Protection Profile för eHandelsplats

Tomas Johansson
tomjo511@student.liu.se

Martin Öberg
marob265@student.liu.se

Abstract

This report presents a Protection Profile for a eMarketplace and summarizes the lessons learned from constructing the document and studying the Common Criteria standard.

Not all sections of the PP is complete, because the goal was to learn about as many different parts of the PP construction process as possible.

One reflection is that the PP creation process is not for everyone. It requires a lot of resources and knowledge to be used to its full potential.

1. Inledning

1.1. Bakgrund

Denna rapport gjordes som en fördjupning inom valbart område i kursen TDDC03 - Informationssäkerhet fortsättningskurs. Momentet var en del av examinationen för kursen.

1.2. Syfte

Syftet med detta projekt var:

- att ge en djupare förståelse för the Common Criteria och hur de kan tillämpas
- att ge insikt om vad en Protection Profile är och vad ett sådant dokument kan användas till
- att projektdeltagarna, genom att själva konstruera en Protection Profile, ska skaffa sig bättre förutsättningar till införstådd läsning av Protection Profiles.

1.3. Mål

Målet med projektet var:

- att göra en litteraturstudie av Common Criteria och Protection Profiles i synnerhet
- att skriva en Protection Profile som om än inte är fullständig åtminstone berör alla delar i processen
- att redovisa en sammanfattning av arbetet och erfarenheter och synpunkter både skriftligt och muntligt.

2. Vad är en PP?

En Protection Profile är en implementationsoberoende uppsättning av säkerhetskrav för en IT-produkt eller en kategori av produkter. En Protection Profile ska bland annat beskriva produktens funktionalitet, produktens omgivning i form av hot, antaganden och policy, och definiera de mål som är nödvändiga för att möta hoten. Den ska vidare innhålla de krav som ställs på produktens och omgivningens säkerhetsfunktioer för att uppnå målen.

3. Beskrivning av projektarbetet

Första momentet var att läsa in sig på Common Criteria. Den officiella beskrivningen av CC v2.1 (ISO/IEC 15408:1999) omfattar ca 600 sidor och är inte avsedd att läsas från pärm till pärm. Det är nödvändigt med en viss grundkunskap innan det går att tillgodogöra sig den. “Information technology – Security techiques – Guide for the production of protection profiles and security targets” är skriven mer som en handledning med många exempel och den lämpade sig bättre som en inkörsport till Protection Profiles.

Nästa steg var att hitta en produkt att skriva en PP för. För att lättare kunna kontrollera omfattningen på uppgiften valdes en påhittad produkt istället för en befintlig. Produkten som valdes blev till slut en e-handelsplats för industriföretag. Produkten valdes på grund av att den för tillfället verkade vara lätt att avgränsa och innehöll flera intressanta aspekter (som till exempel nätverksanslutning, konfidentiell användardata, åtkomstkontroll och krav på tillgänglighet med mera).

Då syftet var att få större förståelse för hur Protection Profiles är uppbyggda och hur processen ser ut, skrevs dokumentet med “djupet först”. Det innebar att alla moment i processen prövades på, men vissa delar blev inte helt klara.

4. Reflektioner

Detta avsnitt sammanställer en rad olika reflektioner som framkom under arbetet att skriva PP:n.

Inläsningen tog längre tid än väntat. Det finns inte så mycket litteratur inom området. Det mesta som finns är i elektroniskt format och det materialet är av varierande kvalitet och för olika målgrupper och därför svårt att ta till sig. Den bästa källan för information visade sig vara den officiella hemsidan, som visserligen är ganska rörig

men innehåller bra referensmaterial och några exempel på färdiga Protection Profiles. Det finns en viss risk att svårigheten att hitta lämplig information på egen hand kan avskräcka nybörjare innan de hinner sätta sig in i standarden.

De officiella dokumenten som beskriver standarden innehåller mycket formaliatext och upprepningar. Det är mycket att läsa på en gång men dokumenten upplevdes ändå som välstrukturerade och givande när de första svårigheterna överkommits.

