Confidential

RedCastle v2.0
Security Target

Doc. No.	RCX06D-ASE-07
Version	Version 1.7
Last Update	2006. 09. 27.

Document History

Doc. No.	Summary	Date	Author
RCX06D-ASE-01	Initial version - RedCastle v2.0 for Asianux	2006-02-21	SC. Kim
RCX06D-ASE-02	Minor updates - Add item of assumption - Add security objectives - Update attack possibilities	2006-03-08	SC. Kim
RCX06D-ASE-03	Minor Updates - Update TOE Scope - Supplement security requirements for the IT environment - Additional rationale description for the IT environment requirements - Additional description of functional strength - Supplement assurance measure	2006-03-14	SC. Kim
RCX06D-ASE-04	ST modification - Additional description of functional strength	2006-03-17	SC. Kim
RCX06D-ASE-05	- 1.4 CC Conformance: Add conformance item for package name - 2.2.2 logical scope and boundaries: identification of non-security item - 5.1 TOE Security Functional Requirements: Add FPT_ITT.1 component - 5.2 TOE Security Assurance Requirements: Reflect CC version 2.3 - Modify contents according to PP v1.1	2006-08-14	HS. Yoon
RCX06D-ASE-06	- Modify contents according to PP v1.1 . Modify according to details revision	2006-09-06	HS. Yoon

RCX06D-ASE-07	-. 5.4 Strength of Function: Modify to the SOF-medium	2006-09-27	HS. Yoon

Table of Contents

List of figures

List of tables

1 Introduction

This is the Security Target document for the evaluation of RedCastle v2.0 for Asianux.

1.1 ST Identification

This document is the security target for the CC evaluation of the RedCastle v2.0, which is the one of the access control software for the commercial operating system, and is conformable to the Common Criteria for Information Technology Security Evaluation [CC] with extensions as defined in the Label-based Access Control System Protection Profile [LSAPP].

[Table 1-1] ST Identification

Title	ST for RedCastle v2.0 for Asianux
Doc. Version	Version 1.7
Date	2006. 09. 27.
Doc. No.	RCX06D-ASE-07
Product Name	RedCastle v2.0 for Asianux
TOE Name	RedCastle v2.0 for Asianux
OS	Asianux 2.0
Evaluation Basis	Common Criteria for Information Security System (No.2005-25 announced by MIC, Korea)
PP Claims	Label-based Access Control System Protection Profile for Government v1.1, 2006-05-17 (LACSPP)
Assurance Level	EAL3+
Authors	SC Kim, HS Yoon
Keywords	MLS, MAC, DAC, Linux

1.2 Conventions and Terminology

This section contains definitions of technical terms that are used with a meaning specific to this document. Terms defined in the [CC] are not reiterated here, unless stated otherwise.

1.2.1 Conventions

Iteration – the usage of a component more than once with varying operations. The result of iterated operation is marked by the serial number in a parenthesis. For examples, FAU_SAR.3(1), FAU_SAR.3(2)**.**

Selection – the specification of one or more items from a list in a component. The result of selection operation is marked by the underlined italic letters.

Refinement – the addition of details to a component. The result of refinement operation is marked by the bold letters.

Assignment – the specification of an identified parameter in a component. The result of assignment operation is marked by the square bracket. For example, [assignment_values].

Application Notes – It is provided when the meaning of requirements are needed to be more clear.

Identifier of security function – the only one identifier assigned to the security function. The usage is <function name>.# , for example, Audit.1 or Admin.1 and so on.

1.2.2 Terminology

Assets - Information or resources to be protected by the countermeasures of a TOE.

Assignment - The specification of an identified parameter in a component.

Attack potential - The perceived potential for success of an attack, expressed in terms of an attacker’s expertise, resources and motivation.

Augmentation - The addition of one or more assurance component(s) from Part 3 to an EAL or assurance package.

Authentication Data - Information used to verify the claimed identity of a user.

Authorized Administrator - A person authorized for the TOE administration.

Authorized user - A user who may, in accordance with the TSP, perform an operation.

Class - A grouping of families that share a common focus.

Component - The smallest selectable set of elements that may be included in a PP, and ST, or a package.

Dependency - A relationship between requirements such that the requirement that is depended upon must normally be satisfied for the other requirements to be able to meet their objectives.

Element - An indivisible security requirement.

Evaluation Assurance Level (EAL) - A package consisting of assurance components from Part 3 that represents a point on the CC predefined assurance scale.

External IT Entity - Any IT product or system, distrusted or trusted, outside of the TOE that interacts with the TOE.

Family - A grouping of components that share security objectives but may differ in emphasis or rigour.

Human User - Any person who interacts with the TOE.

Identity - A representation (e.g. a string) uniquely identifying an authorized user, which can either be the full or abbreviated name of that user or a pseudonym.

Iteration - The use of a component more than once with varying operations.

Object - An entity within the TSC that contains or receives information and upon which subjects perform operations.

Organizational Security Policies – One or more security rules, procedures, practices, or guidelines imposed by an organization upon its operations.

Protection Profile (PP) - An implementation-independent set of security requirements for a category of TOEs that meet specific consumer needs.

Role - A predefined set of rules establishing the allowed interactions between a user and the TOE.

Refinement - The addition of details to a component.

Security Target (ST) - A set of security requirements and specifications to be used as the basis for evaluation of an identified TOE.

Selection - The specification of one or more items from a list in a component.

Strength-of-Function (SOF) - A qualification of a TOE security function expressing the minimum efforts assumed necessary to defeat its expected security behavior by directly attacking its underlying security mechanisms.

SOF-basic - A level of the TOE strength of function where analysis shows that the function provides adequate protection against casual breach of TOE security by attackers possessing a low attack potential.

Target of Evaluation (TOE) - An IT product or system and its associated administrator and user guidance documentation that is the subject of an evaluation.

TOE Security Function (TSF) - A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.

TOE Security Policy (TSP) - A set of rules that regulate how assets are managed, protected and distributed within a TOE.

TSF Data - Data created by and for the TOE, that might affect the operation of the TOE.

TSF Scope of Control (TSC) - The set of interactions that can occur with or within a TOE and are subject to the rules of the TSP.

Subject

An entity within the TSC that causes operations to be performed.

User

Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

Discretionary Access Control (DAC) - A means of restricting access to objects based on the identity of subjects and/or groups to which they belong.

Mandatory Access Control (MAC) - A means of restricting access to objects based on the sensitivity (as represented by a label) of the information contained in the objects and the formal authorization (i.e., clearance) of subjects to access information of such sensitivity.

Common Criteria for Information Security System

Common Criteria(CC), is meant to be used as the basis for evaluation of security properties of IT products and systems. Common Criteria for Information Security System was translated into Korean for CC version 2.3. It was announced by the Ministry of Information and Communication in Korea at 21 May, 2005.

Abstract Machine - A theoretical model of a computer hardware of software system. If TOE is application program, it will be Operating System, and if TOE is operating system, it will be firmware or hardware.

RedCastle

The product name produced by RedGate, which is the target of this evaluation.

RedCastle ESM

The management program of RedCastle, which provide GUI Interface for Administrator.

RedCastle SecureOS

The agent program of RedCastle, which is installed on server system and performs security functions include access control, integrity check, security policy management, etc..

1.3 ST Overview

The ST overview is brief description of the TOE and detailed explanation will be given in the section 2.

The TOE is RedCastle v2.0 that is Label-based Access Control System produced by Redgate Co., Ltd.. This TOE is label-based access control system, and performs as if application program in OS. This TOE provides the following access control policy functions to prevent violation attempt against information assets reside in server system.

This TOE provides the user with multi-level based MAC(Mandatory Access Control), and the ACL(Access Control List) based DAC(Discretionary Access Control). Also, it supports the function of privileges or permission based DAC providing in OS.

The TOE consists of the RedCastle Agent and the RedCastle Manager logically.

- RedCastle Agent: RedCastle SecureOS
- RedCastle Manager: RedCastle ESM(Enterprise Security Management)

The RedCastle SecureOS(RedCastle Agent) is loaded into each OS, and is consisted of the kernel part performing MAC and DAC etc. and the application part performing other security functions. The main access control functions provided in this TOE are as follows.

- Mandatory Access Control (MAC): It provides the MAC policies based on user and multi-level security.
- Discretionary Access Control (DAC): It provides the DAC policy by ACL(Access Control List) and Privileges or Permission.

The RedCastle ESM (RedCastle Manager) is operated in Windows 2000 Professional (Service Pack 4) environment and is provided through GUI interface.

The RedCastle SecureOS is operated in Asianux 2.0(below, LINUX) environment and is consisted of kernel module that performs main security functions of access control etc. and application for security administration.

All data that communicate between the security functions management part and the security administration part are encrypted through SSL(Secure Socket Layer) protocol.

1.4 CC Conformance

This ST confors with the following:

This ST is CC Part2 extended and Part 3 conformant, with a claimed Evaluation Assurance Level of EAL3+.

- Label-based Access Control System Protection Profile, v1.1
- Common Criteria v2.3, Part 2
- Common Criteria v2.3, Part 3
- This ST is CC Part 2 extended and Part 3 conformant, augmented by ADV_IMP.2, ADV_LLD.1, ALC_TAT.1, ATE_DPT.2, AVA_VLA.2, with a claimed Evaluation Assurance Level of EAL3+

The claimed minimum strength of function (SOF) for this TOE is SOF-medium.

1.5 ST Structure

The structure of this document is defined by CC Part 1.

Section 1 is the Introduction of this ST.

Section 2 is the TOE Description.

Section 3 provides the statement of TOE security environment.

Section 4 provides the statement of security objectives.

Section 5 provides the statement of IT security requirements.

Section 6 provides the TOE summary specification, which includes the detailed specification of the IT Security Functions.

Section 7 provides the Protection Profile claim.

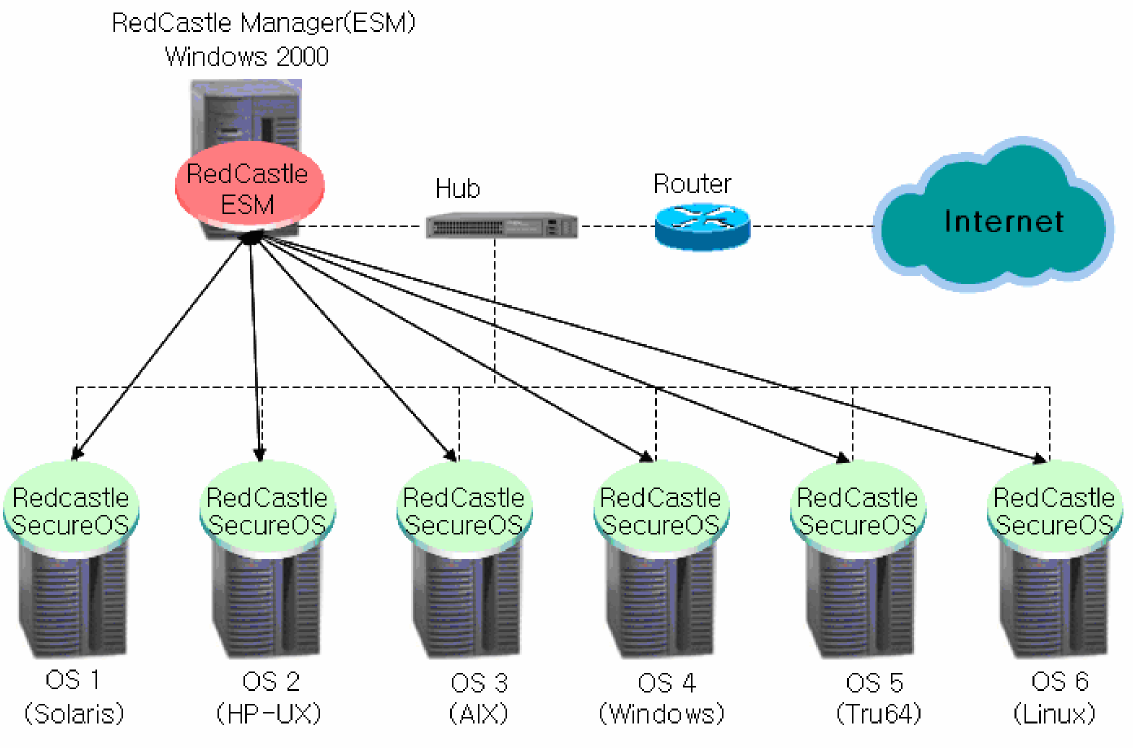
Section 8 provides the rationale for the security objectives, security requirements and the TOE summary specification.

2 TOE Description

The target of evaluation (TOE) is RedCastle v2.0, which is the Label-based Access Control System and it is installed and operated on the Asianux 2.0 Operating System.

2.1 Product Type

The RedCastle v2.0 is label-based access control system operating in OS as a program. This TOE provides MAC and DAC policies, and also supports embedded DAC policies in the OS. This TOE's brief operating configuration is as follows.



[Figure 2-1] Typical Configuration of RedCastle

This TOE consists of the RedCastle SecureOS which is RedCastle Agent and the RedCastle ESM (Enterprise Security Management) which is RedCastle Manager.

The main functions of RedCastle SecureOS are as follows.

- Reference Monitor
- Multi-Level based MAC(Mandatory Access Control)
- ACL based DAC(Discretionary Access Control)
- Allowed/Denied List based DAC
- SecureOS Authentication
- Audit Generation and Collection
- Potential Violation Analysis
- Simple Attack Prevention
- Security Attributes and Security Policies Management
- Security Administration Function
- Simulated Operation
- Integrity Check Functions
- CLI(Command Line Interface) of the RedCastle Agent(RedCastle Secure OS)

The main functions of RedCastle ESM are as follows.

