.7C
The CM system shall uniquely identify all configuration items that comprise the TOE.

ACM_CAP.4.8C
The CM plan shall describe how the CM system is used.

ACM_CAP.4.9C
The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.

ACM_CAP.4.10C
The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.

ACM_CAP.4.11C
The CM system shall provide measures such that only authorized changes are made to the configuration items.

ACM_CAP.4.12C
The CM system shall support the generation of the TOE.

ACM_CAP.4.13C
The acceptance plan shall describe the procedure used to accept modified or newly created configuration items as part of the TOE.

ACM_CAP.4.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ACM_SCP.2
Problem tracking CM coverage

ACM_SCP.2.1D
The developer shall provide a list of configuration items for the TOE.

ACM_SCP.2.1C
The list of configuration items shall include the following: implementation representation; security flaws; and the evaluation evidence required by the assurance components in the ST.

ACM_SCP.2.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.1.2.2 Class ADO: Delivery and operation

ADO_DEL.2
Detection of modification.

ADO_DEL.2.1D
The developer shall document procedures for delivery of the TOE or parts of it to the user.

ADO_DEL.2.2D
The developer shall use the delivery procedures.

© THALES Norway AS

ADO_DEL.2.1C
The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a users' site.

ADO_DEL.2.2C
The delivery documentation shall describe how the various procedures and technical measures provide for the detection of modifications, or any discrepancy between the developer's master copy and the version received at the user site.

ADO_DEL.2.3C
The delivery documentation shall describe how the various procedures allow detection of attempts to masquerade as the developer, even in cases in which the developer has sent nothing to the user's site.

ADO_DEL.2.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADO_IGS.1
Installation, generation, and start-up procedures

ADO_IGS.1.1D
The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE.

ADO_IGS.1.1C
The installation, generation and start-up documentation shall describe the steps necessary for secure installation, generation, and start-up of the TOE.

ADO_IGS.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADO_IGS.1.2E
The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration.

5.1.2.3 Class ADV: Development

ADV_FSP.2
Fully defined external interfaces

ADV_FSP.2.1D
The developer shall provide a functional specification.

ADV_FSP.2.1C
The functional specification shall describe the TSF and its external interfaces using an informal style.

ADV_FSP.2.2C
The functional specification shall be internally consistent.

ADV_FSP.2.3C
The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing complete details of all effects, exceptions and error messages.

ADV_FSP.2.4C
The functional specification shall completely represent the TSF.

© THALES Norway AS

ADV_FSP.2.5C
The functional specification shall include rationale that the TSF is completely represented.

ADV_FSP.2.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.2.2E
The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional requirements.

ADV_HLD.2
Security enforcing high-level design

ADV_HLD.2.1D
The developer shall provide the high-level design of the TSF.

ADV_HLD.2.1C
The presentation of the high-level design shall be informal.

ADV_HLD.2.2C
The high-level design shall be internally consistent.

ADV_HLD.2.3C
The high-level design shall describe the structure of the TSF in terms of subsystems.

ADV_HLD.2.4C
The high-level design shall describe the security functionality provided by each subsystem of the TSF.

ADV_HLD.2.5C
The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software.

ADV_HLD.2.6C
The high-level design shall identify all interfaces to the subsystems of the TSF.

ADV_HLD.2.7C
The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.

ADV_HLD.2.8C
The high-level design shall describe the purpose and method of use of all interfaces to the subsystems of the TSF, providing details of the effects, exceptions and error messages, as appropriate.

ADV_HLD.2.9C
The high-level design shall describe the separation of the TOE into TSP-enforcing and other subsystems.

ADV_HLD.2.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_HLD.2.2E
The evaluator shall determine that the high-level design is an accurate and complete instantiation of the TOE security functional requirements.

ADV_IMP.1
Subset of the implementation of the TSF

© THALES Norway AS

ADV_IMP.1.1D
The developer shall provide the implementation representation for a selected subset of the TSF.

ADV_IMP.1.1C
The implementation representation shall unambiguously define the TSF to a level of detail such that the TSF can be generated without further design decisions.

ADV_IMP.1.2C
The implementation representation shall be internally consistent.

ADV_IMP.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_IMP.1.2E
The evaluator shall determine that the least abstract TSF representation provided is an accurate and complete instantiation of the TOE security functional requirements.

ADV_LLD.1
Descriptive low-level design

ADV_LLD.1.1D
The developer shall provide the low-level design of the TSF.

ADV_LLD.1.1C
The presentation of the low-level design shall be informal.

ADV_LLD.1.2C
The low-level design shall be internally consistent.

ADV_LLD.1.3C
The low-level design shall describe the TSF in terms of modules.

ADV_LLD.1.4C
The low-level design shall describe the purpose of each module.

ADV_LLD.1.5C
The low-level design shall define the interrelationships between the modules in terms of provided security functionality and dependencies on other modules.

ADV_LLD.1.6C
The low-level design shall describe how each TSP-enforcing function is provided.

ADV_LLD.1.7C
The low-level design shall identify all interfaces to the modules of the TSF.

ADV_LLD.1.8C
The low-level design shall identify which of the interfaces to the modules of the TSF are externally visible.

ADV_LLD.1.9C
The low-level design shall describe the purpose and method of use of all interfaces to the modules of the TSF, providing details of effects, exceptions and error messages, as appropriate.

ADV_LLD.1.10C
The low-level design shall describe the separation of the TOE into TSP-enforcing and other modules.

ADV_LLD.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

© THALES Norway AS

ADV_LLD.1.2E
The evaluator shall determine that the low-level design is an accurate and complete instantiation of the TOE security functional requirements.

ADV_RCR.1
Informal correspondence demonstration

ADV_RCR.1.1D
The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.

ADV_RCR.1.1C
For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.

AV_RCR.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_SPM.1
Informal TOE security policy model

ADV_SPM.1.1D
The developer shall provide a TSP model.

ADV_SPM.1.2D
The developer shall demonstrate the correspondence between the functional specification and the TSP model.

ADV_SPM.1.1C
The TSP model shall be informal.

ADV_SPM.1.2C
The TSP model shall describe the rules and characteristics of all policies of the TSP that can be modeled.

ADV_SPM.1.3C
The TSP model shall include a rationale that demonstrates that it is consistent and complete with respect to all policies of the TSP that can be modeled.

ADV_SPM.1.4C
The demonstration of correspondence between the TSP model and the functional specification shall show that all of the security functions in the functional specification are consistent and complete with respect to the TSP model.

ADV_SPM.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.1.2.4 Class AGD: Guidance document

AGD_ADM.1
Administrator guidance

AGD_ADM.1.1D

© THALES Norway AS

The developer shall provide administrator guidance addressed to system administrative personnel.

AGD_ADM.1.1C
The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE.

AGD_ADM.1.2C
The administrator guidance shall describe how to administer the TOE in a secure manner.

AGD_ADM.1.3C
The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment.

AGD_ADM.1.4C
The administrator guidance shall describe all assumptions regarding user behavior that are relevant to secure operation of the TOE.

AGD_ADM.1.5C
The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate.

AGD_ADM.1.6C
The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_ADM.1.7C
The administrator guidance shall be consistent with all other documentation supplied for evaluation.

AGD_ADM.1.8C
The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator.

AGD_ADM.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_USR.1
User guidance

AGD_USR.1.1D
The developer shall provide user guidance.

AGD_USR.1.1C
The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE.

AGD_USR.1.2C
The user guidance shall describe the use of user-accessible security functions provided by the TOE.

AGD_USR.1.3C
The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment.

AGD_USR.1.4C
The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behavior found in the statement of TOE security environment.

AGD_USR.1.5C
The user guidance shall be consistent with all other documentation supplied for evaluation.

© THALES Norway AS

AGD_USR.1.6C
The user guidance shall describe all security requirements for the IT environment that are relevant to the user.

AGD_USR.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.1.2.5 Class ALC: Life cycle support

ALC_DVS.1
Identification of security measures

ALC_DVS.1.1D
The developer shall produce development security documentation.

ALC_DVS.1.1C
The development security documentation shall describe all the physical procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.

ALC_DVS.1.2C
The development security documentation shall provide evidence that these security measures are followed during development and maintenance of the TOE.

ALC_DVS.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_DVS.1.2E
The evaluator shall confirm that the security measures are being applied.

ALC_LCD.1
Developer defined life-cycle model

ALC_LCD.1.1D
The developer shall establish a life-cycle model to be used on the development and maintenance of the TOE.

ALC_LCD.1.2D
The developer shall provide life-cycle definition documentation.

ALC_LCD.1.1C
The life-cycle definition documentation shall describe the model used to develop and maintain the TOE.

ALC_LCD.1.2C
The life-cycle model shall provide for the necessary control over the development and maintenance of the TOE.

ALC_LCD.1.1E
The evaluator shall confirm that the information provided meet all requirements for content and presentation of evidence.

ALC_TAT.1

© THALES Norway AS

Well-defined development tools

ACL_TAT.1.1D
The developer shall identify the development tools being used for the TOE.

ALC_TAT.1.2D
The developer shall document the selected implementation-dependent option for the development tools.

ACL_TAT.1.1C
All development tools used for implementation shall be well defined.

ACL_TAT.1.2C
The documentation of the development tools shall unambiguously define the meaning of all statements used in the implementation.

ACL_TAT.1.3C
The documentation of the development tools shall unambiguously define the meaning of all implementation-dependent options.

ACL_TAT.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.1.2.6 Class ATE: Tests

ATE_COV.2
Analysis of coverage

ATE_COV.2.1D
The developer shall provide an analysis of the test coverage.

ATE_COV.2.1C
The analysis of the test coverage shall demonstrate the correspondence between the tests identified in the test documentation and the TSF as describe in the functional specification.

ATE_COV.2.2C
The analysis of the test coverage shall demonstrate that the correspondence between the TSF as described in the functional specification and the tests identified in the test documentation is complete.

ATE_COV.2.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_DPT.1
Testing: high-level design

ATE_DPT.1.1D
The developer shall provide the analysis of the depth of testing.

ATE_DPT.1.1C
The depth analysis shall demonstrate that the tests identified in the test documentation are sufficient to demonstrate that the TSF operates in accordance with its high-level design.

ATE_DPT.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

© THALES Norway AS

ATE_FUN.1
Functional testing

ATE_FUN.1.1D.
The developer shall test the TSF and document the results.

ATE_FUN.1.2D.
The developer shall provide test documentation.

ATE_FUN.1.1C.
The test documentation shall consist of test plans, test procedure descriptions, expected test results and actual test results.

ATE_FUN.1.2C.
The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed.

ATE_FUN.1.3C.
The test procedure descriptions shall identify the tests to be performed and describe the scenarios for testing each security function. These scenarios shall include any ordering dependencies on the results of other tests.

ATE_FUN.1.4C.
The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE_FUN.1.5C.
The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified.