Att sätta sig in i grundprinciperna och begreppen var med inte så svårt när rätt dokument väl hittats. Dokumentationen visade sig dock bitvis vara mycket generell och ibland även tvetydig. Detta innebar att även när man tror att man förstått hur det hänger ihop stöter man snabbt på problem när det ska göras i praktiken. Många av de Protection Profiles som är certifierade är upplagda på väldigt olika sätt vilket ytterligare spär på osäkerheten hur olika delar av Common Criteria ska tolkas.

Mycket av de senare delarna i en Protection Profile bygger på att de inledande analyserna är korrekta och fullständiga. Felaktiga eller missade hot och antaganden kan göra att säkerhetskraven fokuserar på fel delar eller att vissa riskområden missas helt. Det är alltså fundamentalt att grunden är komplett och korrekt, annars kan hela PP:n bli missvisande.

Som i de flesta standarder finns det delar som där innehållet kan tolkas på flera olika sätt. När det gäller CC sköts dessa tolkningsärenden av CC Interpretations Management Board (CCIMB) som med jämna mellanrum publicerar tilläggstolkningar som gäller före standarden. För att använda CC korrekt räcker det alltså inte med att lära sig standarden utan man måste även hålla koll på de tillägg som görs.

Det verkar ganska uppenbart att det krävs längre tids erfarenhet för att processen ska kunna användas på ett effektivt sätt.

Dessa iaktaganden ledde fram till ett par funderingar angående ekonomin och lönsamheten bakom Common Criteria och Protection Profiles. Att det är en kostam process är det nog inget tvivel om. Är det då verkligen lönsamt att lägga ner stora resurser på att skriva PP:s, ST:s, eller för att få sin produkt certifierad? Det kan finnas fler anledningar. Certifiering av Windows 2000 enligt Common Criteria krävde tre års arbete samt miljontals dollar. Inom vissa områden som till exempel militär och stat, är det ett absolut krav att produkten är certifierad. Det kan även vara en konkurrensfördel inom andra områden, kunden upplever kanske produkten som säkrare än andra, även om detta inte nödvändigtvis är sant.

Detta är kanske ett mindre problem för stora företag än för små. Småföretag med en mindre kundkrets bör antagligen tänka sig för både en och två gånger innan man bestämmer sig för att anpassa sig enligt en så omfattande standard. Processen är inget man inför på några veckor utan kräver ordentlig utbildning för att utnyttjas på rätt sätt.

1 Introduction

1.1 Identification

Title: Protection Profile for Heavy Metal Inc. eMarketplace Service
Authors: Martin Öberg, Tomas Johansson
PP Version: v0.1, May 2003
CC Version: v2.1, August 1999
Keywords: access control, data integrity, availability

1.2 Overview

This Protection Profile defines the security requirements for a electronic business-to-business marketplace service to be used by Heavy Metal Inc. The PP is a draft and it is not complete.
This PP was produced as part of the examination in the course TDDC03 Information Security at Linköping Institute of Technology, spring 2003. The purpose was to gain a deeper understanding of Protection Profiles and Common Criteria in general, not to produce a complete PP. The TOE is a fictional product, invented to provide some interesting cases for the task at hand. The problem was dealt with in a depth first approach, only a subset of the security requirements was included, which allowed the authors to work with all parts of the process.
The TOE consists of a server which is responsible for the secure storage, processing and publication of customer data. The TOE is accessed by clients from an external network. The TOE must prevent unauthorized users from accessing the system and prevent authorized users from accessing information that they are not allowed to access. The TOE must ensure the integrity and the availability of the information stored by users. This protection profile is developed to suit the needs for Heavy Metal Inc. The protection profile is not intended to be generally applicable to other marketplace services. This protection profiles provides a level of protection suitable for a well managed environment with non-hostile users and a moderate risk to assets. The selected assurance level is EAL3.

1.3 Terms

This profile uses the following terms which are described in this section to aid in the application of the requirements:
User
A user is an individual who attempts to invoke a service provided by the TOE.
Authorized User
An authorized user is a user who has been properly identified and authenticated as a legitimate user.
Attacker
An attacker is an individual who deliberately tries to access information or services that he or she is not authorized to access, or who tries to prevent authorized users to access the TOE.
Access
Access to information is defined as the action of creating, retrieving, modifying or deleting information in the TOE. Access to services is defined as the action of using services provided by the TOE.