- ESM Authentication
- ESM Screen Saving
- SecureOS Authentication GUI
- Audit Log GUI
- Real-time Audit Log and Security Alarm GUI
- Reporting Function
- Security Attributes and Security Policies GUI
- Audit Functions Configuration GUI
- Simulated Operation GUI
- Integrity Check Functions GUI

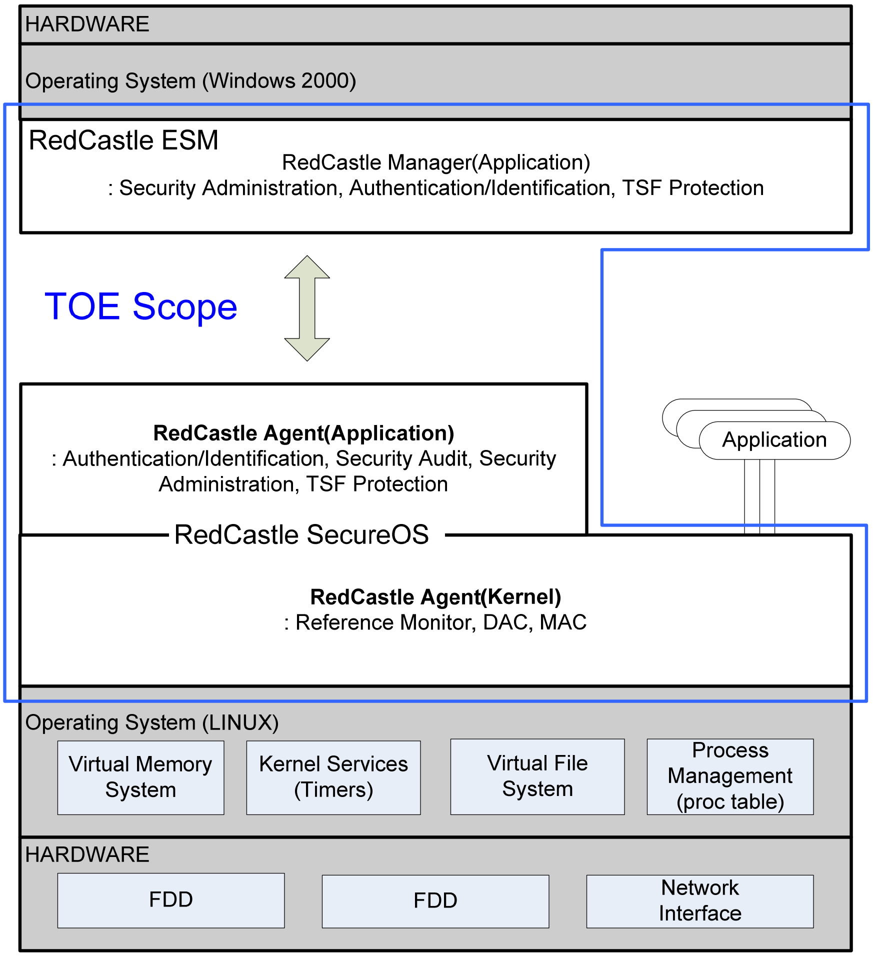
The communication between RedCastle ESM and RedCastle Secure OS uses the SSL(Secure Socket Layer) version 3 protocol to encrypt all communicating data.

2.2 TOE Scope and Boundaries

This section describes physical/logical scope and boundaries of TOE. Physical scope descriptions will be divided into hardware, software, and etc. The logical scope description will be about security functions of which the TOE provides.

2.2.1 Physical Scope and Boundaries

This TOE is RedCastle v2.0 software. The RedCastle v2.0 is divided into the RedCastle SecureOS which is RedCastle Agent and RedCastle ESM which is RedCastle Manager. The RedCastle SecureOS will be operated in application part and kernel part of OS(LINUX) as a program and the RedCastle ESM will be operated in Windows 2000 Professional(Service Pack 4). The RedCastle SecureOS and the RedCastle ESM are connected through 10/100BaseT Ethernet.


HARDWARE
Operating System (Windows 2000)
RedCastle ESM
RedCastle Manager(Application)
: Security Administration, Authentication/Identification, TSF Protection
TOE Scope
RedCastle Agent(Application)
: Authentication/Identification, Security Audit, Security Administration, TSF Protection
Application
RedCastle SecureOS
RedCastle Agent(Kernel)
: Reference Monitor, DAC, MAC
Operating System (LINUX)
Virtual Memory System
Kernel Services (Timers)
Virtual File System
Process Management (proc table)
HARDWARE
FDD
FDD
Network Interface

[Figure 2-2] Physical TOE Scope

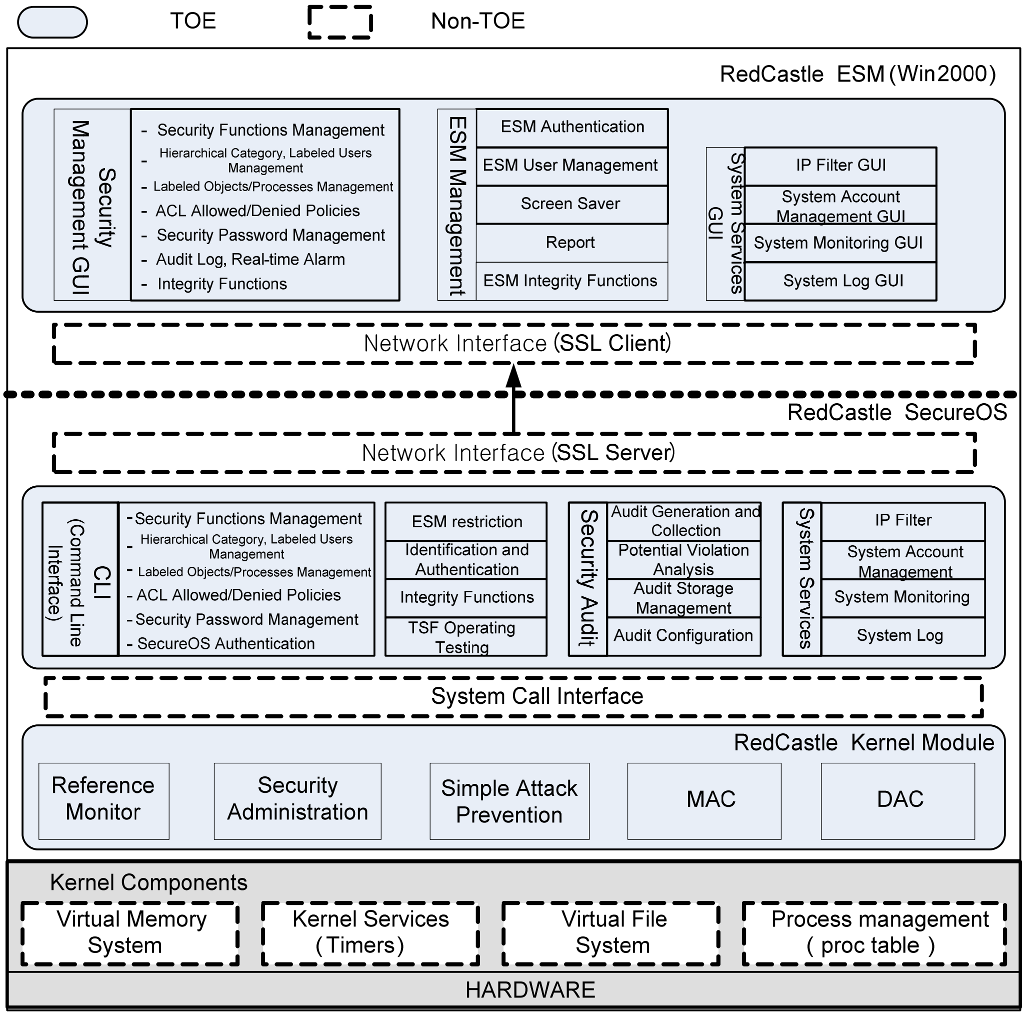
[Table 2-1] is physical environment of the TOE.

[Table 2-1] Physical TOE Environment

Items	RedCastle Agent	RedCastle Manager
Software	RedCastle SecureOS - Asianux 2.0	RedCastle ESM - Windows 2000 Professional SP4
Hardware	CPU: AMD_64 1.4GHz and above RAM: 128MB and above HDD: 200MB and above Network: 10/100BaseT	CPU: Pentium III 600 MHz and above RAM: 128 MB and above HDD: 10 MB and above Network: 10/100BaseT

2.2.2 Logical Scope and Boundaries

The [Figure 2-3] is the logical TOE scope of RedCastle v2.0.


TOE
Non-TOE
RedCastle ESM (Win2000)
Security Management GUI
- Security Functions Management
- Hierarchical Category, Labeled Users Management
- Labeled Objects/Processes Management
- ACL Allowed/Denied Policies
- Security Password Management
- Audit Log, Real-time Alarm
- Integrity Functions
ESM Management
ESM Authentication
ESM User Management
Screen Saver
Report
ESM Integrity Functions
System Services GUI
IP Filter GUI
System Account Management GUI
System Monitoring GUI
System Log GUI
Network Interface (SSL Client)
RedCastle SecureOS
Network Interface (SSL Server)
CLI (Command Line Interface)
-Security Functions Management
- Hierarchical Category, Labeled Users Management
- Labeled Objects/Processes Management
- ACL Allowed/Denied Policies
- Security Password Management
- SecureOS Authentication
ESM restriction
Identification and Authentication
Integrity Functions
TSF Operating Testing
Security Audit
Audit Generation and Collection
Potential Violation Analysis
Audit Storage Management
Audit Configuration
System Services
IP Filter
System Account Management
System Monitoring
System Log
System Call Interface
RedCastle Kernel Module
Reference Monitor
Security Administration
Simple Attack Prevention
MAC
DAC
Kernel Components
Virtual Memory System
Kernel Services (Timers)
Virtual File System
Process management (proc table)
HARDWARE

[Figure 2-3] Logical TOE Scope

The [Figure 2-3] is logical TOE scope of RedCastle that is consisting of TSF protection, security functions' GUI and CLI of reference monitor, MAC, DAC, simple attack prevention, identification and authentication, security audit, security attributes and security data management, integrity check, system services of IP Filter, system account management, and system monitoring.

[Table 2-2] Logical TOE Environment

Items	RedCastle Agent	RedCastle Manager
Security Functions	RedCastle SecureOS o SecureOS identification and authentication o Audit generation and collection o Potential violation analysis o Audit storage management o Audit review o Security functions management o Hierarchical category management, Labeled users management o Labeled Objects/ Labeled Subject(Processes) management o Management of the ACL policies o Allowed/Denied list management o Audit configuration o Security password management o Abstract machine testing o SecureOS integrity functions o Secure Communication(Server) o ESM Restriction o Security management CLI - Security functions management CLI - Hierarchical category management, Labeled users management CLI - Labeled Objects/ Labeled Subject(Processes) management CLI - ACL policies management CLI - Allowed/Denied list management	RedCastle ESM o ESM identification and authentication o ESM user management o ESM screen saving o Secure communication(Client) o ESM integrity functions o Security management GUI - ESM authentication GUI - ESM user management GUI - ESM data management GUI - SecureOS authentication GUI - Security functions management GUI - Real-time alarm GUI - TSF operating testing GUI - Audit review GUI - Hierarchical category management GUI - Labeled users management GUI - Labeled Objects/ Labeled Subject(Processes) management GUI - Discretionary policy management GUI - Audit configuration GUI - Security password management GUI - Integrity management GUI - Security functions configuration GUI

	CLI - Audit review CLI - Security password management CLI RedCastle Kernel Module o Kernel security functions management - Security functions management o Kernel security data management - Hierarchical category, Labeled users data - Labeled Objects/ Labeled Subject(Processes) security attributes - ACL polices, Allowed/Denied list - Audit data o Security violation audit generation o Reference Monitor - System call intercept - Security module separation o Simple attack prevention o MAC o ACL based DAC o Allowed/Denied list based DAC	
System Services	RedCastle SecureOS o IP Filter management o System account management o System monitoring management o System log management	RedCastle ESM o System services management GUI - IP Filter GUI - System account GUI - System monitoring GUI - System log GUI

Application part and kernel part of RedCastle ESM and RedCastle SecureOS provide the following functions.

2.2.2.1 ESM Security Functions

RedCastle ESM provides the following security functions.

- ESM Identification and Authentication: Administrator must perform ESM administrator authentication first to utilize RedCastle's security functions through ESM. ESM provides GUI for this.
- ESM Users Management: It is to manage ESM administrator who can utilize ESM. ESM provides GUI for this.
- ESM Screen Saving: When ESM user vacated the seat for long time, it enables ESM Screen Saving. ESM provides GUI to configure waiting time for this Screen Saving function.
- Secure Communication (Client): SSL version 3 protocols are used for secure communication between ESM and Secure OS, and SSL Client will be located in the ESM.
- ESM Integrity Functions: It enables integrity check over execution files of ESM or any added files by administrator.

2.2.2.2 ESM GUI

RedCastle ESM provides the following GUI for administrators to perform security functions of Secure OS.

- TSF operation monitoring GUI: It enables user to recognize state of Secure OS and state of system operates Secure OS.
- Security Functions Management GUI: It enables user to manage start and stop of security function of access control, audit data and so on.
- Real-time Audit and Alarm GUI: It enables user to monitor the audit event and the potential violation analysis events that occur in Secure OS by real-time.

- Security Log Inquiry and Review GUI: It enables user to inquire or to search the security log data that is managed in Secure OS.
- Hierarchical Category Management GUI: It enables user to inquire, add, change and delete the security category which is one of the multi-level security attributes.
- Labeled Users Management GUI: It enables user to inquire, configure, change and retrieve the security attributes of users.
- Labeled Subject (Processes) Management GUI: It enables user to inquire the security attributes of subject (processes) which is representing the user.
- Labeled Object Management GUI: It enables user to inquire, change, and retrieve the security attributes of object(files).
- ACL Policies Management GUI: It enables user to inquire, add, change and delete the ACL based DAC.
- Allowed/Denied List Management GUI: It enables user to inquire, add, modify and delete the allowed rule for setuid operation, su operation, the denied rule of the restricting command execution, the denied rule of the controlling kill signal, and the allowed rule for command execution.
- Audit Configuration GUI: It enables user to configure the audit storage path, size and alarm, and the potential violation analysis.
- Security Password management GUI: It enables user to change the security password of security users.
- Integrity Management GUI: It enables user to check integrity about execution files of TSF or added data by administrator.
- Reporting GUI: This function provides GUI that can generate reports about audit data.
- Security Functions Configuration GUI: This function provides GUI that can configure operating environment of access control and attack prevention.