ATE_FUN.1.1E.
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.2
Independent testing – sample

ATE_IND.2.1D.
The developer shall provide the TOE for testing.

ATE_IND.2.1C.
The TOE shall be suitable for testing.

ATE_IND.2.2C.
The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

ATE_IND.2.1E.
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.2.2E.
The evaluator shall test a subset of the TSF as appropriate to confirm that the TOE operates as specified.

ATE_IND.2.3E.
The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

© THALES Norway AS

5.1.2.7 Class AVA: Vulnerability assessment

AVA_MSU.2
Validation of analysis

AVA_MSU.2.1D.
The developer shall provide guidance documentation.

AVA_MSU.2.2D.
The developer shall document an analysis of the guidance documentation.

AVA_MSU.2.1C
The guidance documentation shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AVA_MSU.2.2C
The guidance documentation shall be complete, clear, consistent and reasonable.

AVA_MSU.2.3C
The guidance documentation shall list all assumptions about the intended environment.

AVA_MSU.2.4C
The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).

AVA_MSU.2.5C
The analysis shall demonstrate that the guidance is complete.

AVA_MSU.2.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_MSU.2.2E
The evaluator shall repeat all configuration and installation procedures, and other procedures selectively, to confirm that the TOE can be configured and used securely using only the supplied documentation.

AVA_MSU.2.3E
The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected.

AVA_MSU.2.4E
The evaluator shall confirm that the analysis documentation shows that the guidance is provided for secure operation in all modes of operation of the TOE.

AVA_SOF.1
Strength of TOE security function evaluation

AVA_SOF.1.1D
The developer shall perform a strength of TOE security function analysis for each mechanism identified in the ST as having a strength of TOE security function claim.

AVA_SOF.1.1C
For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the ST.

AVA_SOF.1.2C
For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets of exceeds the specific strength of function metric defined in the ST

© THALES Norway AS

AVA_SOF.1.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_SOF.1.2E
The evaluator shall confirm that the strength claims are correct.

AVA_VLA.2
Independent vulnerability analysis

AVA_VLA.2.1D
The developer shall perform a vulnerability analysis.

AVA_VLA.2.2D
The developer shall provide vulnerability analysis documentation.

AVA_VLA.2.1C
The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for ways in which a user can violate the TSP.

AVA_VLA.2.2C
The vulnerability analysis documentation shall describe the disposition of identified vulnerabilities.

AVA_VLA.2.3C
The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

AVA_VLA.2.4C
The vulnerability analysis documentation shall justify that the TOE, with the identified vulnerabilities, is resistant to obvious penetration attacks.

AVA_VLA.2.1E
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VLA.2.2E
The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure the identified vulnerabilities have been addressed.

AVA_VLA.2.3E
The evaluator shall perform an independent vulnerability analysis.

AVA_VLA.2.4E
The evaluator shall perform independent penetration testing, based on the independent vulnerability analysis, to determine the exploitability of the additional identified vulnerabilities in the intended environment.

AVA_VLA.2.5E
The evaluator shall determine that the TOE is resistant to penetration attacks performed by an attacker possessing a low attack potential.

5.2 Security requirements for the IT environment

5.2.1 IT environment security functional requirements

5.2.1.1 Class FAU: Security audit

FAU_GEN.1(2)

Audit data generation.

FAU_GEN.1(2).1
The *IT environment* shall be able to generate audit records of the following auditable events:

a) Start-up and shutdown of the audit functions.

b) All auditable events for the *basic* level of audit*, as listed in*

FTA_TSE.1(2)	All attempts at establishment of a user session.	

c) Table 5-4 and

d) Specifically defined auditable events:

1) Audit events initiated by the TOE.

FAU_GEN.1(2).2
The *IT environment* shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event.

b) For each audit event type, based on the auditable event definitions of the functional components included in the ST, *message identifier for selected audit event types generated by the TOE.*

FAU_GEN.2(2)
User identity association

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

IT Environment Requirement	Event	Note
FAU_GEN.1(2)	N/A	
FAU_GEN.2(2)	N/A	
FAU_SAR.1(2)	Reading of information from the audit records.	
FAU_SAR.2(2)	Unsuccessful attempts to read information from the audit records.	
FAU_STG.1(2)	N/A	
FAU_STG.3(2)	Actions taken due to exceeding of a threshold.	
FIA_UAU.2	All use of the authentication mechanism.	
FIA_UID.2(2)	All use of the user identification mechanism, including the user identity provided.	
FDP_ITT.1	All attempts to transfer data, including the protection method used and any errors that occurred.	
FPT_ITT.1	N/A	

© THALES Norway AS

FPT_STM.1	Changes to the time.	
FTA_TSE.1(2)	All attempts at establishment of a user session.	

Table 5-4: Auditable Events

FAU_SAR.1(2)
Audit review

FAU_SAR.1(2).1
The *IT environment* shall provide *authorized OS-users* with the capability to read *all audit information* from the *OS* audit records.

FAU_SAR.1(2).2
The *IT environment* shall provide the audit records in a manner suitable for the authorized OS-user to interpret the information.

FAU_SAR.2(2) Restricted audit review

FAU_SAR.2(2).1
The *IT environment* shall prohibit all users read-access to the audit records except those users that have been granted explicit read-access.

FAU_STG.1(2)
Protected audit trail storage

FAU_STG.1(2).1
The *IT environment* shall protect the stored audit records from unauthorized deletion.

FAU_STG.1(2).2
The *IT environment* shall be able to *prevent* unauthorized modifications to the audit records in the audit trail.

FAU_STG.3(2)
Action in case of possible audit data loss

FAU_STG.3(2).1
The *IT environment* shall *be able to suspend system operation or overwrite the oldest stored audit records* if the audit trail exceeds *a configurable size*.

5.2.1.2 Class FIA: Identification and authentication

FIA_UAU.2
User authentication before any action

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

FIA_UID.2(2)
User identification before any action

© THALES Norway AS

FIA_UID.2(2).1
The *IT environment* shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

5.2.1.3 Class FDP: User Data Protection

FDP_ITT.1
Basic internal transfer protection

FDP_ITT.1.1
The *IT environment* shall enforce the *access control SFP(s) and information flow control SFP(s)* to prevent the *disclosure and modification* of user data when it is transmitted between physically-separated parts of the TOE.

5.2.1.4 Class FPT: Protection of the TSF

FPT_ITT.1
Basic internal TSF data transfer protection

FPT_ITT.1.1
The *IT environment* shall protect TSF data from *disclosure and modification* when it is transmitted between separate parts of the TOE.

FPT_STM.1
Time stamps.

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

5.2.1.5 Class FTA: TOE Access

FTA_TSE.1(2)
TOE session establishment

FTA_TSE.1(2).1
The *IT environment* shall be able to deny session establishment based on *the following attributes:*
- the type of traffic (management or messaging)
- the ip address, subnet address or hostname of the client

© THALES Norway AS

6. TOE SUMMARY SPECIFICATION

This describes the security functions provided by the TOE to meet the security functional requirements specified. Furthermore a statement of assurance measures specifies the assurance measures of the TOE that are claimed to satisfy the stated assurance requirements.

6.1 TOE security functions

6.1.1 SF.AUDIT

Audit data is stored in the TOE secure database (SS) and through OS auditing mechanisms. The following operations are auditable:

1) Message operations performed by users.
   - Show
   - Print
   - Delete
   - Change
     Does not apply to Draft messages
   - Accept
   - Send
   - Refuse
   - Change Label
   - Copy
   - Forward
   - Reply
   - Export message
   - Export attachment
   - Import message
   - Create from message
   - Restore deleted message

2) Message reception and transmission.

3) All administrator commands.

4) Whether each operation or command succeeded or was denied.

5) User login attempts, both successful and failed, and logout.

6) User lockout after a number of unsuccessful logins.

7) All changes to a user's command access.

8) System failures.

9) Self tests and self test results.

10) Start-up and shutdown of the XOmail Server.
    Note that the audit functions cannot be disabled,
    and are started before any auditable events can be performed.

© THALES Norway AS

The TOE associates a user identity with all records where applicable.

It is possible for the Security Administrators and Primary Security Administrator to configure alarm descriptors for each alarm type. The alarm descriptors include e.g. alarm severity level, and whether successive alarm events in a configurable period shall raise the corresponding alarm.

Additionally, Security Administrators and Primary Security Administrator may configure alarms to be printed, or stored in an external text file on the server, in addition to the secure database. The external file is protected by OS DAC. MAC is enforced when printing alarms.

The XOmail audit log is stored in the XOmail database protected by SF.DAC and SF.MAC. DAC is evaluated for each audit log (e.g. system log), and MAC is evaluated for each record. A user must pass both mechanisms to gain access to audit records.

It is possible to add audit records until the hard disk is full. An alarm will be issued when disk utilization exceeds a configurable limit. When the hard disk is full, XOmail will shut down. The TOE can be started manually after sufficient disk space has been made available in the TOE environment. If an error occurred during shutdown, recovery is handled by SF.DB_SELF_TEST.

When the TOE has initiated the auditing via the OS system call, the responsibility for correct audit handling is transferred to the OS.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.2 SF.AUTHENTICATION

The SF ensures that only authenticated users get access to data held within the TOE. If correct authentication tokens can be provided, the SF associates a role and a security clearance with the user.

SOF measures are not defined for this function as the probabilistic parts of the function is implemented in the OS.

6.1.3 SF.AUTO_LOGOUT

The SF ensures that sessions can be terminated after a period of inactivity. The length of the period of inactivity is configurable.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.4 SF.CLASSIFICATION_TAG

If Mail Guard functionality is activated for a given system unit, all messages (with exception of Reports, Notifications and System Generated Messages) are checked for a valid classification tag. Messages that do not contain a valid tag line are rejected. The set of acceptable tags is part of the system configuration.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

© THALES Norway AS

6.1.5 SF.CLEAR

A CLEAR policy offers the opportunity for (authorized) users, forcibly and deliberately, to initiate the transmission of classified messages over a communication channel, despite its rejection by the basic Communication Policy ("Sending Sufficiency").

The CLEAR policy has three major components:

- The CLEAR-MARKING policy which enables the marking of a message in order for the Communication policy to recognize it
- The CLEAR-SENDING policy, which recognizes a CLEAR message, and allows its transmission. This policy is essentially a part of the Communication Policy ("Sending Exception")
- The CLEAR-RECEPTION policy which recognizes a cleared message upon reception, and enforces that such a message is hierarchically labelled as "CONFIDENTIAL", and non-hierarchically labelled as "Clear"

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.6 SF.COMMAND_ACCESS

Administrative access is configurable on a User Template basis. For every operation on an Admin Main Object or Admin Object it is possible to grant or deny access for users based on the User Template currently edited. Role assignments furthermore apply some restrictions on how command access can be configured.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.7 SF.COMMUNICATION_SECURITY

The interfaces to external channels perform the following security functions:

- Correct labelling of incoming and outgoing messages.
- Evaluate needs for Secure Associations
  The TOE ensures that classified information is sent on secure lines only. It is the responsibility of system administrators to define which lines are secure. See XOmail Administrator's Guide[3] for details. Note that SF.CLEAR provides a controlled override of this policy.
- Provision of a Secure Association Service based on network status and/or manual settings by Security Administrators.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

© THALES Norway AS

6.1.8 SF.DAC

The Discretionary Access Control (DAC) is always invoked when a subject requests access to tables. If an ACL is defined for the table, that ACL is used for identifying the allowable types of access. If an ACL does not exist, only the table owner is allowed to access the table and its content.