2 TOE Description

The TOE is a electronic marketplace that allows users to securely exchange services and goods using a client-server solution. The intended TOE users are large corporations looking for national and international trade partners.
Users connect to the TOE remotely with a client application. Access to the TOE is limited. A user must be registered to use the services provided by the TOE. Users must be identified and authorized every time they access the TOE.
Authorized users may submit information regarding available goods and services he or she wishes to sell. This information is available to all authorized users. Authorized users may also place bids on goods and services offered by other authorized users. Information regarding bids should only be available to the seller.

The user data is the most sensitive asset in the system since loss of confidentiality can cause serious finacial damage to the affected users. It is also important to keep the system available to users at all times as the users otherwise may miss valuable business opportunities.
The TOE is the software that runs the server-side services described above. The TOE does not include hardware, operating systems or the client application.

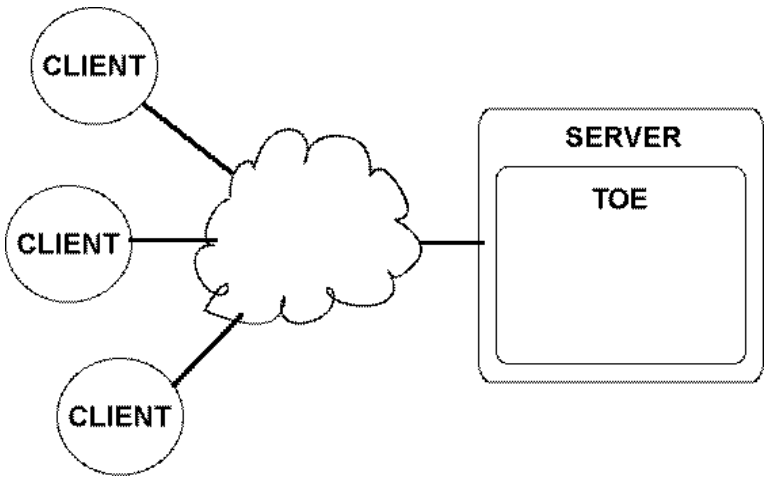

Illustration 1System overview

3 Security Environment

This chapter defines the TOE security environment. This includes all assumptions made regarding environment, usage and personnel, and the threats identified.

3.1 Assumptions

The following assumptions are made regarding the security environment.

3.1.1 Environment Assumptions

A.SECURELOC	The TOE is located in a secure location which only trusted personnel has access to. An attacker cannot physically modify the TOE.

3.1.2 Usage Assumptions

A.INSTALL	The TOE is assumed to have been properly installed and configured prior to being taken into service.

3.1.3 Personnel Assumptions

A.TRUSTED	The administrators of the TOE are assumed to be trustworthy and competent.
A.FORWARD	Authorized users are assumed not to forward any information retrieved from the TOE to any recipients not authorized to access that information.
A.SMARTUSERS	It is assumed the TOE users can be trusted to keep their login credentials securely.

3.2 Threats

The following threats have been identified:

User data confidentiality

T.EAVESDROP	An attacker may gain access to confidential information stored or retrieved by authorized users by eavesdropping on communication between the user and the TOE.
T.ACCESS	An authorised TOE user may gain access to information submitted by other users that he is not allowed to access.
T.CRACKER	An attacker may gain unauthorized access to information stored in the TOE by impersonating an authorized user.

TOE availability

T.LOCALDOS	An authorized user may prevent other authorized users from accessing the TOE by consuming excessive resources, either by mistake or intentionally.

User data integrity

TE.INTEGRITY	The information stored in the TOE may be destroyed or corrupted by hardware malfunctions.
T.TRANSMISSION	Information may be corrupted or lost while being transmitted between authorized users and the TOE due to transmission errors.

The following threats have been identified but cannot be countered:

TE.REMOTEDOS	An attacker prevents authorized access to the TOE by consuming resources.

4 Security Objectives

This chapter defines the security objectives of the TOE security functions.