Also, the RedCastle ESM provides the following GUI to provide system services in Secure OS.

- IP Filter GUI: This function provides GUI that can configure the IP Filter policies.
- System Account Management GUI: This function provides GUI that can add, modify and delete the system account.
- System Monitoring GUI: This function provides GUI that can monitor system state by real-time through configuration of disk usage, process threshold, and so on.
- System Log GUI: This function provides GUI that can inquire and review collected system log separately.

2.2.2.3 Secure OS Security Functions of Application Part

Application part of RedCastle Secure OS provides the following security functions.

- Secure OS Identification and Authentication: It is a function to identify and authenticate administrators to connect through ESM or security users for accessing files which have security attributes by using security password.
- Audit Generation and Collection: This function collects and stores security log, system log, and IP Filter log.
- Audit Storage Management: This function manages audit data storage and responses if it is saturated.
- Audit Review: This function handles requests of security officer (SO) to inquire and search security audit data.
- Security Functions Management: This function handles requests of security officer(SO) to start and stop security functions of access control, audit data, etc.
- Hierarchical Category Management: This function manages a request of security officer(SO) to inquire, add, modify and delete security category.

- Labeled Users Management: This function manages a request of security officer(SO) to inquire, configure, modify and retrieve security attributes of users.
- Labeled Subject(Processes) Inquire: This function manages a request to inquire security attributes about subject of user.
- Labeled Object Management: This function manages a request to inquire, modify and retrieve security attributes about object of security officer(SO).
- ACL Policies Management: This function manages a request of security officer(SO) to inquire, modify and delete ACL based policies.
- Allowed/Denied List Management: This function manages a request of security officer(SO) to inquire, add, modify and delete allowed/denied list.
- Audit Configuration: This function manages a request of security officer(SO) to configure audit data environment.
- Security Password Management: This function manages a request of security users(labeled users) to change security password.
- Abstract Machine Testing: It enables user to recognize state of Secure OS and system which operates Secure OS.
- Secure OS Integrity Functions: It enables user to check integrity about execution files of Secure OS or any added files by administrator.
- Secure Communication (Server): SSL version 3 protocols are used for secure communication between ESM and Secure OS, and SSL Client will be located in Secure OS.
- ESM Connection Control: It determines allowance or denial of connection to the ESM based on its IP Address and the administrator's account before the Secure OS conduct its identification and authentication process.

2.2.2.4 Secure OS CLI

RedCastle Secure OS provides administrator the following CLI to perform security attributes of Secure OS. CLI of Secure OS will be provided through Console of OS only for secure remote management.

- Security Functions Management CLI: It enables user to manage start and stop of security function of access control, audit data, etc..
- Audit Review CLI: It enables user to inquire or to search a security log and a system log that is managed in Secure OS.
- Hierarchical Category Management CLI: It enables user to inquire, add, change and delete the security category which is one of the multi-level security attributes.
- Labeled Users Management CLI: It enables user to inquire, configure, modify and retrieve the security attributes of users.
- Labeled Subject (processes) Inquire CLI: It enables user to inquire the security attributes of subject(processes) representing the user.
- Labeled Object Management CLI: It enables user to inquire, change, and retrieve the security attributes of object(files).
- ACL Policies Management CLI: It enables user to inquire, add, change and delete the ACL based DAC.
- Allowed/Denied List Management CLI: It enables user to inquire, add, modify and delete the allowed rule for setuid, su operation, the denied rule of the restricting command execution and the controlling kill signal.
- Security Password management CLI: It enables user to change the security password of security users.
- Reporting CLI: This function provides to generate reports about audit data.
- Security Functions Configuration CLI: It enables user to configure operating environment of access control and attack prevention.

- Security Users Authentication CLI: Security Users (labeled users) will be authenticated by its security password to access the object that has security attributes.

2.2.2.5 Security Functions of Secure OS Kernel Part

Kernel part of RedCastle Secure OS provides the following security functions.

- Security Functions Operation Management: This function manages a request to change Operation mode and Warning mode of access control, and simple attack prevention.
- Security Data Management: This function manages a request of security officer (SO) to inquire and modify security data as follows.
  - Rule of the security category
  - Rule of the security attributes of user
  - Management of the security attributes of subject(processes)
  - Management of the security attributes of object
  - Rule of the ACL policies
  - The allowed Setuid list for execution
  - The controlling command list for execution
  - The controlling kill signal list
  - The allowed user list for su operation
  - The allowed command list for execution
  - Kernel Log
- Violation Audit Generation: This function generates and manages audit data of security violation which is generated from access control and simple attack prevention.
- Reference Monitor
  - System Call Intercept: By loading and initiating of RedCastle Kernel module, system call of TSC will be replaced for preventing its bypasses.

  - Security Module Separation: Kernel module will be separated from the Operating System's list to prevent unauthorized interference from outside.

- Multi-Level based MAC: It will control subject's access right to the object to conduct read and write operations based on the security levels of the subject and object. It also allocates security attributes to the subject when it is generated and inherits subject's security attributes to the object when it is generated as well as retrieve its security attributes when the object is deleted. And if a registered program executed by an execution bypass allowing list, it will allow the execution through bypassing the multi-level based MAC and re-allocate security attributes described in the list.

- ACL based DAC: ACL based DAC is applied to read, write, execute, create, delete, rename, chmod, and chown operations of a subject over an object.

- Allowed/Denied List based DAC

  - The allow function for setuid execution: This function allows execution of registered setuid programs in list only.

  - The deny function of the restricting command execution: The commands registered in the list can be executed by SO only.

  - Kill signal control function : The processes registered in the list can be killed by SO only.

  - The allow function for su operation: Only registered users in the list can transit from root to su.

- Simple Attack Prevention

  - Reinforce the vulnerability of symbolic-link: It prevents the damage such as the root authoritative gain previously by blocking the illegal modulation of the file through the Symbolic link attack.

  - Prevent an illegal access to FIFO special file: Protect the FIFO by allowing the opening about their FIFO only.

  - Prevent an illegal creation of Hard-link file: If the owner of hard link file to create and the subject's uid is different, it will be recognized as a trial to create unauthorized hard link.

- Prevent the vulnerability of CHROOT: If an executable process on the CHROOT environment requests a system call which could attack CHROOT, it will be detected and blocked.
- Prevent attempts to switch the promiscuous mode: It detects that the network devices are changed into the Promiscuous mode.

3 TOE Security Environment

The statement of TOE security environment identifies the list of assumptions made on the operational environment and the intended method of use of the product, defines the threats that the product is designed to counter, and the organizational security policies with which the product is designed to comply.

3.1 Assumptions

This section contains secure usage assumptions regarding the IT security environment.

3.1.1 Assumptions identical to LACSPP

IT Security Environment of this TOE has the identical assumptions to the ones of “Label-based Access Control System Protection Profile [LSAPP], v1.1.

A.LOCATE

It is assumed that the processing resources of the TOE will be located within controlled access facilities and will be protected from unauthorized physical modification.

A.ADMINS

It is assumed that there are one or more competent individuals who are assigned to manage the TOE and the security of the information it contains. Such personnel are assumed not to be careless, willfully negligent or hostile.

A.PATCH

It is assumed that the operating system, where TOE is installed, is secured and reliable. Before installing the TOE, the operating system will be patched the vulnerabilities and removed the useless services.

3.1.2 Additional Assumptions

These assumptions are added for this ST.

A. SSL

The SSL (Secure Socket Layer) protocol, which is used for the secure communication between RedCastle Agent and Manager, is secured.

A. TIME

It is reliable for the timestamp of Operating System to be used by the TSF of this TOE.

3.2 Threats

The TOE must counter threats to security as below. The assumed level of expertise of the attacker for all the threats is unsophisticated, with access to standard equipment and public information.

3.2.1 Threats countered by the TOE

T. CODE

The TOE itself can be vulnerable if the developers did not develop the TOE in according to the specifications appropriately or if he includes the defect codes by accident or on purpose.

T. AUDIT

The auditable events of TOE, caused by the exhausting storage attack, cannot be audited.

T. INTEGRITY

An authorized or unauthorized user of the TOE could do the unauthorized modification or destruction of the configuration data or the sensitive information, resulting in breaking up the security functions of TOE.

T. UAUTH

The unauthorized authentication to the TOE could be tried by an unauthorized user or threat agent.

T. BYPASS

An unauthorized user may attempt to bypass the IT Environment's information flow control policy to gain access to data stored on a protected by IT system.

T. RESIDUE

Because an object is logically deleted (not available to the user but still within the system and may be recoverable), a threat agent could reuse the residual information illegally that is contained in a deleted object.

3.2.2 Threats within the TOE Environment

TE. MISUSE

The TOE could be configured, managed, operated by an authorized administrator.

TE. INSTALL

During the installation of TOE, the security of TOE could be broken off by a user who is installing.

3.3 Organizational security policies

This TOE complies with the following organizational security policies.

P. ACCOUNTABLE

The users of the system shall be held accountability for their actions within the system.

P. MAC

The right to access specific labeled data objects is determined on the basis of the security label of subject.

P. LABEL

The TOE must assign and revoke the security label of the subject and the object according to the organization access control policies.

P. IA

Only those users who have been authorized to access the information within the system may access the system.

P. ADMIN

An authorized administrator must manage the TOE securely.

P. CIPHER

The encryption algorithm and its modules used for the TOE must be certified by the National Intelligence Service (NIS) in Korea.

P. DAC

The right to access specific data objects is determined on the basis of the identity of user or group.

4 Security Objectives

The statement of security objectives shall define the security objectives for the TOE and its environment. The security objectives shall address the entire security environment aspects identified.

4.1 Security Objectives for the TOE

All security objectives listed in this section are targeted at the TOE itself.

O. AUDIT

The TSF must record the security relevant actions of users of the TOE. The TOE must provide the means of investigating any security relevant events.

O. MAC

The TOE must enforce the access to resources on the basis of the security label of subject and object.

O. CODE

The source codes which are generated by developers must be inspected whether they have some defects or not.

O. MANAGE

The TSF must provide all the functions and facilities necessary to support administrative users that are responsible for the management of TOE security and must ensure that only administrative users are able to access such functionality.

O. INTEGRITY

The TOE must protect the TSF data or the reliable data from the unauthorized disclosure, modification, and deletion.

O. LABEL

The TOE must assign and revoke the security label of the subject and the object according to the organization access control policies.

O. IA

The TOE must identify a user uniquely and ensure that only authorized users gain access to the TOE and its resources.

O. DAC

The TOE must enforce the access to resources on the basis of the identity of user or group.

O. PROTECT

The TOE must protect itself from the deactivation or the modification of the TOE.

O. RESIDUE

The TOE must ensure that any information contained in a protected resource is not released when the resource is recycled.

4.2 Security Objectives for the TOE Environment

All security objectives listed in this section are targeted at the non-IT environment of the TOE.

OE. LOCATE

It must be ensured that the processing resources of the TOE will be located within controlled access facilities and will be protected from unauthorized physical modification.

OE. ADMINS

It must be ensured that there are one or more competent individuals who are assigned to manage the TOE and the security of the information it contains. Such personnel are assumed not to be careless, willfully negligent or hostile.

OE. MANAGE

The TOE must be installed securely and be configured, managed, and used by an only authorized administrator.

OE. PATCH

It must be ensured that the operating system, where TOE is installed, is secured and reliable. Before installing the TOE, the operating system will be patched the vulnerabilities and removed the useless services.

4.2.1 Additional Security Objectives

All security objectives for the environment listed in this section are added in this ST.

OE. SSL

The TOE uses the SSL protocol for ensuring the secure communication which is provided by IT environment.

OE. TIME

The TOE uses the reliable timestamp which is provided by IT environment.

5 IT Security Requirements

This part of the ST defines the detailed functional and assurance security requirements that shall be satisfied by the TOE or its environment.

5.1 TOE Security Functional Requirements

All of the following security functional requirements are taken from the "Label-based Access Control System Protection Profile for Government", version 1.1 [LSAPP].

The claimed minimum strength of function (SOF) for this TOE is SOF-medium.

[Table 5-1] shows the summary of Security Functional Components.