Every user must furthermore be explicitly authorized before being given access to the MailClient, the TSClient or the AdminClient. Authorization is accomplished by requiring users to log on to the applications they use. The Discretionary Access Control (DAC) is inhibited for the OS Root.

A subject must pass both MAC and DAC to access an object.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.9 SF.DB_SELF_TEST

During XOmail start-up, the system database performs a self-test. The test includes a check for whether the database was cleanly shut down, or not. If the database was not cleanly shut down, an internal check for consistency is performed. The check is algorithmic, i.e. for each field it is evaluated whether the value is correct or incorrect. The function also corrects incorrect values, and ensures that XOmail enters a secure state upon startup.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.10 SF.EXECUTION_DOMAINS

The TOE will ensure that execution domains are separated, by using both TSF and OS mechanisms. The OS is responsible for keeping processes separated, and for zeroing memory for processes when created and disk space when allocated. The TOE explicitly classifies processes as trusted or untrusted. Communication between the two classes is brokered by a reference monitor. The TOE will prevent inadvertent disclosure of information by zeroing new database objects.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.11 SF.LABEL_TRANSFORM

During their lifetime messages may be converted between different format representations. Within XOmail four different formats exists, i.e. a human-readable format, the ACP127 format, the STANAG 4406 format and the XOmail internal storage format. The value set for security classification need not be the same for the four different format representations.

There exists a predefined and unambiguous way to transform the security label between these four representations. This transformation is performed by a trusted function. The transformation handles cases with syntactical or semantic errors in a security label, and cases where a security label cannot be represented in the target format. These result in the transformation not taking place with a subsequent failure in processing.

© THALES Norway AS

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.12 SF.LABELLING

On the ACP127-interface and X.400-interface, messages may arrive without labelling or with a label that is not possible to determine. In these cases the messages shall be trapped (ACP), discarded (X.400, invalid label) or labelled with maximum label for the current channel (X.400, without label).

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.13 SF.LOCK

All user accounts can be locked. The TSF supports both manual and automatic locking. A user with the Administrator, Network Administrator or Security Administrator role initiates the manual locking. The automatic locking is initiated by a succession of unsuccessful logon attempts.

This function is a separate function from the locking mechanism implemented in the operating system, and allows for users to be locked out from the TOE while still being allowed to use the operation system in the TOE environment.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.14 SF.MAC

The Mandatory Access Control (MAC) implements access rights according to hierarchical classification level (HCL), non-hierarchical category (NHC) and security policy (SP). For MAC evaluation, SP is handled in the same way as the NHC. A subject must pass both MAC and DAC to access an object.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.15 SF.ROLES

The TOE maintains four different roles (administrator access levels): *User* (no administrator access), *Administrator*, *Network Administrator* and *Security Administrator*. Every authenticated used is associated with a role during logon.

An administrator role is required to access the AdminClient and TSClient. Additionally, the user must be explicitly granted access to commands. The role defines the set of commands a user may be granted.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

© THALES Norway AS

6.1.16 SF.SECURE_STATE_RECOVERY

Security functions of the XOmail system are designed so that they either succeed and lead to a new secure state, or fail and then return to the previous secure state.

Upon fatal failure of the XOmail system, the system is automatically shut down and the current state is preserved for later use. The secure state is restored via the SF.DB_SELF_TEST security function.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.17 SF.SUBNET_RESTRICTION

XOmail Client logon may be restricted to be from a specific IP address, a specific hostname, or a specific IP subnet only. These restrictions can be configured on a per-storage and per-user basis. That is, a user may have access to storages according to ACLs, but SF.SUBNET_RESTRICTION restricts the locations from which the storage can be accessed.

SOF measures are not defined for this function, as it is not realised by probabilistic or permutational mechanisms.

6.1.18 SF.VALIDATE

The validate security function shall be able to verify the integrity of both TSF data and TSF executable code. The validation mechanism produces checksums that must be compared with the developer provided checksums.

The SOF measure for this function is SOF-medium.

© THALES Norway AS

6.2 Assurance measures

Assurance components	Assurance Measures	
ACM_AUT.1	Partial CM automation	CM plan
ACM_CAP.4	Generation support and acceptance procedures	CM plan
ACM_SCP.2	Problem tracking CM coverage.	CM plan
ADO_DEL.2	Detection of modification	CM plan “Programvareleveranse – XOmail”
ADO_IGS.1	Installation, generation and start-up procedures.	712 27734 AXAA EO XOmail Installation and Configuration Guide [1]
ADV_FSP.2	Fully defined external interfaces	MHS Security Concept and Design
ADV_HLD.2	Security enforcing high-level design.	MHS Security Concept and Design
ADV_IMP.1	Subset of the implementation of the TSF	Selected source code modules.
ADV_LLD.1	Descriptive low-level design	MHS Security Concept and Design
ADV_RCR.1	Informal correspondence demonstration	MHS Security Concept and Design
ADV_SPM.1	Informal TOE security policy model	XOmail Security Policy Model
AGD_ADM.1	Administrator Guidance	739 20561 ABAA EO XOmail Administrator’s Guide
AGD_USR.1	User Guidance	739 20529 ABAA EO XOmail User’s Guide
ALC_DVS.1	Identification of security measures.	“Grunnlagsdokument for Sikkerhet”
ALC_LCD.1	Developer defined life-cycle model	CM plan
ALC_TAT.1	Well-defined development tools	CM plan
ATE_COV.2	Analysis of coverage	MHS Security Concept and Design
ATE_DPT.1	Testing: high-level design	MHS Security Concept and Design
ATE_FUN.1	Functional testing	CM plan
ATE_IND.2	Independent testing – sample	Performed by an independent evaluation agency.
AVA_MSU.2	Validation of analysis	MHS Security Concept and Design
AVA_SOF.1	Strength of TOE security function evaluation.	MHS Security Concept and Design
AVA_VLA.2	Independent vulnerability analysis	MHS Security Concept and Design

Table 6-1: Assurance measures

© THALES Norway AS

7. PP CLAIMS

There are no PP claims.

© THALES Norway AS

8. RATIONALE

The rationale demonstrates that threats, assumptions and policies form a basis for the definition of security objectives. Likewise, it is demonstrated that the chosen security requirements cover all security objectives, and that security functions in the TOE or its environment fully cover the security requirements.

8.1 Security objectives rationale

In the following subsections every security objective is correlated with identified threats and assumptions. It is furthermore shown that all identified threats are covered by a security objective.

The following three tables (Table 8-1, Table 8-2 and Table 8-3) demonstrate that all threats, assumption and policies are covered by a security objective. Some threats are fully covered by a single security objective, while others need more than one security objective to be fully covered.

Threats / Security objectives	TT.ADM_ERROR	TT.AUDIT_FAILURE	TT.COM_INTEGRITY	TT.DOS	TT.INSERTION	TT.MASQUERADE	TT.MONITORING	TT.REPLAY	TT.UNATTENDED	TT.UNAUTH_ACCESS	TE.AUDIT_FAILURE	TE.DELIVERY	TE.DOS	TE.IMPROPER_INST	TE.POOR_DESIGN	TE.POOR_IMPL	TE.UNATTENDED
O.ACCESS_HIST		X				X		X		X							
O.AUDIT	X	X		X		X		X		X							X
O.AUTO_LOGOUT									X								
O.CMD_ACL	X									X							X
O.CMD_LOG	X	X		X		X		X		X							X
O.DAC	X									X							
O.ID_AUTH						X		X	X	X							
O.LABELLING										X							
O.LOCK										X							
O.MAC	X									X							
O.MAC_INTEGRITY										X							
O.MANAGE	X																
O.MESSAGING										X							
O.RECOVER				X						X			X			X	
O.REF_MONITOR										X							
O.RESOURCE_SHARE				X						X							
O.REUSE						X		X									
O.ROLE_MNG	X									X							
O.ROLES	X									X							
O.SELF_TEST				X						X							
OE.ACCOUNTABLE	X																
OE.AUDIT	X									X	X		X				X
OE.ID_AUTH																	
OE.NETWORK			X	X	X		X	X		X			X		X	X	
OE.TRAF_SEPARATION							X								X	X	
NOE.ADM_TRUST	X													X			
NOE.INSTALL												X		X			
NOE.PHYSICAL			X	X	X		X	X		X			X		X	X	X

Table 8-1: TOE threats coverage

Assuptions and policies / Security objectives	A.ADM_TRAINING	A.AUDIT_REVIEW	A.CONFIDENCE	A.INVALIDATE	A.NETWORK	A.NOTIFY	A.PHYSICAL	A.PHYSICAL_LOC	A.OS	A.USR_TRAINING
O.ACCESS_HIST										
O.AUDIT										
O.AUTO_LOGOUT										
O.CMD_ACL										
O.CMD_LOG										
O.DAC										
O.ID_AUTH									X	
O.LABELLING										
O.LOCK										
O.MANAGE										
O.MAC										
O.MAC_INTEGRITY										
O.MESSAGING										
O.RECOVER										
O.REF_MONITOR										
O.RESOURCE_SHARE										
O.REUSE										
O.ROLE_MNG										
O.ROLES										
O.SELF_TEST										
OE.ACCOUNTABLE	X	X								X
OE.AUDIT									X	
OE.ID_AUTH									X	
OE.NETWORK					X				X	
OE.TRAF_SEPARATION					X					
NOE.ADM_TRUST			X	X						
NOE.INSTALL	X	X		X		X				X
NOE.PHYSICAL							X	X		

Table 8-2: Assumptions coverage

Assuptions and policies / Security objectives	P.ACCOUNTING	P.CLASSIFICATION	P.CLEAR	P.DAC	P.INTEGRITY	P.INTERFACE_CONTROL	P.MAC	P.MARKING	P.PROTECTION
O.ACCESS_HIST	X								

© THALES Norway AS

Assuptions and policies / Security objectives	P.ACCOUNTING	P.CLASSIFICATION	P.CLEAR	P.DAC	P.INTEGRITY	P.INTERFACE_CONTROL	P.MAC	P.MARKING	P.PROTECTION
O.AUDIT	X								
O.AUTO_LOGOUT									
O.CMD_ACL									
O.CMD_LOG	X								
O.DAC				X					
O.ID_AUTH	X								
O.LABELLING								X	
O.LOCK									
O.MANAGE									
O.MAC		X	X				X		
O.MAC_INTEGRITY							X		
O.MESSAGING									
O.RECOVER									
O.REF_MONITOR									
O.RESOURCE_SHARE									
O.REUSE									
O.ROLE_MNG									
O.ROLES							X		
O.SELF_TEST									
OE.ACCOUNTABLE	X								
OE.AUDIT	X								
OE.ID_AUTH									
OE.NETWORK					X				X
OE.TRAF_SEPARATION									
NOE.ADM_TRUST									
NOE.INSTALL						X			X
NOE.PHYSICAL									X

Table 8-3: Policies coverage

8.1.1 TT.ADM_ERROR

Objectives covering this threat:

- O.AUDIT, O.CMD_LOG, OE.AUDIT
  Knowing that security relevant actions are audited makes this a less likely occurrence, as administrators know that they can be held accountable for their actions. The intention is that administrators perhaps will think twice/double check before initiating a command when knowing that the command is audited. Auditing will also make it possible to identify and correct unwanted operations.