4.1 Security Objectives for the TOE

O.SECCOMM	The TOE will ensure the confidentiality of information transmitted between the TOE and TOE users.
O.AUTH	The TOE will authenticate a claimed user identity before giving the user access to the TOE.
O.ACCESS	The TOE will ensure that users can only access information that they are allowed to access.
O.LOCALDOS	The TOE will limit the resources available to each TOE user.
O.INTCOMM	The TOE will have the capability to detect the loss of integrity regarding information transmitted between the TOE and TOE users.

4.2 Security Objectives for the IT-Environment

OE.BACKUP	The TOE administrators must ensure that they are able to restore the state of the TOE after system failure.

5 Functional Requirements

This chapter defines the functional requirements for the TOE. Functional requirements components in this profile were drawn from Part 2 of the CC. This section is not complete, not all dependencies for the primary functional requirements have been added. No primary requirement for objective O.SECCOMM have been added.

5.1 TOE Security Functional Requirements

The functional security requirements for this Protection Profile consists of the following components:

Component	*Component Name*
Class FIA: Identification and authentication	
FIA_UAU.2	User authentication before any action
FIA_UID.2	User identification before any action
Class FDP: User data protection	
FDP_UIT.2	Source data exchange recovery
Class FRU: Resource utilisation	
FRU_RSA.1	Maximum quotas
Class FTA: TOE Access	
FTA_MCS.1	Basic limitation on multiple concurrent sessions
Class FPT: Protection of the TSF	
FPT_AMT.1	Abstract machine testing
FPT_RCV.1	Manual recovery
FPT_TST.1	TSF Testing

Table 1Security functional requirements

5.1.1 Class FIA

FIA_UAU.2 **User authentication before any action**

Hierarchical to: FIA_UAU.1 Timing of authentication

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

Dependencies: FIA_UID.1 Timing of identification

FIA_UID.2 **User identification before any action**

Hierarchical to: FIA_UID.1 Timing of identification

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

Dependencies: No dependencies.

5.1.2 Class FPT

FPT_AMT.1 **Abstract machine testing**

Hierarchical to: No other components.

FPT_AMT.1.1 The TSF shall run a suite of tests *periodically during normal operation* to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF.

Dependencies: **No dependencies**

FPT_RCV.1 **Manual recovery**

Hierarchical to: No other components.

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

Dependencies: FPT_TST.1 TSF testing
AGD_ADM.1 Administrator guidance
ADV_SPM.1 Informal TOE security policy model

FPT_TST.1 **TSF testing**

Hierarchical to: No other components.

FPT_TST.1.1 The TSF shall run a suite of self tests *periodically during normal operation* to demonstrate the correct operation of the TSF.

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

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

Dependencies: FPT_AMT.1 Abstract machine testing

5.2 Environment Security Requirements

Have been left to be defined.

6 Assurance Requirements

This chapter defines the assurance requirements for the TOE. Assurance requirement components are Evaluation Assurance Level (EAL) 3, augmentated with ADV_SPM.1, from part 3 of the CC.

6.1 Class ACM: Configuration Management

6.1.1 Authorization Controls (ACM_CAP.3)

Developer action elements:

ACM_CAP.3.1D The developer shall provide a reference for the TOE.

ACM_CAP.3.2D The developer shall use a CM system.

ACM_CAP.3.3D The developer shall provide CM documentation.

Content and presentation of evidence elements:

ACM_CAP.3.1C The reference for the TOE shall be unique to each version of the TOE.

ACM_CAP.3.2C The TOE shall be labelled with its reference.

ACM_CAP.3.3C The CM documentation shall include a configuration list and a CM plan.

ACM_CAP.3.4C The configuration list shall describe the configuration items that comprise the TOE.

ACM_CAP.3.5C The CM documentation shall describe the method used to uniquely identify the configuration items.

ACM_CAP.3.6C The CM system shall uniquely identify all configuration items.

ACM_CAP.3.7C The CM plan shall describe how the CM system is used.

ACM_CAP.3.8C The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.

ACM_CAP.3.9C The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.

ACM_CAP.3.10C The CM system shall provide measures such that only authorized changes are made to the configuration items.

Evaluator action elements:

ACM_CAP.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.1.2 Coverage (ACM_SCP.1)

Developer action elements:

ACM_SCP.1.1D The developer shall provide CM documentation.