[Table 5-1] Security Functional Requirements

Class	Functional Component	
Security Audit (FAU)	FAU_ARP.1	Security alarms
	FAU_GEN.1	Audit data generation
	FAU_GEN.2	User identity association
	FAU_SAA.1	Potential violation analysis
	FAU_SAA.3	Simple attack heuristics
	FAU_SAR.1	Audit review
	FAU_SAR.2	Restricted audit review
	FAU_SAR.3	Selectable audit review
	FAU_SEL.1	Selective audit
	FAU_STG.1	Protected audit trail storage
	FAU_STG.3	Action in case of possible audit data loss
	FAU_STG.4	Prevention of audit data loss
User Data Protection (FDP)	FDP_ACC.1	Subset access control
	FDP_ACF.1	Security attribute based access control
	FDP_IFC.1	Subset information flow control
	FDP_IFF.2	Hierarchical security attributes

	FDP_ITC.1	Import of user data without security attributes
	FDP_RIP.1	Subset residual information protection
Identification and authentication (FIA)	FIA_AFL.1	Authentication failure handling
	FIA_ATD.1	User attribute definition
	FIA_SOS.1	Verification of secrets
	FIA_UAU.1	Timing of authentication
	FIA_UAU.4	Single-use authentication mechanisms
	FIA_UAU.7	Protected authentication feedback
	FIA_UID.2	User identification before any action
	FIA_USB.1	User-subject binding
Security management (FMT)	FMT_MOF.1	Management of security functions behavior
	FMT_MSA.1(1)	Management of security attributes (DAC)
	FMT_MSA.1(2)	Management of security attributes (MAC)
	FMT_MSA.3(1)	Static attribute initialization (DAC)
	FMT_MSA.3(2)	Static attribute initialization (MAC)
	FMT_MTD.1(1)	Management of TSF data (Audit data)
	FMT_MTD.1(2)	Management of TSF data (Identification and authentication data)
	FMT_MTD.1(3)	Management of TSF data (authentication data)
	FMT_MTD.1(4)	Management of TSF data (TSF data)
	FMT_REV.1(1)	Revocation (User)
	FMT_REV.1(2)	Revocation (Object)
	FMT_SMF.1	Specification of management functions
	FMT_SMR.1	Security roles
Protection of the TSF (FPT)	FPT_AMT.1	Abstract machine testing
	FPT_ITT.1	Basic internal TSF data transfer protection
	FPT_RVM.1	Non-bypassability of the TSP
	FPT_SEP.1	TSF domain separation
	FPT_STM.1	Reliable time stamps
	FPT_TST.1	TSF testing
TOE access (FTA)	FTA_SSL.1	TSF-initiated session locking
Trusted path/channels	FTP_ITC.1	Inter-TSF trusted channel

(FTP)		

5.1.1 Security Audit (FAU)

FAU_ARP.1 Security Alarms

Hierarchical to: No other components

FAU_ARP.1.1 The TSF shall take [*list of the least disruptive actions*] upon detection of a potential security violation.

a) [Compulsory kill of the violated subject process

b) Alarm RedCastle ESM in real-time

c) Inform to Authorized Administrator by e-mail]

Dependencies: FAU_SAA.1 Potential violation analysis

FAU_GEN.1 Audit data generation

Hierarchical to: No other components

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

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

b) All auditable events for the *not specified* level of audit; and

c) [Successful events in case of bypass the enforcement of Mandatory Access Control,

d) Rejected events when the option of the enforcement of MAC is “Warning”,

e) Auditable Events of [Table 5-2] and [Table 5-3]]

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

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

g) For each audit event type, base on the auditable event definitions of the functional components included in the PP/ST, [Auditable Events of [Table 5-2] and [Table 5-3]]

Dependencies: FPT_STM.1 Reliable time stamps

[Table 5-2] Auditable Events

Component	Event	Details
FAU_ARP.1	Actions taken due to imminent security violations	-
FAU_SAA.1	Enabling and disabling of any of the analysis mechanisms, Automated responses performed by the tool	-
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating	-
FDP_ACF.1	Successful requests to perform an operation on an object covered by the SFP	The identity of the object
FDP_IFF.2	Decisions to permit requested information flows	Security attributes of subject and object
FDP_ITC.1	Successful import of user data, including any security attributes	-
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_SOS.1	Rejection by the TSF of any tested secret	-
FIA_UAU.1	All use of the authentication mechanism	-
FIA_UAU.4	Attempts to reuse authentication data	
FIA_UID.2	Unsuccessful use of the user identification	-

	mechanism, including the user identity provided	
FIA_USB.1	Unsuccessful binding of user security attributes to a subject (e.g. creation of a subject)	-
FMT_MOF.1	All modifications in the behavior of the functions in the TSF	-
FMT_MSA.1	All modifications of the values of security attributes	The modified security attributes
FMT_MTD.1	All modifications to the limits on TSF data	The modified TSF data
FMT_REV.1	All attempts to revoke security attributes	-
FMT_SMF.1	Use of management functions	-
FMT_SMR.1	Modifications to the group of users that are part of a role	-
FPT_STM.1	Changes to the time	-
FPT_TST.1	Execution of the TSF self tests and the result of the tests	The name of modified TSF data
FTA_SSL.1	Locking of an interactive session by the session locking mechanism, Successful unlocking of an interactive session	-
FTP_ITC.1	Failure of the trusted channel functions, Identification of the initiator and target of failed trusted channel functions	-

[Table 5-3] Additional Auditable Events

Component	Event	Details
FAU_SAA.3	Enabling and disabling of any of the analysis mechanisms, Automated responses performed by the tool	Identity of authorized administrator performs actions
FAU_STG.3	Actions taken due to exceeding of a threshold	-
FAU_STG.4	Actions taken due to the audit storage	-

	failure	

FAU_GEN.2 User identity association

Hierarchical to: No other components

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

Dependencies: FAU_GEN.1 Audit data generation

FIA_UID.1 Timing of identification

FAU_SAA.1 Potential violation analysis

Hierarchical to: No other components

FAU_SAA.1.1 The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the TSP.

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

a) Accumulation or combination of [Failure event among the auditable events of FIA_UAU.1, Rejected event among the auditable events of FDP_ACF.1, Violated event among the auditable events of FPT_TST.1] ;

b) [Accumulation of the Simple Attack Heuristics violation]

Dependencies: FAU_GEN.1 Audit data generation

FAU_SAA.3 Simple attack heuristics

Hierarchical to: FAU_SAA.1

FAU_SAA.3.1 The TSF shall be able to maintain an internal representation of the following signature events [*a subset of system events as below*] that may indicate a violation of the TSP.

a) [Attack on the vulnerability of Symbolic-link

b) An illegal access to FIFO special file

c) An illegal attempts to create Hard-link file

d) Attack on the vulnerability of CHROOT

e) An attempts to switch the promiscuous mode]

FAU_SAA.3.2 The TSF shall be able to compare the signature events against the record of system activity discernible from an examination of [the information to be used to determine system activity].

FAU_SAA.3.3 The TSF shall be able to indicate an imminent violation of the TSP when a system event is found to match a signature event that indicates a potential violation of the TSP.

Dependencies: No dependencies

FAU_SAR.1 Audit review

Hierarchical to: No other components

FAU_SAR.1.1 The TSF shall provide [*authorized administrator*] with the capability to read [*all audit information*] from the audit records.

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

Dependencies: FAU_GEN.1 Audit data generation

FAU_SAR.2 Restricted audit review

Hierarchical to: No other components

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

Dependencies: FAU_SAR.1 Audit review

FAU_SAR.3 Selectable audit review

Hierarchical to: No other components

FAU_SAR.3.1 The TSF shall provide the ability to perform *searches, sorting* of audit data based on [the following attribute].

a) User identity

b) Object identity

c) Security label of subject

d) Security label of object

e) Period of the audited time

f) Event type

Dependencies: FAU_SAR.1 Audit review

FAU_SEL.1 Selective audit

Hierarchical to: No other components

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

a) Object identity, user identity

b) [security label of subject

c) Security label of object

d) Period of the audited time

e) Event type]

Dependencies: FAU_GEN.1 Audit data generation

FMT_MTD.1 Management of TSF data

FAU_STG.1 Protected audit trail storage

Hierarchical to: No other components

FAU_STG.1.1 The TSF shall protect the stored audit records from unauthorized deletion.

FAU_STG.1.2 The TSF shall be able to ***<u>prevent</u>*** unauthorized modifications to the stored audit records in the audit trail.

Dependencies: FAU_GEN.1 Audit data generation

FAU_STG.3 Action in case of possible audit data loss

Hierarchical to: No other components

FAU_STG.3.1 The TSF shall take [the following actions] if the audit trail exceeds [the audit space (= file counts x file size) defined by the authorized administrator, the default values are 5 files and 10MB per file].

a) [generate an alarm by e-mail to the authorized administrator if the audit trail exceeds 80% of the defined audit space.

b) Generate an alarm by e-mail to the authorized administrator for the increases at an interval of 5%.

c) If the audit trail is full, sending an alarm by e-mail to the authorized administrator and acting a function of FAU_STG.4 Prevention of audit data loss]

Dependencies: FAU_STG.1 Protected audit trail storage

FAU_STG.4 Prevention of audit data loss

Hierarchical to: FAU_STG.3

FAU_STG.4.1 The TSF shall prevent auditable events, except those taken by the authorized user with special rights and [take the following actions to be taken in case of audit storage failure] if the audit trail is full.

a) [As soon as the audit trail is full, deferring the TSF to be called by all users except the authorized administrator

b) The TSF shall be resumed by an authorized administrator after the backup of audit trail]

Dependencies: FAU_STG.1 Protected audit trail storage

5.1.2 User Data Protection (FDP)

FDP_ACC.1 Subset access control

Hierarchical to: No other components

FDP_ACC.1.1 The TSF shall enforce the [Discretionary Access Control Policy] on [all processes], [the following list of objects and the following list of operations among subjects and objects covered by SFP].

a) [list of objects: file system objects of the Linux Operating System

b) List of operations among subjects and objects

i) read,
ii) write,
iii) execute,
iv) create,

v) delete,
vi) rename,
vii) chmod,
viii) chown]

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attributes based access control

Hierarchical to: No other components

FDP_ACF.1.1 The TSF shall enforce the [Discretionary Access Control Policy] to objects based on [the following attributes].

a) [user identity associated with a subject

b) Group membership associated with a subject

c) process

d) [the list of DAC attributes related the following:

i) Enforce the allowance or denial of operation based on user identity
ii) Enforce the allowance or denial of operation based on group membership
iii) Enforce the allowance or denial of operation based on process
iv) The default value of the allowance or denial of operation is [denial]

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

[rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects:

a) For each operation, if the identity of a user (subject) is specified in the attributes of the access control rules, this operation is allowed. Otherwise it is denied.

b) For each operation, if the group membership of user (subject) is specified in the attributes of the access control rules, this operation is allowed. Otherwise it is denied.

c) For each operation, if the name of process (subject) is specified in the attributes of the access control rules, the operation is allowed. Otherwise it is denied.

d) If an operation among controlled subjects and controlled objects is specified in the attributes of the access control rules, this operation is allowed. Otherwise it is denied.]

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

[rules, based on security attributes, that explicitly authorize access of subjects to objects

a) If the security attribute of user is an authorized administrator (Security Officer, SO), the TSF shall explicitly authorize access of subjects to objects.

b) If an operation which is called is not for an operation among controlled subjects and controlled objects, the TSF shall provide the additional rules associated with the following:

  v) The allowed rule for setuid operation
  vi) The allowed rule for su operation

c) If an operation which is called is not for an operation among controlled subjects and controlled objects, the TSF shall provide the additional rules to be allowed associated with the following:

  i) For setuid operation, if the name of setuid program, which is executed by subject, is specified in the allowed rule of setuid operation, executing setuid program is allowed.
  ii) For su operation, while the right of subject is escalated to root by su program, if the identity of subject is specified in the allowed rule of su operation, su is allowed.]

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the [following rules, based on security attributes, that explicitly deny access of subjects to objects].

a) [If the security attribute of object is SO, the TSF shall explicitly deny access of subjects to objects except that the security attribute of subject is SO.

b) If an operation which is called is not for an operation among controlled subjects and controlled objects, the TSF shall provide the additional rules to be denied associated with the following

   i) The denied rule of the restricting command execution
   ii) The denied rule of the controlling kill signal

c) If an operation which is called is not for an operation among controlled subjects and controlled objects, the TSF shall provide the additional rules to be allowed associated with the following:

   i) For the command execution operation, if the name of command is specified in the denied rule of the restricting command execution, executing command is denied.
   ii) For kill operation, if the name of process is specified in the denied rule of the controlling kill signal, sending a signal to that process is denied]

Dependencies: FDP_ACC.1 Subset access control

FMT_MSA.3 Static attribute initialization

FDP_IFC.1 Subset information flow control

Hierarchical to: No other components

FDP_IFC.1.1 The TSF shall enforce the [Mandatory Access Control Policy] on [all subjects], [list of objects and list of operations among subject and subject, and among subject and objects.

a) [list of objects]

i) Operation among subject and subject: Subjects are user or process in the Linux operating system.

ii) Operation among subject and object: Objects are user, process, or file in the Linux operating system.

b) List of operations among subject and subject

i) Write operation: kill

c) List of operations among subject and object

i) Read operations: read, execute

ii) Write operations: write, delete, create

Dependencies: FDP_IFF.1 Simple security attributes

FDP_IFF.2 Hierarchical security attributes

Hierarchical to: FDP_IFF.1

FDP_IFF.2.1 The TSF shall enforce the [Mandatory Access Control Policy] base on the following types of subject and information security attributes:

a) [Security Label of subject

b) Security Label of object]

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) [For read operation, if the security attribute of subject is greater than the one of object, a subject can read the information of object.

b) For write operation, if the security attributes of subject and object are equal, the information of subject can be written to object.

c) If the security attribute of subject A is greater than the one of subject B, the information of subject B flows to subject A.]

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

FDP_IFF.2.4 The TSF shall provide the following [list of additional SFP capabilities].

FDP_IFF.2.5 The TSF shall explicitly authorize an information flow based on the following rules: [rules based on security attributes, that explicitly authorize information flows].

a) [provides the rules that authorize the information flows based on the security attributes of users, such as SO(Security Officer), SA(System Administrator), MU(Multi-Label Security User).

b) If the security attribute of subject is SO, the TSF shall explicitly authorize the access of subject to object.

c) If the SO executes the file which has the SA attribute, the TSF assign SA to a process.

d) The TSF shall allow to execute and re-assign the security attribute for the commands registered in the list of the command to be allowed

e) The TSF shall allow the transition user who is SO or SA to system root by the su command.

f) If the subject has the security attribute access to the non-labeled object, the TSF shall explicitly allow the access to object.]