- O.CMD_ACL
  Command ACLs makes it possible to restrict administrative access to a minimum for every administrator. This in turn makes administrative errors less likely to happen.

© THALES Norway AS

- O.MANAGE
  Providing means for effective management of the TOE reduces the possibility of unintentional administrator errors.

- O.ROLE_MNG
  Managing role associations makes it possible to avoid wrong role associations.

- O.ROLES
  By introducing roles the administrative responsibilities are defined. Well-defined responsibilities reduce the possibility of administrative errors.

- NOE.ADM_TRUST
  Ensuring that administrators are trustworthy is an essential countermeasure for unintentional administrative errors. Trustworthy personnel improve the quality of the work performed.

8.1.2 TT.AUDIT_FAILURE

Objectives covering this threat:

- O.ACCESS_HIST
  Access history is useful when audit failure already has occurred. It can help determining from where and by whom a possible attack was performed. Other reasons for audit failure may also be determined from the access history.

- O.AUDIT
  The objective shall prevent audit failures. Audit failures include audit overflow, erroneous audit data and missing audit records.

- O.CMD_LOG
  Logging administrative commands will ensure that it is possible to detect actions that may lead to audit failure.

8.1.3 TT.COM_INTEGRITY

Objectives covering this threat:

- O.DIG_SIGN
  The objective ensures integrity of message content, and discovery of integrity violations.

- OE.NETWORK
  The objective ensures integrity of messages transmitted over the network.

- OE.PHYSICAL
  The objective ensures integrity by restricting physical access.

8.1.4 TT.DOS

Objectives covering this threat:

© THALES Norway AS

- O.AUDIT
  The audit can be used to detect possible DOS attacks so that proper measures can be applied in an early stage of the attack.

- O.CMD_LOG
  The command log can in certain situations both reveal by whom and from where a DOS attack was launched. Having identified by whom and from where the attack is coming, it is considerably easier to assign countermeasures.

- O.RECOVER
  The objective covers recovery from possible DOS attacks.

- O.RESOURCE_SHARE.
  The objective ensures that no entity can block others from using the shared resource.

- O.SELF_TEST.
  Ensuring that a database self test is being performed during start-up reduces the chances of the TOE entering an unknown state after a DOS attack.

- OE.NETWORK.
  The objective reduces the possibility for DOS attacks met by the TOE as it becomes considerably more difficult to send data resulting in TOE resource exhaustion. The network protection will to a certain degree prevent external sources from accessing the computer equipment hosting the TOE.

- NOE.PHYSICAL
  The objective reduces the possibility for DOS attacks met by the TOE as it becomes considerably more difficult to send data resulting in TOE resource exhaustion. The physical protection will to a certain degree prevent unauthorized sources from accessing the computer equipment hosting the TOE, thereby complicating the DOS attack. Other channels than physical access to the TOE must be used.

8.1.5 TT.INSERTION

- O.DIG_SIGN
  The objective makes it possible verify the originator of digitally signed messages.

- OE.NETWORK
  The objective ensures unauthorized entities are prevented from accessing the network, thus preventing unauthorized data insertion.

- NOE.PHYSICAL
  The objective ensures unauthorized entities are prevented from accessing the TOE, thus preventing unauthorized data insertion.

8.1.6 TT.MASQUERADE

Objectives covering this threat:

- O.ACCESS_HIST
  The objective will provide means for detection of entity masquerading. The objective itself is not to perform detection of masquerading, but to provide evidence of access that can be crosschecked with the authorized entities.

- O.AUDIT
  Auditing operations issued during a masquerade attack can help both in determining from where and by whom the attack was launched. It may also to a certain degree help in reversing actions performed as part of the attack.

- O.CMD_LOG
  Logging commands issued during a masquerade attack can help both in determining from where and by whom the attack was launched. It may also to a certain degree help in reversing actions performed as part of the attack.

- O.ID_AUTH
  Provided good authentication mechanisms it will be harder to masquerade as entities in the network.

- O.REUSE.
  Proper reuse of resources prevents attackers from being able to retrieve information that later can be used in a masquerade attack.

8.1.7 TT.MONITORING

Objectives covering this threat:

- OE.NETWORK
  Monitoring by intercepting the network traffic between two distinct TOE locations becomes considerably more difficult as the traffic has to be decrypted first.

- OE.TRAF_SEPARATION
  Monitoring is significantly more difficult when traffic separation is implemented. Users without privileges for a specific traffic type cannot easily monitor that traffic. This is because the attacker must first find a way to gain access to the traffic that shall be monitored.

- NOE.PHYSICAL
  Monitoring the physical assets of the TOE become considerably more difficult as the physical barriers must be broken first.

8.1.8 TT.REPLAY

Objectives covering this threat:

- O.ACCESS_HIST
  The objective will ease the detection of successful replayed authentication attempts. The objective itself does not perform detection of authentication replays; it only provides evidence for access. This evidence must be crosschecked with authorized user.

- O.AUDIT
  Auditing operations issued during a replay attack can help both in determining from where and by

© THALES Norway AS

whom the attack was launched. It may also to a certain degree help in reversing actions performed as part of the attack.

- O.CMD_LOG
  Command log can be used to detect replay of administrative commands. Once detected it is considerably easier to assign countermeasures for an attack.

- O.ID_AUTH
  Because users need to be identified and authorized to perform security relevant action within the TOE replaying information is made more difficult.

- O.REUSE
  Proper reuse of resources prevents attackers from being able to retrieve information that can be replayed.

- OE.NETWORK
  Proper network protection prevents attackers from being able to retrieve information that can be replayed.

- NOE.PHYSICAL
  Proper physical protection of the TOE prevents attackers from being able to retrieve information that can be replayed.

8.1.9 TT.UNATTENDED

Objectives covering this threat:

- O.AUTO_LOGOUT
  The objective provides means for minimizing the probability of someone finding an unattended session as the sessions are terminated automatically after some period of user inactivity.

- O.ID_AUTH
  Together with O.AUTO_LOGOUT this objective ensures that unattended sessions after some time can be made completely unavailable for unauthorized personnel.

8.1.10 TT.UNAUTH_ACCESS

Objectives covering this threat:

- O.ACCESS_HIST
  The access history can help determining that unauthorized access has occurred. It can furthermore be used to determine from where and by whom the unauthorized access was obtained.

- O.AUDIT
  The audit can help determining that unauthorized access has occurred. It can furthermore be used to determine from where and by whom the unauthorized access was obtained.

- O.CMD_ACL
  Command ACLs provide means for restricting each users access to perform administrative actions. Unauthorized access can be obtained by performing administrative actions that in turn provide access to the system. Restricting access to administrative tasks will therefore eliminate the possibility for

© THALES Norway AS

unauthorized administrators to gain access to the TOE via intentional malicious TOE administration. Roles provide default command ACLs.

- O.CMD_LOG
  The command log can help determining that unauthorized access has occurred. It can furthermore be used to determine from where and by whom the unauthorized access was obtained.

- O.DAC
  The objective is to perform DAC in order to avoid unauthorized access. DAC is performed for access to all database records. Initially this ensures that only the owner has access to records stored within the table. It is furthermore possible to add an ACL to the table definition so that access for other entities can be controlled.

- O.ID_AUTH
  Requiring users to identify themselves and provide valid authentication tokens before being allowed access to TOE assets prevents (in combination with O.DAC) unauthorized access to that information.

- O.LABELLING
  Information must be labelled so that it can be handled according to its classification. Improper handling of classified information may result in unauthorized access.

- O.LOCK
  Locking the user account after a given number of logon attempts reduces the chances of a successful brute force attack. The TOE will lock accounts after a succession of unsuccessful logon attempts.

- O.MAC
  The objective shall enforce the separation of data based on HCL, and NHC and SP. MAC is performed whenever access to subjects and object is requested.

- O.MAC_INTEGRITY
  In order to dynamically change clearance and sensitivity label, there must be functions to maintain this information. Dynamic change of clearance and sensitivity labels is necessary because the authenticated resources (e.g. users) change over time, and objects do not necessary have the same sensitivity label over time. This security objective covers the need for functions to maintain the integrity of data used in MAC so that unauthorized access based on old clearance/classification data is not given.

- O.MESSAGING
  To ensure unambiguous and correct information security labelling the TOE rejects all security marks that cannot be unambiguously converted into internal representation. Furthermore, the TOE rejects messages that have security marks outside the restrictions set by the system configuration, which in turn ensures that the system can only store information that it has clearance for.

- O.RECOVER
  The objective is to preserve a state in case of a system failure, so that XOmail can perform secure recovery and thereby avoid entering an insecure state allowing unauthorized access to TOE assets upon start-up.

- O.REF_MONITOR
  This objective ensures that unauthorized access cannot be achieved through bypassing the TSP. Untrusted subjects cannot interfere with the operation of the reference monitor, and they cannot bypass its checks.

© THALES Norway AS

- O.RESOURCE_SHARE
  The objective ensures secure resource sharing. This implies that resource sharing does not result in unauthorized access to information held in shared resources.

- O.ROLE_MNG
  The objective shall ensure that roles and role-belongings can be managed so that role assignments can be performed on an as-needed basis, and changed according to changing needs.

- O.ROLES
  The roles contain predefined access restrictions for administrative tasks, thereby providing a default set of available administrative commands. This reduces the chance of an attacker to successfully perform an unauthorized administrative task, thereby allowing himself/herself or a third person access to TOE assets.

- O.SELF_TEST
  Performing a database self-test ensures that corrupt ownership, ACLs and security parameters do not lead to unauthorized access to TOE assets.

- OE.AUDIT
  The audit can help determining that unauthorized access has occurred. It can furthermore be used to determine from where and by whom the unauthorized access was obtained.

- OE.NETWORK
  The network protection prevents external sources from accessing the network connected to the TOE. This in turn complicates the process of gaining unauthorized access to the TOE equipment and TOE's assets.

- NOE.PHYSICAL
  The physical protection prevents external sources from accessing the computer equipment hosting the TOE. This in turn complicates the process of gaining access to the TOE's assets.