Content and presentation of evidence elements:

ACM_SCP.1.1C The CM documentation shall show that the CM system, as a minimum, tracks the following: the TOE implementation representation, design documentation, test documentation, user documentation, administrator documentation, and CM documentation.

ACM_SCP.1.2C The CM documentation shall describe how configuration items are tracked by the CM system.

Evaluator action elements:

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

6.2 Class ADO: Delivery and Operation

6.2.1 Delivery Procedures (ADO_DEL.1)

Developer action elements:

ADO_DEL.1.1D The developer shall document procedures for delivery of the TOE or parts of it to the user.

ADO_DEL.1.2D The developer shall use the delivery procedures.

Content and presentation of evidence elements:

ADO_DEL.1.1C The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site.

Evaluator action elements:

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

6.2.2 Installation, Generation, and Start-up Procedures (ADO_IGS.1)

Developer action elements:

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

Content and presentation of evidence elements:

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

Evaluator action elements:

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.

6.3 Class ADV: Development

6.3.1 Functional Specification (ADV_FSP.1)

Developer action elements:

ADV_FSP.1.1D The developer shall provide a functional specification.

Content and presentation of evidence elements:

ADV_FSP.1.1C The functional specification shall describe the TSF and its external interfaces using an informal style.

ADV_FSP.1.2C The functional specification shall be internally consistent.

ADV_FSP.1.3C The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing details of effects, exceptions and error messages, as appropriate.

ADV_FSP.1.4C The functional specification shall completely represent the TSF.

Evaluator action elements:

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

ADV_FSP.1.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional requirements.

6.3.2 High-Level Design (ADV_HLD.2)

Developer action elements:

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

Content and presentation of evidence elements:

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 effects, exceptions and error messages, as appropriate.

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

Evaluator action elements:

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.

6.3.3 Correspondence Demonstration (ADV_RCR.1)

Developer action elements:

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

Content and presentation of evidence elements:

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.

Evaluator action elements:

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

6.3.4 ADV_SPM.1 Informal TOE security policy model

Developer action elements:

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

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

Content and presentation of evidence elements:

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.

Evaluator action elements:

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

6.4 Class AGD: Guidance Documents

6.4.1 Administrator Guidance (AGD_ADM.1)

Developer action elements:

AGD_ADM.1.1D The developer shall provide administrator guidance addressed to system administrative personnel.

Content and presentation of evidence elements:

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 documents supplied for evaluation.

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

Evaluator action elements:

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

6.4.2 User Guidance (AGD_USR.1)

Developer action elements:

AGD_USR.1.1D The developer shall provide user guidance.

Content and presentation of evidence elements:

AGD_USR.1.1C The user guidance shall describe the functions and interfaces available to the nonadministrative 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.

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

Evaluator action elements:

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

6.5 Class ALC: Life Cycle Support

6.5.1 Identification of Security Measures (ALC_DVS.1)

Developer action elements:

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

Content and presentation of evidence elements:

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 the development and maintenance of the TOE.

Evaluator action elements:

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

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

6.6 Class ATE: Security Testing

6.6.1 Coverage (ATE_COV.2)

Developer action elements:

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

Content and presentation of evidence elements:

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 described in the functional specification.

ATR_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.

Evaluator action elements:

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

6.6.2 Depth (ATE_DPT.1)

Developer action elements:

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

Content and presentation of evidence elements:

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.

Evaluator action elements:

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

6.6.3 Functional Testing (ATE_FUN.1)

Developer action elements:

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

ATE_FUN.1.2D The developer shall provide test documentation.

Content and presentation of evidence elements:

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.

Evaluator action elements:

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

6.6.4 Independent Testing (ATE_IND.2)

Developer action elements:

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

Content and presentation of evidence elements:

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.

Evaluator action elements:

ATE_IND.2.1.E 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.

6.7 Class AVA: Vulnerability Assessment

6.7.1 Examination of Guidance (AVA_MSU.1)

Developer action elements:

AVA_MSU.1.1D The developer shall provide guidance documentation.

Content and presentation of evidence elements:

AVA_MSU.1.1C 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.1.2C The guidance documentation shall be complete, clear, consistent and reasonable.