FDP_IFF.2.6 The TSF shall explicitly deny an information flow based on the following rules: [rules based on security attributes, that explicitly deny information flows].

a) [provides the rules that deny the information flows based on the security attributes of users, such as SO, SA, MU

b) If the security attribute of object is SO, the TSF shall explicitly deny the access of subject except SO to object.

c) The SA subject cannot access to the MU object.

d) The MU subject cannot access to the SA object]

FDP_IFF.2.7 The TSF shall enforce the following relationships for any two valid information flow control security attributes:

a) There is an ordering function that, given two valid security attributes, determines:

   i) if the security attributes are equal,
   ii) if one security attribute is greater than the other, or
   iii) if the security attributes are incomparable

b) There is a “least upper bound (LUB)” in the set of security attributes, such that, given any two valid security attributes, there is a valid security attribute that is greater than or equal to the two valid security attributes;

c) There is a “greater lower bound (GLB)”, in the set of security attributes, such that, given any two valid security attributes, there is a valid security attribute that is not greater than the two valid security attributes.

Dependencies: FDP_IFC.1 Subset information flow control

FMT_MSA.3 Static attribute initialization

FDP_ITC.1 Import of user data without security attributes

Hierarchical to: No other components

FDP_ITC.1.1 The TSF shall enforce the [Mandatory Access Control Policy] when importing user data, controlled under the **Mandatory Access Control Policy**, from outside of the TSC.

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

FDP_ITC.1.3 The TSF shall enforce the [following additional importation control rules] when importing user data controlled under the Mandatory Access Control Policy from outside the TSC:

a) [An authorized administrator shall specify the security attribute of the user data imported, it is based on the identity of the subject importing data.

b) An authorized administrator shall specify the security attribute of the user data imported, it is based on the security attribute of subject importing data.]

Dependencies: [FDP_IFC.1 Subset information flow control]

FMT_MSA.3 Static attribute initialization

FDP_RIP.1 Subset residual information protection

Hierarchical to: No other components

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the *deallocation of the resource from* the following objects: [files].

Dependencies: No dependencies

5.1.3 Identification and authentication (FIA)

FIA_AFL.1 Authentication failure handling

Hierarchical to: No other components

FIA_AFL.1.1 The TSF shall detect when [5] unsuccessful authentication attempts occur related to [all authentication events].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [disable the account until unlocked by the authorized administrator].

Dependencies: FIA_UAU.1 Timing of authentication

FIA_ATD.1 User attributes definition

Hierarchical to: No other components

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

a) [user identifier

b) Group memberships

c) Security Label

d) Authentication Data

e) Security-relevant Roles]

Dependencies: No dependencies

FIA_SOS.1 Verification of secrets

Hierarchical to: No other components

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet [the following defined quality metric].

a) [Password length: between 8 and 15 characters

b) Acceptable password characters:

  i) 52 alphabetic letters (lowercase or uppercase)

  ii) 10 numeric digits (0-9)

  iii) 16 special characters (!, @, #, $, %, ^, &, *, (,), +, =, <, >, :, ;)

c) A password must contain at least one character of alphabetic, numeric, and special characters.

d) Allow to use the consecutive alphabetic or numeric

e) Allow to use the repetition of characters]

Dependencies: No dependencies

Application Notes: Examples of the defined quality metric could include minimum length, mixing rule, or change period for password authentication mechanism.

FIA_UAU.1 Timing of authentication

Hierarchical to: No other components

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

a) [Control of the Manager (RedCastle ESM) connections: whether a connection to Agent on the basis of a Manager's IP address is allow or not.]

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Dependencies: FIA_UID.1 Timing of identification

Application Notes: The user must be authenticated before any action and there must be no the TSF-mediated actions before authentication. But, in case of requiring the access right of an authorized administrator, the TSF-mediated actions are needed.

FIA_UAU.4 Single-use authentication mechanisms

Hierarchical to: No other components

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to [the following authentication mechanism] .

a) [Secure OS Authentication mechanism]

Dependencies: No dependencies

FIA_UAU.7 Protected authentication feedback

Hierarchical to: No other components

FIA_UAU.7.1 The TSF shall provide only [‘*’ or space] to the user while the authentication is in progress.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_UID.2 User identification before any action

Hierarchical to: FIA_UID.1

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

FIA_USB.1 User-subject binding

Hierarchical to: No other components

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

a) [the user identity

b) Security label

c) Security roles]

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) [The security attribute of subject acting on the behalf of users shall assign based on the following security attribute of user:

i) The subject identity based on the user

ii) The subject security attributes based on the user

iii) The security role status of user identity and security roles

b) If the subject identity acting on the behalf of users would change, the role status shall be changed on the basis of the user identity.]

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) [If the security attribute acting on the behalf of users had changed, the security attribute of subject cannot be changed.]

Dependencies: FIA_ATD.1 User attributes definition

5.1.4 Security Management (FMT)

FMT_MOF.1 Management of security functions behavior

Hierarchical to: No other components

FMT_MOF.1.1 The TSF shall restrict the ability to *disable, enable* the functions [as below] to [the authorized administrator].

a) [Security management functions

b) Audit functions

c) Access Control functions]

Dependencies: FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

FMT_MSA.1(1) Management of security attributes

Hierarchical to: No other components

FMT_MSA.1.1 The TSF shall enforce the [DAC Policy] to restrict the ability to *change, query, and modify the default value of* the security attributes [DAC policy associated with objects] to [authorized administrator(SO), the owner of object].

Dependencies: [FDP_ACC.1 Subset access control]

FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

FMT_MSA.1(2) Management of security attributes

Hierarchical to: No other components

FMT_MSA.1.1 The TSF shall enforce the [Mandatory Access Control Policy] to restrict the ability to *change, query, and modify the default value of* the security attributes [MAC policy associated with subjects or objects] to [authorized administrator (SO)].

Dependencies: [FDP_IFC.1 Subset information flow control]

FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

FMT_MSA.3(1) Static attribute initialization

Hierarchical to: No other components

FMT_MSA.3.1 The TSF shall enforce the [DAC Policy] to provide *restrictive* default values for security attributes that are used to enforce the **DAC policy**.

FMT_MSA.3.2 The TSF shall allow [the authorized administrator(SO)] to specify alternative initial values to override the default values when an object or information created.

Dependencies: FMT_MSA.1 Management of security attributes

FMT_SMR.1 Security roles

FMT_MSA.3(2) Static attribute initialization

Hierarchical to: No other components

FMT_MSA.3.1 The TSF shall enforce the [MAC Policy] to provide *restrictive* default values for security attributes that are used to enforce the **MAC policy**.

FMT_MSA.3.2 The TSF shall allow [the authorized administrator(SO)] to specify alternative initial values to override the default values when an object or information created.

Dependencies: FMT_MSA.1 Management of security attributes

FMT_SMR.1 Security roles

FMT_MTD.1(1) Management of TSF data

Hierarchical to: No other components

FMT_MTD.1.1 The TSF shall restrict the ability to change *default, query, delete, and clear* the [audit data] to [the authorized administrator (SO)].

Dependencies: FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

FMT_MTD.1(2) Management of TSF data

Hierarchical to: No other components

FMT_MTD.1.1 The TSF shall restrict the ability to *delete, [initialize]* the [identification and authentication data] to [the authorized administrator (SO)].

Dependencies: FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

FMT_MTD.1(3) Management of TSF data

Hierarchical to: No other components

FMT_MTD.1.1 The TSF shall restrict the ability to *modify* the [authentication data] to [the authorized administrator (SO) or the owner of authentication data].

Dependencies: FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

FMT_MTD.1(4) Management of TSF data

Hierarchical to: No other components

FMT_MTD.1.1 The TSF shall restrict the ability to change *default, query, delete, clear, [create]* the [TSF data associated with security] to [the authorized administrator (SO)].

Dependencies: FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

FMT_REV.1(1) Revocation

Hierarchical to: No other components

FMT_REV.1.1 The TSF shall restrict the ability to revoke security attributes associated with the *users* within the TSC to [the authorized administrator (SO)].

FMT_REV.1.2 The TSF shall enforce the rules [as below]:

a) [The authentication data associated with the security must be revoked immediately.]

Dependencies: FMT_SMR.1 Security roles

FMT_REV.1(2) Revocation

Hierarchical to: No other components

FMT_REV.1.1 The TSF shall restrict the ability to revoke security attributes associated with the *objects* within the TSC to [the authorized administrator (SO)].

FMT_REV.1.2 The TSF shall enforce the rules [as below]:

a) [The access right associated with object must be revoked immediately when the access to object is verified.]

Dependencies: FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components

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

[list of security management functions to be provided by the TSF

a) Startup or stop of the security functions

b) Management of the security category

c) Management of the security attributes of user

d) Management of the security attributes of object

e) Management of the security attributes of subject(processes)

f) Management of the ACL policies

g) Management of the allow/deny policies

h) Configuration of the audit functions

i) Management of the ESM administrators

j) Management of the Secure OS Authentication

k) Management of the file integrity functions]

Dependencies: No dependencies

FMT_SMR.1 Security roles

Hierarchical to: No other components

FMT_SMR.1.1 The TSF shall maintain the roles [as below].

l) [the authorized administrator (SO),

m) The owner of the authentication data]

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

Dependencies: FIA_UID.1 Timing of identification

5.1.5 Protection of the TSF (FPT)

FPT_AMT.1 Abstract machine testing

Hierarchical to: No other components

FPT_AMT.1.1 The TSF shall run a suite of tests *during initial start-up, periodically during normal operation, at the request of an authorized user* to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF.

Dependencies: No dependencies

FPT_ITT.1 Basic internal TSF data transfer protection

Hierarchical to: No other components

FPT_ITT.1.1 The TSF shall protect TSF data from [disclosure, modification] when it is transmitted between separate parts of the TOE.

Dependencies: No dependencies

FPT_RVM.1 Non-bypassability of the TSP

Hierarchical to: No other components

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

Dependencies: No dependencies

FPT_SEP.1 TSF domain separation

Hierarchical to: No other components

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

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

Dependencies: No dependencies

FPT_STM.1 Reliable time stamps

Hierarchical to: No other components

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

Dependencies: No dependencies

Application Notes: This component is required only for ensuring the audit data create one after the other. So, the TOE does not implement this component but just use the time-stamp which is provided by the operating system.

FPT_TST.1 TSF testing

Hierarchical to: No other components

FPT_TST.1.1 The TSF shall run a suite of self tests *during initial start-up, periodically during normal operation, at the request of the authorized user* to demonstrate the correct operation of the TSF.

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

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

Dependencies: FPT_AMT.1 Abstract machine testing

5.1.6 TOE access (FTA)

FTA_SSL.1 TSF-initiated session locking

Hierarchical to: No other components

FTA_SSL.1.1 The TSF shall lock an interactive session after [time interval of administrator inactivity] by:

a) Clearing or overwriting display devices, making the current contents unreadable;

b) Disabling any activity of the user's data access/display devices other than unlocking the session.

FTA_SSL1.2 The TSF shall require the following events to occur prior to unlocking the session: [the authentication for the authorized administrator]

Dependencies: FIA_UAU.1 Timing of authentication

5.1.7 Trusted path/channels (FTP)

FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to: No other components

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 The TSF shall permit *the TSF* to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [the remote management functions].

Dependencies: No dependencies

Application Notes**:** The TOE implements the FPT_ITC.1 component instead of this component.

5.2 TOE Security Assurance Requirements

The assurance requirements of this ST are consist of the assurance components of Common Criteria Part 3, the target evaluation assurance level for the product is EAL3+. The augmented components in this ST are listed as below.

- ADV_IMP.2: Implementation of the TSF
- ADV_LLD.1: Descriptive low-level design

- ALC_TAT.1: Well-defined development tools
- ATE_DPT.2: Testing: low-level design
- AVA_VLA.2: Independent vulnerability analysis

[Table 5-4] Assurance Requirements

Class	Component	
Configuration management	ACM_CAP.3	Authorization controls
	ACM_SCP.1	TOE CM coverage
Delivery and operation	ADO_DEL.1	Delivery procedures
	ADO_IGS.1	Installation, generation, and start-up procedures
Development	ADV_FSP.1	Informal functional specification
	ADV_HLD.2	Security enforcing high-level design
	ADV_IMP.2	Implementation of the TSF
	ADV_LLD.1	Descriptive low-level design
	ADV_RCR.1	Informal correspondence demonstration
Guidance documents	AGD_ADM.1	Administrator guidance
	AGD_USR.1	User guidance
Life cycle support	ALC_DVS.1	Identification of security measures
	ALC_TAT.1	Well-defined development tools
Tests	ATE_COV.2	Analysis of coverage
	ATE_DPT.2	Testing: low-level design
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
Vulnerability assessment	AVA_MSU.1	Examination of guidance
	AVA_SOF.1	Strength of TOE security function evaluation
	AVA_VLA.2	Independent vulnerability analysis

5.3 Security Requirements for the IT Environment

The security requirements for the IT environment are as below.

FPT_ITT.1 Basic internal TSF data transfer protection

Hierarchical to: No other components

FPT_ITT.1.1 The TSF shall protect TSF data from [disclosure, modification] when it is transmitted between separate parts of the TOE.

Dependencies: No dependencies

FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to: No other components

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 The TSF shall permit *the TSF* to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [the remote management functions].

Dependencies: No dependencies

FPT_STM.1 Reliable time stamps

Hierarchical to: No other components

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

Dependencies: No dependencies

5.4 Strength of Function

The TOE has the password based security function for identification and authentication that is implemented by a probabilistic or permutation mechanism. The mechanism rated in the strength of function analysis is the password mechanism for user authentication. The strength claimed for this function is SOF-medium. The security functional requirements to be claimed the SOF-medium are the FIA_UAU.1 and FIA_UAU.4, and the security functions of this TOE to satisfy the SOF-medium are Auth.1, Auth.2 and Protect.4.

6 TOE Summary Specification

6.1 TOE Security Functions

This chapter describes the security functions of RedCastle that is subject to this evaluation.

6.1.1 Reference Monitor (Refer)

6.1.1.1 System Call Interception (Refer.1)

TSF provides the reference monitor function to ensure the calling and succeeding of TSP enforcing functions before each function in TSC (TSF Scope of Control) to be allowed its conduct.