8.1.11 TE.AUDIT_FAILURE

Objectives covering this threat:

- OE.AUDIT
  The objective is to prevent audit records from being lost or modified.

8.1.12 TE.DELIVERY

Objectives covering this threat:

- NOE.INSTALL
  Secure delivery routines can prevent attackers from compromising the TOE with viruses and other malicious software. Secure operation of the TOE cannot be obtained if viruses or other malicious software has been incorporated into the TOE.

© THALES Norway AS

8.1.13 TE.DOS

Objectives covering this threat:

- O.RECOVER
  The objective covers recovery from an abnormal termination due to denial of service attacks on the OS that the TOE runs on. It is ensured that the TOE does not enter an insecure state upon startup.

- OE.AUDIT
  The audit can be used to detect possible DOS attacks so that proper measures can be applied in an early stage of the attack.

- OE.NETWORK
  The objective reduces the possibility for DOS attacks met by the TOE host as it becomes more difficult to send data to the host. The network protection will to a certain degree prevent external sources from accessing the computer equipment hosting the TOE.

- NOE.PHYSICAL
  The objective reduces the possibility for DOS attacks met by the TOE environment as it becomes more difficult to send data resulting in TOE environment resource exhaustion. The physical protection will to a certain degree prevent unauthorized sources from accessing the computer equipment hosting the TOE, thereby complicating the DOS attack. Channels other that physical attack must be used.

8.1.14 TE.IMPROPER_INST

Objectives covering this threat:

- NOE.ADM_TRUST
  The administrator must be trusted to install the TOE correctly. The objective addresses the need for administrators that in a correct and secure manner performs proper installation of the TOE.

- NOE.INSTALL
  The objective seeks to eliminate improper installation and improper initial configuration of the TOE.

8.1.15 TE.POOR_DESIGN

Objectives covering this threat:

- OE.NETWORK
  By reducing the network access to the TOE, the chance for exploitation of possible design flaws is drastically reduced. The objective ensures that only trusted personnel have access to the TOE, thereby reducing the set of possible threat agents.

- OE.TRAF_SEPARATION
  Restricting access to the network traffic reduces the set of threat agents being able to exploit any potential flaws.

- NOE.PHYSICAL
  By reducing the physical access to the TOE, the chance for exploitation of possible design flaws is

© THALES Norway AS

drastically reduced. The objective ensures that only trusted personnel have access to the TOE, thereby reducing the set of possible threat agents.

8.1.16 TE.POOR_IMPL

Objectives covering this threat:

- O.RECOVER
  The objective covers recovery from an abnormal termination due to poor implementation of the TOE. It is ensured that the TOE does not enter an insecure state upon start-up.

- OE.NETWORK
  By reducing the network access to the TOE, the chance for exploitation of possible implementation flaws is drastically reduced. The objective ensures that only trusted personnel have access to the TOE, thereby reducing the set of possible threat agents.

- OE.TRAF_SEPARATION
  Restricting access to network traffic reduces the set of threat agents being able to exploit potential implementation flaws.

- NOE.PHYSICAL
  By reducing the network access to the TOE, the chance for exploitation of possible implementation flaws is drastically reduced. The objective ensures that only trusted personnel have access to the TOE, thereby reducing the set of possible threat agents.

8.1.17 TE.UNATTENDED

Objectives covering this threat:

- O.AUDIT
  The audit can help determining that access via unattended sessions has occurred. It can furthermore be used to determine from where and by whom the access was obtained.

- O.CMD_ACL
  Restricting access using the command ACLs may to a certain degree make this a less attractive way of gaining unauthorized access to the TOE. This is because the session may not necessarily provide the commands that was necessary for completing the attack.

- O.CMD_LOG
  The command log can help determining that access via unattended sessions has occurred. It can furthermore be used to determine from where and by whom the access was obtained.

- OE.AUDIT
  The audit can help determining that access via unattended sessions has occurred. It can furthermore be used to determine from where and by whom the access was obtained.

- NOE.PHYSICAL
  The objective addresses the need for the physical protection of the TOE environment to avoid exploitation of an unattended session.

© THALES Norway AS

8.1.18 A.ADM_TRAINING

Objectives covering this assumption:

- OE.ACCOUNTABLE
  All administrator shall be aware that they are accountable for their actions. Consequently the administrators must bear in mind his/her responsibilities at all time during administration of the TOE.

- NOE.INSTALL
  Secure installation, management and operation of the TOE require administrators to be properly trained.

8.1.19 A.AUDIT_REVIEW

Objectives covering this assumption:

- OE.ACCOUNTABLE
  Holding users of the TOE accountable for their actions means that the audit must be regularly, reviewed to detect inconsistencies or abnormal patterns and traces.

- NOE.INSTALL
  Administrators perform audit reviews.

8.1.20 A.CONFIDENCE

Objectives covering this assumption:

- NOE.ADM_TRUST
  The objective ensures trustworthy administrators which results in confidence that the system will not intentionally be misconfigured.

8.1.21 A.INVALIDATE

Objectives covering this assumption:

- NOE.ADM_TRUST
  Trust in administrative personnel is essential to ensure proper invalidation of authentication data.

- NOE.INSTALL
  Keeping user access updated is an integral part of the secure management of the TOE.

8.1.22 A.NETWORK

Objectives covering this assumption:

© THALES Norway AS

- OE.NETWORK
  The objective ensures that protection of the network that TOE use for communication, is pointed out as a responsibility of the TOE owners.

- OE.TRAF_SEPARATION
  The objective provides additional protection for TOE management traffic.

8.1.23 A.NOTIFY

Objectives covering this assumption:

- NOE.INSTALL
  Handling security issues are required for the secure management and operation of the TOE.

8.1.24 A.PHYSICAL

Objectives covering this assumption:

- NOE.PHYSICAL
  The TOE owners are responsible for implementing and maintaining sufficient physical protection for the system.

8.1.25 A.PHYSICAL_LOC

Objectives covering this assumption:

- NOE.PHYSICAL
  The TOE owners are responsible for restricting access to the areas where XOmail is used.

8.1.26 A.OS

Objectives covering this assumption:

- O.ID_AUTH
  The identification and authentication mechanisms of the TOE rely on basic mechanisms in the OS. Authentication tokens are only defined and protected by the OS, and the TOE shall not be able to maintain identities that do not exist in the OS.

- OE.AUDIT
  The objective describes use of OS audit mechanisms to store auditable events performed by the TOE.

- OE.ID_AUTH
  The objective identifies the need for user identification and authentication in the OS that the TOE runs on. The assumption A.OS covers this, as the OS need to CAPP-evaluated.

© THALES Norway AS

- OE.NETWORK
  The objective identifies the need for protection of network communication. The OS may provide lower layer protocol security mechanisms.

8.1.27 A.USR_TRAINING

Objectives covering this assumption:

- OE.ACCOUNTABLE
  All users shall be aware that they are accountable for their actions. Consequently the users must bear in mind their responsibilities at all time during administration of the TOE.

- NOE.INSTALL
  The secure management and operation of the TOE requires users to be properly trained.

8.1.28 P.ACCOUNTING

The objective of the policy for accounting is to provide sufficient information to be able to investigate a deliberate or accidental compromise of accountable information and assess the damage arising from the compromise. This calls for a unique identification of the users (O.ID_AUTH, OE.ACCOUNTABLE) and logging of sensitive events (O.ACCESS_HIST, O.AUDIT, O.CMD_LOG, OE.AUDIT). The user identification will appear in the log records.

8.1.29 P.CLASSIFICATION

The classification of information is done by the originator (O.MAC).

8.1.30 P.CLEAR

The CLEAR procedures allows exceptions in the mandatory access control mechanisms (O.MAC) for authorised users to send classified messages in clear on unsecure lines.

8.1.31 P.DAC

The TOE ensures Discretionary Access Control (O.DAC) by controlling access to resources based on the identity of users and groups of users.

8.1.32 P.INTEGRITY

The TOE shall be able to ensure integrity of message data (O.DIG_SIGN, OE.NETWORK).

© THALES Norway AS

8.1.33 P.INTERFACE_CONTROL

The TOE, the host computer and computer network must be installed and configured in accordance with the policy for the system (NOE.INSTALL).

8.1.34 P.MAC

The TOE ensures Mandatory Access Control (O.MAC) based on user clearances and object security classifications.

The TOE allows authorized security administrators (O.ROLES) to specify the security clearance of users and resources (O.MAC_INTEGRITY).

8.1.35 P.MARKING

The TOE ensures that information is labelled with the correct human-readable label (O.LABELLING).

8.1.36 P.PROTECTION

Those responsible for the TOE will ensure that the TOE is installed, managed and operated in a manner that maintains security (NOE.INSTALL) and that network communication to/from the TOE is protected (OE.NETWORK).

8.2 Security requirements rationale

8.2.1 Requirements are appropriate

The following two tables (Table 8-4 and Table 8-5) show that requirements are appropriate to cover TOE security objectives and TOE environment security objectives.

Objective / Req.	O.ACCESS_HIST	O.AUDIT	O.AUTO_LOGOUT	O.CMD_ACL	O.CMD_LOG	O.DAC	O.ID_AUTH	O.LABELLING	O.LOCK	O.MAC	O.MAC_INTEGRITY	O.MANAGE	O.MESSAGING	O.RECOVER	O.REF_MONITOR	O.RESOURCE_SHAR	O.REUSE	O.ROLE_MNG	O.ROLES	O.SELF_TEST
FAU_ARP.1									X					X						
FAU_GEN.1(1)	X	X																		
FAU_GEN.2(1)	X	X																		
FAU_SAA.1		X																		
FAU_SAR.1(1)		X																		

Req. \ Objective	O.ACCESS_HIST	O.AUDIT	O.AUTO_LOGOUT	O.CMD_ACL	O.CMD_LOG	O.DAC	O.ID_AUTH	O.LABELLING	O.LOCK	O.MAC	O.MAC_INTEGRITY	O.MANAGE	O.MESSAGING	O.RECOVER	O.REF_MONITOR	O.RESOURCE_SHAR	O.REUSE	O.ROLE_MNG	O.ROLES	O.SELF_TEST
FAU_SAR.2(1)		X																		
FAU_STG.1(1)		X																		
FAU_STG.3(1)		X																		
FAU_STG.4(1)		X																		
FDP_ACC.1						X							X							
FDP_ACF.1				X		X							X							
FDP_ETC.2						X		X		X										
FDP_IFC.2										X			X							
FDP_IFF.2										X			X							
FDP_ITC.2						X				X										
FDP_RIP.2																	X			
FIA_AFL.1									X											
FIA_ATD.1						X				X									X	
FIA_UAU.2				X	X		X													
FIA_UAU.5							X													
FIA_UID.2(1)				X	X		X													
FIA_USB.1			X	X	X															
FMT_MSA.1				X							X	X								
FMT_MSA.3											X									
FMT_MTD.1												X								
FMT_SMF.1											X	X								
FMT_SMR.1																		X	X	
FPT_FLS.1														X						
FPT_RCV.1		X																		
FPT_RCV.2														X						X
FPT_RCV.4														X						
FPT_RVM.1															X					
FPT_SEP.3															X	X				
FPT_TDC.1										X			X							
FPT_TST.1																				X
FTA_SSL.3			X																	
FTA_TSE.1(1)						X			X											