AVA_MSU.1.3C The guidance documentation shall list all assumptions about the intended environment.

AVA_MSU.1.4C The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).

Evaluator action elements:

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

AVA_MSU.1.1E The evaluator shall repeat all configuration and installation procedures to confirm that the TOE can be configured and used securely using only the supplied guidance documentation.

AVA_MSU.1.3E The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected.

6.7.2 Strength of TOE Security Function Evaluation (AVA_SOF.1)

Developer action elements:

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 PP/ST.

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

Evaluator action elements:

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.

6.7.3 Developer Vulnerability Analysis (AVA_VLA.1)

Developer action elements:

AVA_VLA.1.1D The developer shall perform and document an analysis of the TOE deliverables searching for obvious ways in which a user can violate the TSP.

AVA_VLA.1.2D The developer shall document the disposition of obvious vulnerabilities.

Content and presentation of evidence elements:

AVA_VLA.1.1C The documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

Evaluator action elements:

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

AVA_VLA.1.2E The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure obvious vulnerabilities have been addressed.

7 Rationale

7.1 Rationale for (IT) Security Objectives

O.ACCESS This security objective is necessary to counter the threat T.ACCESS because it only lets users access information they are allowed to access.

O.AUTH This security objective is necessary to counter the threats T.ACCESS and T.CRACKER because it requires that the users are uniquely identified before granting access to the TOE.

O.INTCOMM This security objective is necessary to handle the threat T.TRANSMISSION as it detects loss of integrity in the transmissions.

O.LOCALDOS This security objective is necessary to counter the threat T.LOCALDOS because it limits the resources available for any given user.

O.SECCOMM This security objective is necessary to counter the threat T.EAVESDROP as it ensures the confidentiality of the transmitted information.

7.2 Rationale for Security Objectives for the Environment

OE.BACKUP This security objective for the environment is necessary to counter the threat TE.INTEGRITY becauses it ensures that the administrators of the TOE are able to restore the state of the TOE after system failure

7.3 Rationale for Security Requirements

Note: Functional requirements meeting the security objective O.SECCOMM have been left to be defined.

	O.ACCESS	O.AUTH	O.INTCOMM	O.LOCALDOS	O.SECCOMM	OE.BACKUP
FDP_UIT.2			**X**			
FIA_UAU.2	**X**	**X**				
FIA_UID.2	**X**	**X**				
FRU_RSA.1				**X**		
FTA_MCS.1				**X**		
FPT_AMT.1						**X**
FPT_RCV.1						**X**
FPT_TST.1						**X**

Table 2Requirements to objectives mapping

FIA_UAU.2 **User authentication before any action**
This component ensures that all users' claimed identities are authenticated before they are allowed to access any information in the TOE. A secure user identity is required to enforce the access restrictions in the TOE. This component traces back to and aids in meeting the objectives O.ACCESS and O.AUTH.

FIA_UID.2 **User identification before any action**
This component ensures that all users are forced to identify themselves before allowed to access any information in the TOE. This component traces back to and aids in meeting the objectives O.ACCESS and O.AUTH.

FDP_UIT.2 **Source data exchange recovery**
This component ensures the integrity of user information transmitted between the TOE and the TOE users. It traces back to and aids in meeting the objective O.INTCOMM.

FRU_RSA.1 **Maximum quotas**
Limits the amount of TOE resources each user can consume to reduce the risk of one or several users overloading the system. This component traces back to and aids in meeting the objective O.LOCALDOS.

FTA_MCS.1 **Basic limitation on multiple concurrent sessions**

	Limits the number of concurrent session for each user to to reduce the risk of one or several users overloading the system. This component traces back to and aids in meeting the objective O.LOCALDOS.
FPT_AMT.1	**Abstract machine testing** This component is a dependency from FPT_TSF.1 and aids in meeting objective OE.BACKUP.
FPT_RCV.1	**Manual recovery** This component provides the administrators means of restoring the system state of the TOE after a system failure. This is needed to meet objective OE.BACKUP.
FPT_TST.1	**TSF Testing** This component is a dependency from FPT_RCV.1 and aids in meeting objective OE.BACKUP.