In the TOE, the reference monitor function is provided with the following purposes.

- TSF provides the reference monitor function to ensure the calling and succeeding of TSP enforcing functions before each function in TSC (TSF Scope of Control) to be allowed its conduct.
- For the security policy and security function environment settings of RedCastle Secure OS, system call will be added for use in the TOE. The newly added system call will be used to collect audit record also in the audit recording subsystem.

Security Requirements to be satisfied: FPT_RVM.1

6.1.1.2 Security Module Separation (Refer.2)

By deleting the security module from the operating system’s kernel module list when the security module is loaded into kernel, the TSF will be protected from external interference and tampering by subjects that is not trusted.

This function is not provided with security module information not only to a unauthorized user but also to a security officer(SO), and SO can confirm loading availability and performing status of security module by GUI(Graphical User Interface) or CLI(Command Line Interface).

Security Requirements to be satisfied: FPT_SEP.1

6.1.2 Mandatory Access Control (Ac_mac)

6.1.2.1 Security Label Assignment (Ac_mac.1)

Before the TOE performs access control function, identification and security attribute assignment to the subject will be conducted first. As the information of subject in general LINUX system can be changed by su and setuid program, assignment security attribute to a subject based on its information can not be coherent when it tried at a time of applying mandatory access control (MAC). Therefore, to grant MAC policy, the information of subject should be kept coherently.

Before apply access control rule, the TSF must know identification and security attribution of subject and object. Therefore, in the subject security attribute assignment function, it tries to identify system information and security attribute of subject and object and if it is not identified properly, it can arrange security attributes according to the subject’s own character.

In this TOE, if a user does login or a new process is created, the procedure of subject identification and user's security attributes-subject connection will be performed as follows before the TSF allows any actions concerned.

a) After user login, the security attribute assigned to the user will not be assigned to the subject before the independent identification and authentication process conducted on it.

b) The identified and authenticated subject will be assigned the user's security attribute, and after that point, all the created new processes will inherit security attributes of parent processes.

c) The subject security attribute that is assigned to a process will be presented in [Role status, Security level, Security category] and each item is as follows.

- Security label of subject
  - i) Security Level
  - ii) Security Category
- Role status(Process status) of subject

Security Requirements to be satisfied: FDP_IFC.1, FDP_IFF.2, FIA_USB.1

6.1.2.2 Multi-Level based MAC (Ac_mac.2)

If all processes are identified and assigned its security attributes in the subject security attribute setting function, the multi-level based MAC functions will be performed.

MAC policy is method that limits access to the object based on access clearance about allowed level of information that is included in object and the access information of this.

MAC policy applies a lot of information required for strong protection between the classified system data and the user of each level. As MAC also can prohibit the information flow to the object of lower secret level, it can be defined as information flow-control policy. Access to data will be decided by mandatory policy through definition of security level which is assigned to the subject and object.

This TOE forces the multi-level based MAC base on type of following security attributes of subject and object.

a) Security label of subject

b) Security label of object

Security label of subject and object are as follows.

a) Security Category

b) Security Level(classification)

Security Requirements to be satisfied: FDP_IFC.1, FDP_IFF.2, FDP_ITC.1

6.1.2.3 Inheritance and Revocation (Ac_mac.3)

When an object is created, the object will inherit the subject's security attribute automatically and when it is deleted, the security attribute and information of object will be deleted through the function of object security attribute inheritance and revocation. This function assures the prevention of previous information abuse.

Security Requirements to be satisfied: FDP_ITC.1, FDP_RIP.1

6.1.3 Discretionary Access Control (Ac_dac)

6.1.3.1 ACL based DAC (Ac_dac.1)

System call which allowed in MAC is transmitted to DAC. DAC forces access control rules on the basis of ACL(Access Control List) according to identity of subject in DAC.

This TOE provides DAC on the basis of ACL(Access Control List) separately with DAC by permission bit supporting in OS.

Subject information that can be identified on ACL based DAC policy is as follows.

a) User identifier

    i) User ID
    ii) User name

b) Group memberships

    i) Security category

    ii) Security-relevant Roles

c) Subject program name

    i) Process name

ACL based DAC control the following operation.

a) read

b) write

c) execute

d) create

e) delete

f) rename

g) chmod

h) chown

Security Requirements to be satisfied: FDP_ACC.1, FDP_ACF.1

6.1.3.2 Allowed/Denied List based DAC (Ac_dac.2)

Allowed/Denied list based DAC is control according to allowed/denied list for operation to manage specially or for operation that is not applied to ACL.

Allowed/Denied list is separated as follows.

a) Policy based allowed list: If an act performed that is not in the allowed list, it will be denied.

b) Policy based denied list: If a user performs an act which is in the denied list, it will be denied unless a user is SO.

The TOE forces rules based on the following list.

a) The denied rule of the restricting command execution: If the command is specified in the denied rule of the restricting command execution, executing command will be denied.

b) The denied rule of the controlling kill signal: If the process is specified in the denied rule of the controlling kill signal, sending a signal to the process will be denied.

c) The allowed rule for setuid operation: If the setuid program to execute by subject is specified in the allowed rule of setuid operation, this setuid program execution will be allowed.

d) The allowed rule for su operation: while the right of subject is escalated to root by su program, if the identity of subject is specified in the allowed rule of su operation, this su program execution will be allowed.

The Allowed system call is returned to reference monitor in DAC, and if the original system call is called, DAC that provides in OS will be performed.

Security Requirements to be satisfied: FDP_ACC.1, FDP_ACF.1

6.1.4 Identification and Authentication (Auth)

There are three different kinds of Identification and authentication in the TOE ; user's identification and authentication to use ESM, SO's identification and authentication for connecting to SecureOS by ESM, and identification and authentication of labeled users to gain permission before access to file of security attribute.

6.1.4.1 ESM Authentication (Auth.1)

RedCastle ESM’s user will be classified as follows.

a) ESM administrator: A user which can connect to RedCastle Secure OS. It can add or delete ESM user.

b) ESM user: A user who can be connected to RedCastle SecureOS.

ESM administrator and user (ESM user hereinafter) can be connected to the RedCastle Secure OS through the ESM identification and authentication in the system that RedCastle ESM is installed.

In the RedCastle ESM's identification and authentication function, the access right to ESM will be decided through the authentication data verification by decryption of encrypted authentication data which is generated when an ESM user was registered based on the authentication information provided by an ESM user.

If five unsuccessful authentication attempts were occurred, repetition will be prohibited because the RedCastle ESM is killed compulsory.

Security Requirements to be satisfied: FIA_AFL.1, FIA_UAU.1, FIA_UAU.7

6.1.4.2 Secure OS Authentication (Auth.2)

The Secure OS Authentication provides following security function..

- SecureOS identification and authentication of SO by ESM.
- Identification and authentication of SecureOS user(SO, SA, MU).

(1) SecureOS identification and authentication through ESM

ESM user can connect by each server that RedCastle SecureOS is installed if succeed in login to console for administration in RedCastle ESM. SO must identify and authenticate about relevant SecureOS in case of connect to each server.

Before perform SecureOS identification and authentication through ESM, it provides the identifying functions whether administrator account and ESM IP can connect to the Secure OS.

Identification and authentication information, that is provided by SO through GUI for Secure OS identification and authentication, are as follows.

- System account(SO's ID)
- System account password
- Secure OS security password

The communication between RedCastle SecureOS and RedCastle ESM uses the SSL(Secure Socket Layer) version 3 protocol. The SSL protocol encodes data to secure the authentication information.

(2) Secure OS identification and authentication of user for login

All users identified in Secure OS will be permitted the access to the object which has its own security attribute after the completion of authentication process by security password. Users who must through the identification and authentication process are as follows and the identification and authentication process for system root and system user who has security level '0' will be rejected.

- Security Officer (SO)
- System Administrator (SA)
- Multi-Label Security User (MU)

This TOE provides CLI(Command Line Interface) for login user's identification and authentication process, and the information provided through CLI is as follows.

- User's security password

Security Requirements to be satisfied: FIA_AFL.1, FIA_UAU.1, FIA_UAU.4, FIA_UAU.7, FIA_UID.2

6.1.5 Security Audit (Audit)

6.1.5.1 Audit Generation and Collection (Audit.1)

The following audit data will be generated in the TOE and log daemon will collect and store this data.

- The generated security log in security management
- The generated kernel log in RedCastle kernel
- The generated log in system monitoring

Also, The TOE's log daemon provides the function to collect IP Filter log and system log located in the outside of the TOE and sore it in the specified position separately.

- IP Filter Log
- Login History Log
- Login Fail Log
- syslog

The security management and kernel module's security function of the TOE will generate the security logs for the audit targeting incidents and those logs are as follows; audit data of reference monitor, MAC, DAC, IP Filter events generated in kernel level, audit data of identification and authentication, security management, system monitoring, TSF protection generate in application level.

Also, the log daemon of Secure OS provides a function to collect system logs generated in the system and stores them in the same location with separate files.

Security Requirements to be satisfied: FAU_GEN.1, FAU_GEN.2, FAU_SEL.1, FPT_STM.1

6.1.5.2 Potential Violation Analysis (Audit.2)

The SO can define unit time and the accumulation frequency limit for potential violation analysis in collected audit data in the TOE. In collected audit data, if a reviewed security violation exceeds accumulation frequency per unit time, TSF detects and warns this.

Rules for potential security violation analysis are as follows.

- Unit time of potential security violation analysis
- limit frequency

A set of rules for potential security violation analysis are as follows.

- Violation of identification and authentication security policy
- Violation of access control rules
- Violation of other security policy

Security Requirements to be satisfied: FAU_ARP.1, FAU_SAA.1

6.1.5.3 Audit Storage Management (Audit.3)

Audit storage provides 50MB of file size and 5 files count in default value, and if this limit is exceeded, it will warn the administrator by registry e-mail, and perform audit data loss prevention function.

The TOE provides function for SO to configure audit storage environment.

Security Requirements to be satisfied: FAU_STG.1, FAU_STG.3, FAU_STG.4

6.1.5.4 Audit Review (Audit.4)

In this TOE, the audit record will be stored in the directory to which SO can access only and provided in the format of which SO is able to read. The audit data which can query and review in this TOE is as follows.

- Security Log
- System Log
- System Monitoring Log
- IP Filter Log

SO queries and reviews a selective audit data through GUI provided in RedCastle ESM and CLI provided in RedCastle SecureOS.

The SO can develop the report about the stored audit data by using GUI(Graphical User Interface) and CLI(Command Line Interface).

- Security violation statistics/specification report
- User/IP login analysis report
- System information report

Security Requirements to be satisfied: FAU_SAR.1, FAU_SAR.2, FAU_SAR.3

6.1.5.5 Simple Attack Prevention (Audit.5)

The Simple attack prevention security function is to deny specific abuse actions detected as a violation conducted by a user over resources such as files and has the following detailed.

- Prevent the vulnerability of symbolic-link

- Prevent an illegal access to FIFO special file
- Prevent an illegal creation of Hard-link file
- Prevent the vulnerability of CHROOT
- Prevent attempts to switch the promiscuous mode

Security Requirements to be satisfied: FAU_SAA.3

6.1.6 Security Administration (Admin)

6.1.6.1 Security Functions Management (Admin.1)

When OS is booting and shutdown, or for SO to perform exceptional operations, the TOE provides the functions as follows.

- Collection and storage of audit log
- Security kernel module
- IP Filter which provides system services

SO manages security functions through GUI provided in RedCastle ESM and CLI provided in RedCastle SecureOS. For initiating and terminating of security functions through GUI, communication function of RedCastle SecureOS must be in operation and should be keep its operation state even though the security functions are terminated.

Administrator is not possible to perform any security management functions such as security policy-making unless he starts security function. But identification and authentication processes are possible to perform.

Security Requirements to be satisfied: FMT_MOF.1, FMT_SMF.1, FMT_SMR.1

6.1.6.2 Hierarchical Category Management (Admin.2)

The security category corresponds to non-hierarchical attribute among security label of subject or object and SO is able to set this according to the character of organization.

The security category means category of subject or object that defined in MAC and will be configured reflecting organization's system area or department usually.

SO can manage the security category through GUI provided in RedCastle ESM and CLI provided in RedCastle Secure OS.

- Inquiry of the security category
- Addition of the security category
- Delete of the security category
- Rename of the security category
- Transition of the security category

Rule of the security category is assigned as follows when the security category is added.

- Security Officer(SO): Security category's ID is default 1(Security Admin), and SO's category can not be added or deleted.
- System Administrator(SA): Security category's ID is default 2(System Admins), and new SA's category is assigned a low category of existing SA's category. An added security category's ID will be assigned the value between 3 and 127 in order.
- Multi-Label Security User(MU): Security category's ID is not assign default value, and new MU's category is assigned a low category of SA's category or existing MU's category. An added security category's ID is assigned value between 3 and 127 in order.

Security Requirements to be satisfied: FMT_MSA.1(2), FMT_MSA.3(2), FMT_SMF.1, FMT_SMR.1

6.1.6.3 Labeled Users Management (Admin.3)

Labeled users in the TOE are consisted as follows.

- User identifier
- Group memberships
- Authentication data
- Security label
  - Security category
  - Security level
  - Security-relevant roles: assigned according to rules of security category

The function of labeled user management is able to set security label, and retrieve, change, and delete security attribute of labeled users. The TOE categorize user as follows.

- Labeled users: Security level is not 0. Labeled users will be categorized into security officer(SO), system administrator(SA), and multi-label security User(MU) according to its assigned security category.
- System user: Security level is 0 and security category is 0.
- System root: User who has corresponding UID to the super user in the existing OS is 0. Both security level and security category are 0.

User identifier and it's group identifier among labeled users' attributes will be used equally with the user identifier and group identifier of the OS. In the function of labeled user addition and modification, unsuccessful authentication attempts limit can be configured for each labeled users and its default set value is 5 times.