Table 8-4: Security objectives satisfaction

© THALES Norway AS

Objective / Req	OE.ACCOUNTABLE	OE.AUDIT	OE.ID_AUTH	OE.NETWORK	OE.TRAF_SEPARATION	NOE.ADM_TRUST	NOE.INSTALL	NOE.PHYSICAL
FAU_GEN.1(2)		X						
FAU_GEN.2(2)	X	X						
FAU_SAR.1(2)	X	X						
FAU_SAR.2(2)		X						
FAU_STG.1(2)	X	X						
FAU_STG.3(2)		X						
FIA_UAU.2			X					
FIA_UID.2(2)	X	X						
FDP_ITT.1				X				
FPT_ITT.1				X				
FPT_STM.1		X						
FTA_TSE.1(2)					X			
ADO_DEL.2							X	
ADO_IGS.1						X	X	X
AGD_ADM.1						X	X	
AGD_USR.1							X	

Table 8-5: Environment security objectives satisfaction

8.2.1.1 O.ACCESS_HIST

Requirements covering this objective:

- FAU_GEN.1(1)
  The TOE is required to maintain an access history list i.e. a list of successful and unsuccessful session establishment attempts. Authorized administrators may access this list.

- FAU_GEN.2(1)
  The TOE associates user identities for the events.

  Note: If the username is unknown to the system, it should not be logged in order to prevent unintentional logging of user password.

8.2.1.2 O.AUDIT

Requirements covering this objective:

- FAU_GEN.1(1)
  The TOE is responsible for recording audit data or initiating OS audit system calls when auditable events occur within the TSC.

© THALES Norway AS

- FAU_GEN.2(1)
  The TOE provides user identities for events. The user identity for each event conforms to the owner of the software process that caused the event. When TOE audit mechanisms are initiated, the user identity must be specified by the TOE.

- FAU_SAA.1
  The audit records are required by the TSF self-monitoring introduced by this requirement.

- FAU_SAR.1(1)
  The TOE audit must be possible to read and interpret for the authorized TOE administrators.

- FAU_SAR.2(1)
  It must be possible to restrict access to the audit so that only authorized TOE administrators are allowed access.

- FAU_STG.1(1)
  The audit must be protected against unauthorized deletion or modification.

- FAU_STG.3(1)
  The TOE issues alarms to ensure that administrators are warned before the audit storage is exhausted.

- FAU_STG.4(1)
  The TOE implements means to ensure that auditing is always available as long as the system is running.

- FPT_RCV.1
  If the audit trail storage is exhausted, the TOE will shut down to preserve a secure state.

8.2.1.3 O.AUTO_LOGOUT

Requirements covering this objective:

- FTA_SSL.3.
  The TOE is responsible for providing mechanisms that are capable of automatically terminate sessions after a configurable period of user inactivity.

- FIA_USB.1
  The TOE is responsible for automatic logout of the user when the user clearance or user command access has been changed.

8.2.1.4 O.CMD_ACL

Requirements covering this objective:

- FDP_ACF.1.
  For the TOE to be able to restrict access to commands based on user identity, the DAC mechanisms must be implemented.

© THALES Norway AS

- FIA_UAU.2.
  For the TOE to be able to restrict access to administrative commands, it must require all users to identify and authenticate themselves before access to administrative functions can be given. The FIA_UAU.2 ensures that authentication is performed before users are given access to administrative commands.

- FIA_UID.2(1).
  For the TOE to be able to enforce the ACL for administrative commands, each administrator must be identified before performing any other action. The identification is used to determine whether the user is allowed to perform the command.

- FIA_USB.1
  Command access restrictions rely heavily on the TOE's ability to associate user identity with subjects acting on behalf of users. Accordingly, FIA_USB.1 is necessary for correct command access restriction functionality.

- FMT_MSA.1
  The requirement restricts access to the management of command ACLs. Only security administrators are allowed to change the command ACLs of user templates.

8.2.1.5 O.CMD_LOG

Requirements covering this objective:

- FIA_UAU.2.
  In order to perform recording of administrative commands and associate user identification with each of the records, all users must identify and authenticate. The FIA_UAU.2 ensures authentication of all users before administrative commands can be performed.

- FIA_UID.2(1).
  Each entry in the command log must be associated with the user that caused the command. Therefore it is necessary for administrators to identify themselves before any other action is performed.

- FIA_USB.1.
  Command log functionality relies heavily on the TOE's ability to associate user identity with subjects acting on behalf of users. Accordingly, FIA_USB.1 is necessary for correct command log functionality.

8.2.1.6 O.DAC

Requirements covering this objective:

- FDP_ACC.1.
  Requires DAC to be performed on database objects and database subjects.

- FDP_ACF.1.
  Specifies how DAC shall be applied on database objects and database subjects.

- FDP_ETC.2.
  Requires the TOE to perform DAC during export to outside the TSC.

© THALES Norway AS

- FDP_ITC.2.
  Requires the TOE to perform DAC during import from outside the TSC.

- FIA_ATD.1.
  Requires security attributes to be maintained for each individual user. Some of the security attributes are necessary for DAC operation.

- FTA_TSE.1(1)
  Requires the TOE to perform DAC based on the client host address, and authentication tokens.

8.2.1.7 O.ID_AUTH

Requirements covering this objective:

- FIA_UAU.2.
  Ensures that unauthorized users are not given access to the TOE's assets.

- FIA_UAU.5.
  Specifies authentication methods that shall be present in the TOE.

- FIA_UID.2(1).
  FIA_UID.2(1) allows the user to receive an error message upon failed identification before being successfully identified. The error message reveals no assets, nor will it give assistance in finding a correct identification.

8.2.1.8 O.LABELLING

Requirements covering this objective:

- FDP_ETC.2.
  Upon export outside TSC information is labelled with a human-readable label representation of internal information label.

8.2.1.9 O.LOCK

Requirements covering this objective:

- FAU_ARP.1
  The requirement implements automatic lockout of users upon selected potential security violations.

- FIA_AFL.1.
  Describes the conditions for automatic locking of user accounts (setting of the lock-attribute).

- FTA_TSE.1(1).
  The TOE is required to be able to deny users access based on a lock-attribute.

© THALES Norway AS

8.2.1.10 O.MAC

Requirements covering this objective:

- FDP_ETC.2.
  Requires the TOE to perform MAC during export to outside the TSC.

- FDP_IFC.2.
  Requires MAC to be performed on all non-trusted subjects and all information.

- FDP_IFF.2.
  Describes how MAC and MAC support-functions shall be realized. The TOE is required to ensure that access to resources is given based on clear rules for clearance and label comparison.

- FDP_ITC.2.
  Requires the TOE to perform MAC during import from outside the TSC.

- FIA_ATD.1.
  Requires security attributes to be maintained for each individual user. Some of the security attributes are necessary for MAC operation.

- FPT_TDC.1.
  The TOE's ability to perform MAC correctly relies on its ability to assign object label and subject clearance upon importing data from other trusted IT products. FPT_TDC.1 addresses the requirements that ensure TSF data consistency.

8.2.1.11 O.MAC_INTEGRITY

Requirements covering this objective:

- FMT_MSA.1.
  The TOE is required to enforce MAC and DAC to restrict the ability to read or write object and subject security attributes. This in turn ensures that unauthorized personnel cannot violate the integrity of the MAC security attributes.

- FMT_MSA.3.
  The TOE is required to provide default values for security attributes and additionally let *Security Administrators* override the default settings. This allows *Security Administrator* to maintain integrity of MAC security attributes.

- FMT_SMF.1.
  Management functions that ensure integrity of MAC security attributes must be well defined.

8.2.1.12 O.MANAGE

Requirements covering this objective:

- FMT_MSA.1.
  Enforcing MAC and DAC to restrict ability to modify security attributes support secure and error free management.

© THALES Norway AS

- FMT_MTD.1.
  Being able to allow only administrators to read or write TOE configuration data minimises the effort necessary for TOE owners to accomplish secure and effective management of the TOE.

- FMT_SMF.1.
  Management functions that ensure secure, efficient and accurate management of the TOE must be well defined.

8.2.1.13 O.MESSAGING

Requirements covering this objective:

- FDP_ACC.1.
  Enforcing DAC ensures that message can be created, viewed, modified or deleted only by those explicitly granted access to the TOE.

- FDP_ACF.1.
  The DAC mechanisms must be implemented as specified in this requirement in order to have it applied to the message handling.

- FDP_IFC.2.
  Enforcing MAC ensures that message can be created, viewed, modified or deleted only by those explicitly granted clearance for that type of information.

- FDP_IFF.2.
  The MAC mechanisms must be implemented as specified in this requirement in order to have it applied to the message handling.

- FPT_TDC.1.
  The requirements in FPT_TDC.1 ensure that the TOE must reject all messages that do not have a valid security mark that can be verified against internal rules.

8.2.1.14 O.RECOVER

Requirements covering this objective:

- FAU_ARP.1
  The TOE performs automated recovery actions upon detection of potential security violations.

- FPT_FLS.1.
  The TOE is required to preserve a secure state in the case of any failure. The preserved secure state can later be used to recover from the failure.

- FPT_RCV.2
  This requirement ensures that automated recovery is performed, and that the TOE does not start in an insecure state.

- FPT_RCV.4
  This requirement ensures that TSF shall either succeed and enter a new secure state, or fail and return to another secure state.

© THALES Norway AS

8.2.1.15 O.REF_MONITOR

Requirements covering this objective:

- FPT_RVM.1.
  The TOE is required to ensure that untrusted subjects cannot bypass the checks of the reference monitor.

- FPT_SEP.3.
  TOE must ensure that untrusted subjects cannot interfere with the operation of the reference monitor.

8.2.1.16 O.RESOURCE_SHARE

Requirements covering this objective:

- FPT_SEP.3.
  Separation of security domains implies that upon allocation of resources all previously stored information must be unavailable. This enforces secure resource sharing.

8.2.1.17 O.REUSE

Requirements covering this objective:

- FDP_RIP.2
  Ensures that all informational content of a resource is unrecoverable upon reuse of that resource.

8.2.1.18 O.ROLE_MNG

Requirements covering this objective:

- FMT_SMR.1
  Defines requirements to the TOE on how the roles shall be managed.

8.2.1.19 O.ROLES

Requirements covering this objective:

- FIA_ATD.1
  Requires the TOE to use roles.

- FMT_SMR.1
  Defines the roles that the TOE shall maintain and associate users with.