When SO adds a labeled user, security password will be registered as a user's system password before the first identification and authentication process will be conducted. And if security attribute of labeled user is deleted, its security password will be deleted too.

Security Requirements to be satisfied: FIA_ATD.1, FMT_MSA.1(2), FMT_MSA.3(2), FMT_MTD.1(2), FMT_REV.1(1), FMT_SMF.1, FMT_SMR.1

6.1.6.4 Labeled Objects Management (Admin.4)

In the TOE, the security attribute of file which is an object is consisted as follows.

- File Name
- Security Label
  - Security Category
  - Security Level
  - Security Role: it is selected based on the security category role

When a new file is created, the TOE is designed to inherit the security attributes of subject automatically and this function will be used also when it assigns specific security attribute by security officer (SO).

The management of the security attribute of object is able to set security label of the file which is the object, and also can refer, change, and cancel the security attributes of the file.

Security Requirements to be satisfied: FMT_MSA.1(2), FMT_MSA.3(2), FMT_REV.1(2), FMT_SMF.1, FMT_SMR.1

6.1.6.5 Labeled Processes Management (Admin.5)

In the TOE, the security attribute of process which is a subject is consisted as follows.

- Process id
- Owner Identifier
- Security Label
    - Security Category
    - Security Level
    - Security Role Status: it is selected based on the security category role and the identifier of owner

When a new process is created, the TOE is designed to assign the security attributes of subject automatically and this function will be used when security officer (SO) want to query the security attributes of process.

Security Requirements to be satisfied: FMT_MSA.1(2), FMT_SMF.1, FMT_SMR.1

6.1.6.6 ACL policies Management (Admin.6)

In the TOE, the security officer(SO) is able to configure following rule of discretionary access control by file or file group for ACL policies management.

a) Subject information : Default value - Any
- The owner of subject
- The security category of subject
- The security role status of subject

b) Subject program name : Default value - NULL

c) Operation : Default value – Allow access

- read
- write
- execute
- create
- delete
- rename
- chmod
- chown

SO manages ACL policies through GUI provided in RedCastle ESM and CLI provided in RedCastle SecureOS.

- Management of policy group: retrieve, add, delete, rename
- Management of individual file policy: retrieve, add, modify, delete
- Management of group file policy: retrieve, add, modify, delete

Security Requirements to be satisfied: FMT_MSA.1(1), FMT_MSA.3(1), FMT_SMF.1, FMT_SMR.1

6.1.6.7 Allowed/Denied List Management (Admin.7)

The security officer(SO) is able to add, search, and delete the following rules by GUI(Graphical User Interface) and CLI(Command Line Interface) for managing the policy that will allow or deny explicitly the access of subjects to objects based on the security attribute.

a) The denied rule of the restricting command execution

b) The denied rule of the controlling kill signal

c) The allowed rule for setuid operation

d) The allowed rule for su operation

e) The allowed rule for bypass of execution

Security Requirements to be satisfied: FMT_MSA.1(1), FMT_MSA.3(1), FMT_SMF.1, FMT_SMR.1

6.1.6.8 Audit Configuration (Admin.8)

A security officer(SO) in the TOE is provided functions that are able to configure path, file, and alarm of audit storage.

Security Requirements to be satisfied: FMT_MTD.1(1), FMT_SMF.1, FMT_SMR.1

6.1.6.9 ESM Users Management (Admin.9)

By using the GUI(Graphical User Interface) provided in RedCastle ESM, it is possible to add/delete ESM administrator and user, and to change own ESM password.

The first registered user into an ESM when it is initiated after its installation will have the authority to being an ESM administrator. And the ESM administrator can register and delete new ESM user afterward. Required information for ESM user registration is as follows.

- ESM user ID
- ESM user password

An authentication data will be generated combining ID with password and this will be encrypted using SEED and SHA-1 algorithm.

The password to be used for an ESM user registration or its changes must satisfy the following conditions.

- Password length: between 8 and 15 characters
- Acceptable password characters.
  - 52 alphabetic letters (lowercase or uppercase)
  - 10 numeric digits (0-9)
  - 16 special characters (!,@,#,$,%,^,&,*,(,),+,=,<,>,:,;)
- A password must contain at least one character of alphabetic, numeric, and special characters.
- Allow to use the consecutive alphabetic or numeric
- Allow to use the repetition of characters

Security Requirements to be satisfied: FIA_SOS.1, FMT_MTD.1(2), FMT_MTD.1(3), FMT_REV.1(1), FMT_SMF.1, FMT_SMR.1

6.1.6.10 Security Password Management (Admin.10)

This TOE provides function that can register and change authentication data, namely security password.

A registration and a change of security password are possible only related user. A security officer (SO) is possible to change a security password by GUI and CLI. A system administrator except SO and a multi-label security user (MU) are possible to change a security password of self.

If SO registers SA (system administrator) as MU in the labeled users management(Admin.3), a password of that user is initialized.

An authentication data is generated mixing ID and password and this is encrypted using SEED algorithm and SHA-1 algorithm.

When security password registered or changed, a password must satisfy following conditions.

- Password length: between 8 and 15 characters
- Acceptable password characters.
  - 52 alphabetic letters (lowercase or uppercase)
  - 10 numeric digits (0-9)
  - 16 special characters (!,@,#,$,%,^,&,*,(,),+,=,<,>,:,;)
- A password must contain at least one character of alphabetic, numeric, and special characters.
- Allow to use the consecutive alphabetic or numeric
- Allow to use the repetition of characters

Security Requirements to be satisfied: FIA_SOS.1, FMT_MTD.1(3), FMT_SMF.1, FMT_SMR.1

6.1.6.11 Security Functions Configuration (Admin.11)

After the security function activated, the TOE provides the following functions to the security Officer to configure the security operating environment by GUI and CLI.

- Multi-level based MAC : On, Warning
- ACL based DAC : On, Warning, Off
- Allowed/Denied list based DAC : On, Warning, Off
  - Command execution control
  - SETUID execution allowance
  - Process kill prevention list
  - su allowance setting
- Simple attack prevention : On, Warning, Off
  - An illegal access to FIFO file
  - An illegal access to Symbolic-link

  - An illegal creation of Hard-link
  - An Attack on the vulnerability of CHROOT
  - A detection switching the promiscuous mode: On, Off
- Security module hiding : On, Off
- Restriction of process attribute monitoring : On, Off

Security Requirements to be satisfied: FMT_MTD.1(4), FMT_SMF.1, FMT_SMR.1

6.1.6.12 System Services Management (Admin.12)

These functions provide the system services management's GUI(Graphical User Interface) and these are as follows.

- IP Filter
  - Inbound network connection control
- System Monitoring
  - System performance
  - Monitor process
  - Process limit(CPU)
  - Disk usage
- System Account Management
  - System account user and group add/modification/deletion
  - System account password policy configuration/modification

Security Requirements to be satisfied: FMT_SMF.1, FMT_SMR.1

6.1.7 TSF Protection (Protect)

6.1.7.1 Abstract machine testing (Protect.1)

The TOE provides abstract machine and TSF operation testing function to check operating system state and operation state of the TOE on the system.

Security Requirements to be satisfied: FMT_SMF.1, FPT_AMT.1, FPT_TST.1

6.1.7.2 Integrity Checking Functions (Protect.2)

For TSF's secure operation, the TOE provides integrity checking functions over TSF's execute file, TSF's data file, and files which administrator select.

Execution files of RedCastle ESM and RedCastle SecureOS is registered as a default file for integrity checking target and will not be deleted from the list.

In the case of a first integrity checking, the integrity checking value will be stored, and afterward created integrity value and stored integrity checking value will be compared. The TOE will use SHA-1 for integrity checking.

Security Requirements to be satisfied: FMT_MTD.1(4), FMT_SMF.1, FPT_TST.1

6.1.7.3 ESM Screen Saving (Protect.3)

The TSF can lock the interactive sessions of security officer(SO) while he is inactive and unlock the sessions through the identification and authentication of the administrator. The TOE provides a session locking of inactivation through the ESM Screen Saving function.

The screen saving function is applied to Windows system that RedCastle ESM is installed and the queuing time will be set by using GUI(Graphical User Interface). If the ESM user’s inactive state lasts more than the queuing time set in advance, then the Screen Saving function will start. The screen saving function can be unlocked only by the authentication of ESM user's password in the TOE.

Security Requirements to be satisfied: FMT_MTD.1(4), FMT_SMF.1, FTA_SSL.1

6.1.7.4 Secure Communication (Protect.4)

The TOE installed the communication Server in the RedCastle SecureOS and the communication Client in the RedCastle ESM.

The communication between RedCastle SecureOS and RedCastle ESM uses the SSL(Secure Socket Layer) version 3 protocol. The SSL protocol encodes data and therefore authentication information is secured by SSL protocol.

The secure communication server provides access control function to the RedCastle ESM by identifying whether it is connectable by using SO's account and ESM IP address for the Security Officer to manage RedCastle SecureOS.

Security Requirements to be satisfied: FIA_UAU.4, FPT_ITT.1, FTP_ITC.1

6.2 Assurance Measures

The following table provides an overview showing how the assurance measures of EAL3+ are met by RedCastle.

[Table 6-1] Mapping Assurance Requirements to Measures

Assurance Component		Measures
ACM_CAP.3	Authorization controls	Configuration management
ACM_SCP.1	TOE CM coverage	
ADO_DEL.1	Delivery procedures	Delivery procedures
ADO_IGS.1	Installation, generation, and start-up procedures	Installation guidance
ADV_FSP.1	Informal functional specification	Functional specification

ADV_HLD.2	Security enforcing high-level design	High Level Design
ADV_IMP.2	Implementation of the TSF	Implementation representation
ADV_LLD.1	Descriptive low-level design	Low Level Design
ADV_RCR.1	Informal correspondence demonstration	Correspondence information of the functional specification
AGD_ADM.1	Administrator guidance	Administrator guidance
AGD_USR.1	User guidance	User guidance
ALC_DVS.1	Identification of security measures	Life cycle support
ALC_TAT.1	Well-defined development tools	
ATE_COV.2	Analysis of coverage	Testing
ATE_DPT.2	Testing: low-level design	
ATE_FUN.1	Functional testing	
ATE_IND.2	Independent testing-sample	N/A(evaluator)
AVA_MSU.1	Examination of guidance	N/A(evaluator)
AVA_SOF.1	Strength of TOE security function evaluation	Vulnerability analysis
AVA_VLA.2	Independent vulnerability analysis	

7 PP claims

7.1 PP Reference

This Security Target claims conformance with the “Label-based Access Control System Protection Profile for Government v1.1, 2006-05-17 (LACSPP)”. This Protection Profile was developed by the “IT Security Certification Center (ITSCC)” of the National Intelligence Service (NIS) of Korea.

7.2 PP Refinements and Additions

7.2.1 Refinement of PP functional requirements

The security functional requirements of this ST are the same as the ones of the Protection Profile.

[Table 7-1] PP Refinements and Additions

Class	Component of Protection Profile		Refinement
Security Audit	Security alarms	FAU_ARP.1.1	Refinement
	Audit data generation	FAU_GEN.1.1 FAU_GEN.1.2	
	User identity association	FAU_GEN.2.1	
	Potential violation analysis	FAU_SAA.1.1 FAU_SAA.1.2	
	Simple attack heuristics	FAU.SAA.3.1 FAU.SAA.3.2 FAU.SAA.3.3	assignment assignment
	Audit review	FAU_SAR.1.1 FAU_SAR.1.2	
	Restricted audit review	FAU_SAR.2.1	
	Selectable audit review	FAU_SAR.3.1	Refinement
	Selective audit	FAU_SEL.1.1	Refinement

	Protected audit trail storage	FAU_STG.1.1 FAU_STG.1.2	
	Action in case of possible audit data loss	FAU_STG.3.1	Refinement
	Prevention of audit data loss	FAU_STG.4.1	Refinement
User Data Protection	Subset access control	FDP_ACC.1.1	assignment, Refinement
	Security attribute based access control	FDP_ACF.1.1 FDP_ACF.1.2 FDP_ACF.1.3 FDP_ACF.1.4	assignment, Refinement assignment, Refinement assignment, Refinement assignment, Refinement
	Subset information flow control	FDP_IFC.1.1	assignment, Refinement
	Hierarchical security attributes	FDP_IFF.2.1 FDP_IFF.2.2 FDP_IFF.2.3 FDP_IFF.2.4 FDP_IFF.2.5 FDP_IFF.2.6 FDP_IFF.2.7	Selection assignment, Refinement assignment, Refinement assignment, Refinement assignment, Refinement
	Import of user data without security attributes	FDP_ITC.1.1 FDP_ITC.1.2 FDP_ITC.1.3	Refinement
	Subset residual information protection	FDP_RIP.1.1	
Identification and authentication	Authentication failure handling	FIA_AFL.1.1 FIA_AFL.1.2	assignment, Refinement
	User attribute definition	FIA_ATD.1.1	Refinement

	Verification of secrets	FIA_SOS.1.1	assignment, Refinement
	Timing of authentication	FIA_UAU.1.1 FIA_UAU.1.2	
	Single-use authentication mechanisms	FIA_UAU.4.1	assignment, Refinement
	Protected authentication feedback	FIA_UAU.7.1	assignment, Refinement
	User identification before any action	FIA_UID.2.1	
	User-subject binding	FIA_USB.1.1 FIA_USB.1.2 FIA_USB.1.3	
Security Management	Management of security functions behaviour	FMT_MOF.1.1	assignment, Refinement Selection
	Management of security attributes (DAC)	FMT_MSA.1(1).1	
	Management of security attributes (MAC)	FMT_MSA.1(2).1	
	Static attribute initialization (DAC)	FMT_MSA.3(1).1 FMT_MSA.3(1).2	
	Static attribute initialization (MAC)	FMT_MSA.3(2).1 FMT_MSA.3(2).2	
	Management of TSF data (Audit data)	FMT_MTD.1(1).1	
	Management of TSF data (Identification and authentication data)	FMT_MTD.1(2).1	
	Management of TSF data (authentication data)	FMT_MTD.1(3).1	
	Management of TSF data (TSF data)	FMT_MTD.1(4).1	
	Revocation (User)	FMT_REV.1(1).1 FMT_REV.1(1).2	

	Revocation (Object)	FMT_REV.1(2).1 FMT_REV.1(2).2	
	Specification of management functions	FMT_SMF.1	assignment
	Security roles	FMT_SMR.1.1 FMT_SMR.1.2	Refinement
Protection of the TSF	Abstract machine testing	FPT_AMT.1.1	
	Basic internal TSF data transfer protection	FPT_ITT.1	Selection
	Non-bypassability of the TSP	FPT_RVM.1.1	
	TSF domain separation	FPT_SEP.1.1 FPT_SEP.1.2	
	Reliable time stamps	FPT_STM.1.1	
	TSF testing	FPT_TST.1.1 FPT_TST.1.2 FPT_TST.1.3	
TOE Access	TSF-initiated session locking	FTA_SSL.1.1 FTA_SSL.1.2	assignment, Refinement assignment, Refinement
Trusted path/channels	Inter-TSF trusted channel	FTA_ITC.1.1 FTA_ITC.1.2 FTA_ITC.1.3	

7.2.2 Additional Functional Requirements by PP

There are no additional security functional requirements and the requirements of this ST are the same as the ones of Protection Profile.