© THALES Norway AS

8.2.1.20 O.SELF_TEST

Requirements covering this objective:

- FPT_RCV.2
  The TOE is required to evaluate the need for, and if needed perform an automated recovery upon start-up.

- FPT_TST.1.
  The TOE is required to run a database integrity check during start-up.

8.2.1.21 OE.ACCOUNTABLE

Requirements covering this objective:

- FAU_GEN.2(2)
  To be able to hold users accountable for their actions, there must be a user identity associated with all audited events caused by a user.

- FAU_SAR.1(2)
  It shall be possible to perform review of the audit.

- FAU_STG.1(2)
  The audit records must be protected against unauthorized deletion or modification so that users can be held accountable for audited actions performed within the TOE.

- FIA_UID.2(2)
  The user identifier is used to attach events to users, thereby hold the users accountable for their actions.

8.2.1.22 OE.AUDIT

Requirements covering this objective:

- FAU_GEN.1(2)
  When the TOE initiates OS auditing mechanisms, the OS is responsible for correct auditing. This implies that when OS auditing mechanisms are used, the TOE must supply the OS with all necessary information (subject identity, date, time, type of event, and outcome of event).

- FAU_GEN.2(2)
  The OS must be able to store the associated identity of the user that caused the event.

- FAU_SAR.1(2)
  The OS audit must be possible to read and interpret for the authorized TOE administrators.

- FAU_SAR.2(2)
  It must be possible to restrict access to the OS audit so that only authorized TOE administrators are allowed access.

© THALES Norway AS

- FAU_STG.1(2)
  The OS audit must be protected against unauthorized deletion or modification.

- FAU_STG.3(2)
  The OS implements means to ensure that auditing is always available as long as the system is running.

- FIA_UID.2(2)
  The OS ensures that correct identifier is associated with the audit event.

- FPT_STM.1
  The OS ensures that correct timestamp is associated with the audit event.

8.2.1.23 OE.ID_AUTH

Requirements covering this objective:

- FIA_UAU.2
  Ensures that unauthorized users are not given access to the TOE's assets.

8.2.1.24 OE.NETWORK

Requirements covering this objective:

- FDP_ITT.1
  Protection of data transmitted between separate parts of the TOE is ensured by physical protection or encryption mechanisms.

- FPT_ITT.1
  Integrity and confidentiality of data transmitted between separate parts of the TOE is ensured by physical protection or encryption mechanisms.

8.2.1.25 OE.TRAF_SEPARATION

- FTA_TSE.1(2)
  Management access is only allowed from selected networks.

8.2.1.26 NOE.ADM_TRUST

The TOE shall be managed as described by the guidance documentation (ADO_IGS.1, AGD_ADM.1).

© THALES Norway AS

8.2.1.27 NOE.INSTALL

The TOE shall be securely installed, maintained and operated in accordance with the guidance documentation (ADO_DEL.1, ADO_IGS.1, AGD_ADM.1, AGD_USR.1).

8.2.1.28 NOE.PHYSICAL

Those responsible for the TOE shall ensure sufficient physical protection of the TOE and the TOE environment (ADO_IGS.1).

8.2.2 Functional security requirements dependencies

The table shows each component's direct dependencies to other components. This demonstrates that the set of security requirements form a mutually supportive and consistent whole.

TOE Requirement	Dependency	Included
FAU_ARP.1	FAU_SAA.1	Yes
FAU_GEN.1(1)	FPT_STM.1	Yes
FAU_GEN.2(1)	FAU_GEN.1, FIA_UID.1 (via FIA_UID.2(1))	Yes
FAU_SAA.1	FAU_GEN.1(1)	Yes
FAU_SAR.1(1)	FAU_GEN.1(1)	Yes
FAU_SAR.2(1)	FAU_SAR.1(1)	Yes
FAU_STG.1(1)	FAU_GEN.1(1)	Yes
FAU_STG.3(1)	FAU_STG.1(1)	Yes
FAU_STG.4(1)	FAU_STG.1(1)	Yes
FDP_ACC.1	FDP_ACF.1	Yes
FDP_ACF.1.	FDP_ACC.1, FMT_MSA.3	Yes
FDP_ETC.2	FDP_ACC.1, FDP_IFC.1	Yes
FDP_IFC.2	FDP_IFF.1	Yes
FDP_IFF.2	FDP_IFC.1, FMT_MSA.3	Yes
FDP_ITC.2	FDP_ACC.1, FDP_IFC.1, FPT_TDC.1, (*FTP_ITC.1* or *FTP_TRP.1*)	No
FDP_RIP.2		N/A
FIA_AFL.1	FIA_UAU.2	Yes
FIA_ATD.1		N/A
FIA_UAU.2	FIA_UID.1 (via FIA_UID.2(1))	Yes
FIA_UAU.5		N/A
FIA_UID.2(1)		N/A
FIA_USB.1	FIA_ATD.1	Yes
FMT_MSA.1	FDP_ACC.1, FDP_IFC.1, FMT_SMF.1, FMT_SMR.1	Yes
FMT_MSA.3	FMT_MSA.1, FMT_SMR.1	Yes
FMT_MTD.1	FMT_SMF.1, FMT_SMR.1	Yes
FMT_SMF.1		Yes
FMT_SMR.1	FIA_UID.1 (via FIA_UID.2(1))	Yes
FPT_FLS.1	ADV_SPM.1	Yes
FPT_RCV.1	AGD_ADM.1, ADV_SPM.1	Yes
FPT_RCV.2	AGD_ADM.1, ADV_SPM.1	Yes

© THALES Norway AS

TOE Requirement	Dependency	Included
FPT_RCV.4	ADV_SPM.1	Yes
FPT_RVM.1		N/A
FPT_SEP.3		N/A
FPT_TDC.1		N/A
FPT_TST.1	(*FPT_AMT.1*)	No
FTA_SSL.3		N/A
FTA_TSE.1(1)		N/A
IT Environment Requirement	**Dependency**	**Included**
FAU_GEN.1(2)	FPT_STM.1	Yes
FAU_GEN.2(2)	FAU_GEN.1, FIA_UID.1 (via FIA_UID.2(2))	Yes
FAU_SAR.1(2)	FAU_GEN.1(2)	Yes
FAU_SAR.2(2)	FAU_SAR.1(2)	Yes
FAU_STG.1(2)	FAU_GEN.1(2)	Yes
FAU_STG.3(2)	FAU_STG.1(2)	Yes
FIA_UAU.2	FIA_UID.1 (via FIA_UID.2(2))	Yes
FIA_UID.2(2)		N/A
FDP_ITT.1	FDP_ACC.1 or FDP_IFC.1	Yes
FPT_ITT.1		N/A
FPT_STM.1		N/A
FTA_TSE.1(2)		N/A

Table 8-6: Functional requirements dependency check

Dependencies are met with three exceptions:

FDP_ITC.2 specifies that either FTP_ITC.1 or FTP_TRP.1 must be present. Given that only OS-authenticated users can perform import, the physical protection of the TOE environment, and the integrity protection of the network, none of these requirements are considered necessary for secure operation.

FPT_TST.1 specifies that FPT_AMT.1 must be present. This dependency is not considered necessary to fulfil as the OS provides an abstract machine that can be used. The OS performs abstract machine testing.

8.2.3 Security assurance requirements dependencies

The table shows each component's direct dependencies to other components. This demonstrates that the set of security assurance requirements form a mutually supportive and consistent whole.

TOE Requirement	Dependency	Included
ACM_AUT.1	ACM_CAP.3	Yes
ACM.CAP.4	ALC_DVS.1	Yes
ACM_SCP.2	ACM_CAP.3	Yes
ADO_DEL.2	ACM_CAP.3	Yes
ADO_IGS.1	AGD_ADM.1	Yes
ADV_FSP.2	ADV_RCR.1	Yes
ADV_HLD.2	ADV_FSP.1, ADV_RCR.1	Yes

© THALES Norway AS

TOE Requirement	Dependency	Included
ADV_IMP.1	ADV_LLD.1, ADV_RCR.1, ALC_TAT.1	Yes
ADV_LLD.1	ADV_HLD.2, ADV_RCR.1	Yes
ADV_RCR.1		N/A
ADV_SPM.1	ADV_FSP.1	Yes
AGD_ADM.1	ADV_FSP.1	Yes
AGD_USR.1	ADV_FSP.1	Yes
ALC_DVS.1		N/A
ALC_LCD.1		N/A
ALC_TAT.1	ADV_IMP.1	Yes
ATE_COV.2	ADV_FSP.1, ATE_FUN.1	Yes
ATE_DPT.1	ADV_HLD.1, ATE_FUN.1	Yes
ATE_FUN.1		N/A
ATE_IND.2	ADV_FSP.1, ADG_ADM.1, ADG_USR.1, ATE_FUN.1	Yes
AVA_MSU.2	ADO_IGS.1, ADV_FSP.1, AGD_ADM.1, AGD_USR.1	Yes
AVA_SOF.1	ADV_FSP.1, ADV_HLD.1	Yes
AVA_VLA.2	ADV_FSP.1, ADV_HLD.2, ADV_IMP.1, ADV_LLD.1, AGD_ADM.1, AGD_USR.1	Yes

Table 8-7: Assurance requirements dependency check

8.3 TOE summary specification rationale

8.3.1 TOE security functional requirements satisfaction

Table 8-8 shows that the security functions are appropriate to cover the security requirements.

The following sections show that functional requirements are covered by identified security functions. The complete coverage of assurance requirements by assurance measures is shown in Table 6-1.

© THALES Norway AS

Security function / Req.	SF.AUDIT	SF.AUTHENTICATION	SF.AUTO_LOGOUT	SF.CLASSIFICATION_TAG	SF.CLEAR	SF.COMMAND_ACCESS	SF.COMMUNICATION_SECURIT	SF.DAC	SF.DB_SELF_TEST	SF.EXECUTION_DOMAIN	SF.LABELLING	SF.LABEL_TRANSFORM	SF.LOCK	SF.MAC	SF.ROLES	SF.SECURE_STATE_RECOVER	SF.SUBNET_RESTRICTION	SF.VALIDATE
FAU_ARP.1																X		
FAU_GEN.1(1)	X																	
FAU_GEN.2(1)	X																	
FAU_SAA.1	X																	
FAU_SAR.1(1)	X							X						X				
FAU_SAR.2(1)	X							X						X				
FAU_STG.1(1)	X							X						X				
FAU_STG.3(1)	X																	
FAU_STG.4(1)	X																	
FDP_ACC.1								X										
FDP_ACF.1								X										
FDP_ETC.2								X				X		X				
FDP_IFC.2														X				
FDP_IFF.2					X									X				
FDP_ITC.2								X			X	X		X				
FDP_RIP.2										X								
FIA_AFL.1													X					
FIA_ATD.1		X																
FIA_UAU.2		X																
FIA_UAU.5		X																
FIA_UID.2(1)		X																
FIA_USB.1								X						X				
FMT_MSA.1						X		X						X				
FMT_MSA.3								X						X				
FMT_MTD.1															X			
FMT_SMF.1	X																	
FMT_SMR.1															X			
FPT_FLS.1																X		
FPT_RCV.1	X								X									
FPT_RCV.2									X									
FPT_RCV.4																X		
FPT_RVM.1										X								
FPT_SEP.3										X								
FPT_TDC.1				X			X					X						
FPT_TST.1									X									X
FTA_SSL.3			X															
FTA_TSE.1(1)							X						X				X	

Table 8-8: Functional requirements satisfaction

© THALES Norway AS

8.3.1.1 FAU_ARP.1

This requirement is enforced by the following security functions:

- SF.SECURE_STATE_RECOVERY
  The TOE performs recovery actions upon detection of potential security violations.

8.3.1.2 FAU_GEN.1(1)

This requirement is enforced by the following security functions:

- SF.AUDIT
  The TOE audit trail satisfies the FAU_GEN.1(1) list of auditable events.

8.3.1.3 FAU_GEN.2(1)

This requirement is enforced by the following security functions:

- SF.AUDIT
  The TOE is required to associate a user identity with all audit records where applicable.

8.3.1.4 FAU_SAA.1

This requirement is enforced by the following security functions:

- SF.AUDIT
  The TOE can be configured for each alarm type to raise an alarm when successive alarm events occur in a configurable period.

8.3.1.5 FAU_SAR.1(1)

This requirement is enforced by the following security functions:

- SF.AUDIT
  The TOE allows the audit information to be read by authorized and sufficiently cleared administrators.

- SF.DAC
  The audit logs stored in the XOmail database is protected by DAC. This will enable authorised users to read audit records.

- SF.MAC
  The audit logs stored in the XOmail database is protected by MAC. This will enable authorised users to read audit records for which the user has proper clearance.

© THALES Norway AS

8.3.1.6 FAU_SAR.2(1)

This requirement is enforced by the following security functions:

- SF.AUDIT
  The TOE restricts audit information to authorized administrators.

- SF.DAC
  The audit logs stored in the XOmail database is protected by DAC. This will ensure that only authorised users get read access to audit records.

- SF.MAC
  The audit logs stored in the XOmail database is protected by MAC. This will ensure that users only get read access according to the users clearance and the audit records security classification.

8.3.1.7 FAU_STG.1(1)

This requirement is enforced by the following security functions:

- SF.AUDIT
  The TOE protects audit records from modification and unauthorized deletion.

- SF.DAC
  The audit logs stored in the XOmail database is protected by DAC. This will ensure that only authorised users get delete access to audit records.

- SF.MAC
  The audit logs stored in the XOmail database is protected by MAC. This will ensure that users only get delete access according to the users clearance and the audit records security classification.

8.3.1.8 FAU_STG.3(1)

This requirement is enforced by the following security functions:

- SF.AUDIT
  The TOE will issue alarms to warn the system administrators when the audit storage nears exhaustion.

8.3.1.9 FAU_STG.4(1)

This requirement is enforced by the following security functions:

- SF.AUDIT
  The TOE will suspend the operation when audit trail is full i.e. the hard disk is full.

8.3.1.10 FDP_ACC.1

This requirement is enforced by the following security functions:

© THALES Norway AS

- SF.DAC
  The TOE is required to apply DAC within the TSC.

8.3.1.11 FDP_ACF.1

This requirement is enforced by the following security functions:

- SF.DAC
  The requirement describes how DAC shall be applied.

8.3.1.12 FDP_ETC.2

This requirement is enforced by the following security functions:

- SF.DAC
  The TOE is required to apply DAC during export of information from the TSC.

- SF.LABEL_TRANSFORM
  The TOE is required to convert internal label representation to a representation that unambiguously can be associated with the exported data. The exported representation is defined by the export medium.

- SF.MAC
  The TOE is required to apply MAC during export of information from the TSC.

8.3.1.13 FDP_IFC.2

This requirement is enforced by the following security functions:

- SF.MAC
  The TOE is required to apply MAC within the TSC.

8.3.1.14 FDP_IFF.2

This requirement is enforced by the following security functions:

- SF.CLEAR
  Requires the TOE to allow CLEAR-marking of information. The requirements of FDP_IFF.2 specify how to handle CLEAR-marked information.

- SF.MAC
  The requirements specify how MAC shall be applied.

© THALES Norway AS

8.3.1.15 FDP_ITC.2

This requirement is enforced by the following security functions:

- SF.DAC
  The TOE is required to apply DAC during import of information into the TSC.

- SF.LABELLING
  The TOE is required to assign a label during import of information into the TSC if correct label cannot be determined. This security function covers cases where either label is not readable, or where label does not exist.

- SF.LABEL_TRANSFORM
  The TOE is required to ensure that interpretation of security label information is as intended by the source of the user data. This covers potentially converting the label information into the internal representation.

- SF.MAC
  The TOE is required to apply MAC during import of information into the TSC.

8.3.1.16 FDP_RIP.2

This requirement is enforced by the following security functions:

- SF.EXECUTION_DOMAINS.
  Different execution domains may over time reuse system resources. It is therefore necessary that information is made unavailable upon allocation of all new system resources. FDP_RIP.2 specifies such requirements for the TOE.

8.3.1.17 FIA_AFL.1

This requirement is enforced by the following security functions:

- SF.LOCK.
  Requirements for account locking are specified in the FIA_AFL.1.

8.3.1.18 FIA_ATD.1

This requirement is enforced by the following security functions:

- SF.AUTHENTICATION.
  The authentication mechanism shall assign security attributes to the user's subjects. FIA_ATD.1 specifies the security attributes available to the authentication mechanism.

8.3.1.19 FIA_UAU.2

This requirement is enforced by the following security functions:

© THALES Norway AS

- SF.AUTHENTICATION.
  Users are required to be authenticated.

8.3.1.20 FIA_UAU.5

This requirement is enforced by the following security functions:

- SF.AUTHENTICATION
  FIA_UAU.5 specifies how the authentication mechanisms shall work.

8.3.1.21 FIA_UID.2(1)

This requirement is enforced by the following security functions:

- SF.AUTHENTICATION
  FIA_UID.1 specifies when the TOE shall require the users to identify themselves.

8.3.1.22 FIA_USB.1

This requirement is enforced by the following security functions:

- SF.DAC
  For the TOE to be able to perform DAC, it is required that user identity of subjects acting on behalf of users have the correct user security attributes associated. This is part of the DAC functionality; user identity must be associated with all subjects acting on behalf of a user.

- SF.MAC
  Requirements defined in FIA_USB.1 ensure that the necessary security attributes for MAC functionality are associated with the subjects acting on behalf of the users.

8.3.1.23 FMT_MSA.1

This requirement is enforced by the following security functions:

- SF.DAC
  The TOE is required to apply DAC on the management functions for security attributes.

- SF.MAC
  The TOE is required to apply MAC on the management functions for security attributes.

- SF.COMMAND_ACCESS
  The SF provides management of administrators' command access. Command access is restricted by an administrator access level, as well as access to individual commands. Only Security Administrators are allowed to manage command ACLs.

© THALES Norway AS

8.3.1.24 FMT_MSA.3

This requirement is enforced by the following security functions:

- SF.DAC
  FMT_MSA.3 specifies how new objects and subjects shall be initialized with regards to security attributes relevant for DAC functionality.

- SF.MAC
  FMT_MSA.3 specifies how new objects and subjects shall be initialized with regards to security attributes relevant for MAC functionality.

8.3.1.25 FMT_MTD.1

This requirement is enforced by the following security functions:

- SF.ROLES
  The requirements describe how role belonging shall put restrictions on access to administrative tasks.

8.3.1.26 FMT_SMF.1

The requirement enforces the following security functions:

- SF.AUDIT
  The requirement defines the management functions for which execution must be logged.

8.3.1.27 FMT_SMR.1

This requirement is enforced by the following security functions:

- SF.ROLES
  Requirements defining the available roles.

8.3.1.28 FPT_FLS.1

This requirement is enforced by the following security functions:

- SF.SECURE_STATE_PRESERVATION
  The TOE is required to preserve a secure state in the occurrence of a failure.

8.3.1.29 FPT_RCV.1

- SF.AUDIT
  The security function ensures the TOE is shut down safely then the audit trail is full.

© THALES Norway AS

- SF.SECURE_STATE_RECOVERY
  The security function ensures the system can be restarted when sufficient storage space has been made available for the audit trail.

8.3.1.30 FPT_RCV.2

This requirement is enforced by the following security functions:

- SF.DB_SELF_TEST
  This security function covers the automatic and manual recovery functions required by FPT_RCV.2.

8.3.1.31 FPT_RCV.4

This requirement is enforced by the following security functions:

- SF.SECURE_STATE_RECOVERY
  This security function ensures that all SFs ends in a new secure state as required by FPT_RCV.4.

8.3.1.32 FPT_RVM.1

This requirement is enforced by the following security functions:

- SF.EXECUTION_DOMAINS
  This security function implements the reference monitor that is required in FPT_RVM.1

8.3.1.33 FPT_SEP.3

This requirement is enforced by the following security functions:

- SF.EXECUTION_DOMAINS.
  The TOE needs to implement separation of the execution domains. Requirements to the separation are specified in FPT_SEP.3.

8.3.1.34 FPT_TDC.1

This requirement is enforced by the following security functions:

- SF.CLASSIFICATION_TAG.
  If configured as a Mail Guard, the TOE is required to check the classification tag upon reception of an incoming message.

- SF.COMMUNICATION_SECURITY.
  During sending and reception of messages, the TOE must handle security attributes according to the requirements specified in the FPT_TDC.1.

© THALES Norway AS

- SF.LABEL_TRANSFORM.
  FPT_TDC.1 address requirements concerning the unambiguous representation of security labels and the possibility to convert to/from internal representation.

8.3.1.35 FPT_TST.1

This requirement is enforced by the following security functions:

- SF.DB_SELF_TEST.
  The requirements of FPT_TST.1 specify the features of the database self test.

- SF.VALIDATE.
  The TOE is required to be able to perform validation of the TSF data and executable code.

8.3.1.36 FTA_SSL.3

This requirement is enforced by the following security functions:

- SF.AUTO_LOGOUT
  The TOE is required to provide means for session termination after a period of user inactivity.

8.3.1.37 FTA_TSE.1(1)

This requirement is enforced by the following security functions:

- SF.COMMUNICATION_SECURITY
  Communications security is enforced with the requirements for the TOE to be able to restrict access based on attributes defined in FPT_TSE.1.

- SF.LOCK
  FTA_TSE.1 requires the TOE to be able to deny session establishment based on the lock attribute.

- SF.SUBNET_RESTRICTION
  FTA_TSE.1 requires the TOE to be able to deny session establishment based on IP address, subnet address or hostname of the client initiating the connection.

8.4 PP rationale

Not applicable.

© THALES Norway AS

© THALES Norway AS