7.2.3 Additional Assurance Requirements by PP

There are no additional security assurance requirements and the requirements of this ST are the same as the ones of Protection Profile.

The assurance requirements of protection profile are adopted to satisfy the EAL3+ from the components of the Common Criteria, Part 3. And the Protection Profile requires the following additional assurance components

[Table 7-2] Additional assurance requirements

Class	Component	
Development	ADV_IMP.2	Implementation of the TSF
	ADV_LLD.1	Descriptive low-level design
Life Cycle Support	ALC_TAT.1	Well-defined development tools
Tests	ATE_DPT.2	Testing: low-level design
Vulnerability assessment	AVA_VLA.2	Independent vulnerability analysis

8 Rationale

This chapter presents the evidence used in the Security Target evaluation. This evidence supports the claims that the ST is a complete and cohesive set of requirements, that a conformant TOE would provide an effective set if IT security countermeasures within the security environment, and that the TOE summary specification addresses the requirements. The rationale also demonstrates that any PP conformance claims are valid.

8.1 Security Objectives Rationale

The security objectives rationale shall demonstrate that the stated security objectives are traceable to all of the aspects identified in the TOE security environment and are suitable to cover them.

8.1.1 Rationale of Security Objectives for TOE

[Table 8-1] Mapping Objectives to assumptions, threats, polices

Security Objectives Assumptions, Threats, policies	TOE Security Objectives									
	O.AUDIT	O.MAC	O.CODE	O.MANAGE	O.INTEGRITY	O.LABEL	O.IA	O.DAC	O.PROTECT	O.RESIDUE
T.CODE			X							
T.AUDIT	X									
T.INTEGRITY					X		X		X	
T.UAUTH							X			
T.BYPASS							X		X	
T.RESIDUE										X
P.ACCOUNTABLE	X									
P.MAC		X				X				
P.LABEL		X				X				

P.IA							X			
P.ADMIN				X						
P.CIPHER				X						
P.DAC								X		

O.AUDIT

Since the TOE provides the means of recording the security relevant actions of users and investgating any security relevant events, O.AUDIT is required to counter the threat **T.AUDIT** and to support the policy **P.ACCOUNTABLE.**

O.MAC

Since the TOE ensures the access to resources on the basis of the security label of subject and object, **O.MAC** is required to counter policy **P.MAC** and **P.LABEL**.

O.CODE

Since the source codes which are generated by developers must be inspected whether they have some defects or not, **O.CODE** is required to counter assumption **T.CODE**.

O.MANAGE

Since the TOS provides the means of managing securely by the administrative users, **O.MANAGE** is required to support **P.ADMIN** and **P.CIPHER** in the case of managing remotely.

O.INTEGRITY

Since the TOE must protect the TSF data or the reliable data from the unauthorized disclosure, modification, and deletion, **O.INTEGRITY** is required to counter **T.INTEGRITY**.

O.LABEL

Since the TOE must assign and revoke the security label of the subject and the object according to the organization access control policies, **O.LABEL** is required to counter **P.LABEL** and **P.MAC**.

O.IA

Since the TOE must identify a user uniquely and ensure that only authorized users gain access to the TOE and its resources, O.IA is required to counter **T.INTEGIRTY**, **T.UAUTH**, **T.BYPASS** and **P.IA.**

O.DAC

Since the TOE must enforce the access to resources on the basis of the identity of user or group, O.DAC is required to counter **P.DAC.**

O.PROTECT

Since the TOE must protect itself from the deactivation or the modification of the TOE, O.PROTECT is required to counter **T.INTEGRITY and T.BYPASS**.

O.RESIDUE

Since the TOE must ensure that any information contained in a protected resource is not released when the resource is recycled, O.RESIDUE is required to counter **T.RESIDUE**.

8.1.2 Rationale of Security Objectives for Environment

[Table 8-2] Mapping Objectives to assumptions, threats within Environment

Objectives	SO for TOE Environment

	OE.LOCATE	OE.ADMINS	OE.MANAGE	OE.PATCH	OE.SSL	OE.TIME
A.LOCATE	X					
A.ADMINS		X				
A.PATCH				X		
TE.MISUSE		X	X			
TE.INSTALL		X	X			
P.ADMIN			X			
A.SSL					X	
A.TIME						X

OE.LOCATE

Since the TOE shall be located within controlled access facilities and protected from unauthorized physical modification, **OE.LOCATE** is countered to **A.LOCATE**.

OE.ADMINS

Since it is ensured that one or more competent individuals who are assigned to manage the TOE are assumed not to be careless, willfully negligent or hostile, OE.ADMINS is countered to **TE.MISUSE, TE.INSTALL,** and **A.ADMINS**.

OE.MANAGE

Since the TOE must be installed securely, configured, managed, and used by an only authorized administrator, OE.MANAGE is countered to **TE.MISUSE**, **TE.INSTALL,** and **P.ADMIN**.

OE.PATCH

Since the operating system will be patched the vulnerabilities and removed the useless services before the installing the TOE, **OE.PATCH** is countered to **A.PATCH**.

OE.SSL

Since the TOE uses the SSL protocol for ensuring the secure communication which is provided by IT environment, **OE.SSL** is counter to **A.SSL**.

OE.TIME

Since the TOE uses the reliable timestamp which is provided by IT environment, **OE.TIME** is countered to **A.TIME**.

8.2 Security Requirements Rationale

The security requirements rationale shall demonstrate that the set of security requirements (TOE and environment) is suitable to meet and traceable to the security objectives.

8.2.1 Rationale for TOE Security Requirements

The following table shows how the security functional requirements map to the objectives defined for the TOE.

[Table 8-3] Mapping security requirements to objectives

Objectives Functions	O.AUDIT	O.MAC	O.CODE	O.MANAGE	O.INTEGIRTY	O.LABEL	O.IA	O.DAC	O.PROTECT	O.RESIDUE
FAU_ARP.1	X									
FAU_GEN.1	X									
FAU_GEN.2	X									
FAU_SAA.1	X									
FAU_SAA.3	X									

FAU_SAR.1	X									
FAU_SAR.2	X									
FAU_SAR.3	X									
FAU_SEL.1	X									
FAU_STG.1	X									
FAU_STG.3	X									
FAU_STG.4	X									
FDP_ACC.1								X		
FDP_ACF.1								X		
FDP_IFC.1		X								
FDP_IFF.2		X								
FDP_ITC.1		X								
FDP_RIP.1										X
FIA_AFL.1							X			
FIA_ATD.1							X			
FIA_SOS.1							X			
FIA_UAU.1				X	X		X			
FIA_UAU.4							X			
FIA_UAU.7							X			
FIA_UID.2				X	X		X			
FIA_USB.1							X			
FMT_MOF.1				X						
FMT_MSA.1(1)				X				X		
FMT_MSA.1(2)		X		X		X				
FMT_MSA.3(1)				X				X		
FMT_MSA.3(2)		X		X		X				
FMT_MTD.1(1)				X						
FMT_MTD.1(2)				X						
FMT_MTD.1(3)				X						
FMT_MTD.1(4)				X						
FMT_REV.1(1)				X		X				
FMT_REV.1(2)				X		X				
FMT_SMF.1				X						
FMT_SMR.1				X						

FPT_AMT.1					X				X	
FPT_ITT.1				X						
FPT_RVM.1									X	
FPT_SEP.1									X	
FPT_STM.1	X									
FPT_TST.1					X				X	
FTA_SSL.1					X				X	
FTP_ITC.1				X						

O.AUDIT

FAU_ARP.1 Security alarms

This component satisfies **O.AUDIT** because the TSF ensures the functions taking the list of the least disruptive actions upon detection of a potential security violation.

FAU_GEN.1 Audit data generation

This component satisfies **O.AUDIT** because the TSF ensures the functions generating an audit record for the auditable events.

FAU_GEN.2 User identity association

This component satisfies **O.AUDIT** because the TSF ensures the functions associating each auditable event with the identity of the user that caused the event.

FAU_SAA.1 Potential violation analysis

This component satisfies **O.AUDIT** because the TSF ensures the functions enforcing the rules for monitoring audited events and indicating a potential violation of the TSP.

FAU_SAA.3 Simple attack heuristics

This component satisfies **O.AUDIT** because the TSF ensures the functions detecting the signature events from the auditable events and indicating a potential violation.

FAU_SAR.1 Audit review

This component satisfies **O.AUDIT** because the TSF ensures the functions providing the authorized administrator the capability to read all audit information from the audit records.

FAU_SAR.2 Restricted audit review

This component satisfies **O.AUDIT** because the TSF ensures the functions prohibiting all users read access to the audit records except those the authorized administrators.

FAU_SAR.3 Selectable audit review

This component satisfies **O.AUDIT** because the TSF ensures the functions providing the ability to perform searching and sorting of audit data.

FAU_SEL.1 Selective audit

This component satisfies **O.AUDIT** because the TSF ensures the functions including or excluding auditable events from the set of audited events.

FAU_STG.1 Protected audit trail storage

This component satisfies **O.AUDIT** because the TSF ensures the functions protecting the stored audit records from the unauthorized deletion and modification.

FAU_STG.3 Action in case of possible audit data loss

This component satisfies **O.AUDIT** because the TSF ensures the functions taking the corresponding actions if the audit trail exceeds the audit space defined by the authorized administrator.

FAU_STG.4 Prevention of audit data loss

This component satisfies **O.AUDIT** because the TSF ensures the functions preventing auditable events and taking the corresponding actions if the audit trail is full.

FDP_ACC.1 Subset access control

This component satisfies **O.DAC** because the TSF ensures the functions enforcing the Discretionary Access Control Policy on all processes.

FDP_ACF.1 Security attributes based access control

This component satisfies **O.DAC** because the TSF ensures the functions enforcing the Discretionary Access Control Policy to objects based on attributes.

FDP_IFC.1 Subset information flow control

This component satisfies **O.MAC** because the TSF ensures the functions enforcing the Mandatory Access Control Policy on all subjects.

FDP_IFF.2 Hierarchical security attributes

This component satisfies **O.MAC** because the TSF ensures the functions enforcing the Mandatory Access Control Policy based on the security label of subject and object.

FDP_ITC.1 Import of user data without security attributes

This component satisfies **O.MAC** because the TSF ensures the functions enforcing the Mandatory Access Control Policy when importing user data from outside of the TSC.

FDP_RIP.1 Subset residual information protection

This component satisfies **O.RESIDUE** because the TSF ensures the functions ensuring that any previous information content of a resource is made unavailable.

FIA_AFL.1 Authentication failure handling

This component satisfies **O.IA** because the TSF ensures the functions detecting when 5 unsuccessful authentication attempts occur.

FIA_ATD.1 User attributes definition

This component satisfies **O.IA** because the TSF ensures the functions maintaining the list of security attributes belonging to individual users.

FIA_SOS.1 Verification of secrets

This component satisfies **O.IA** because the TSF ensures the functions providing a mechanism to verify that secrets meet the defined quality metric.

FIA_UAU.1 Timing of authentication

This component satisfies **O.IA** because the TSF ensures the functions allowing the list of TSF mediated actions on behalf of the user to be performed before the user is authenticated.

FIA_UAU.4 Single-use authentication mechanisms

This component satisfies **O.IA** because the TSF ensures the functions preventing reuse of authentication data related to the authentication mechanism.

FIA_UAU.7 Protected authentication feedback

This component satisfies **O.IA** because the TSF ensures the functions providing only ‘*’ or ‘space’ to the user while the authentication is in progress.

FIA_UID.2 User identification before any action

This component satisfies **O.MANAGE, O.INTEGRITY, and O.IA** because the TSF ensures the functions identifying each user before allowing any other TSF-mediated actions on behalf of that user.

FIA_USB.1 User-subject binding

This component satisfies **O.IA** because the TSF ensures the functions associating user security attributes with subjects acting on behalf of that user.

FMT_MOF.1 Management of security functions behavior

This component satisfies **O.MANAGE** because the TSF ensures to be managed the functions by the authorized administrator only.

FMT_MSA.1(1) Management of security attributes

This component satisfies **O.MANAGE and O.DAC** because the TSF ensures to be managed the DAC policy by the authorized administrator only.

FMT_MSA.1(2) Management of security attributes

This component satisfies **O.MANAGE, O.LABEL and O.MAC** because the TSF ensures to be managed the security label for the MAC policy by the authorized administrator only.

FMT_MSA.3(1) Static attribute initialization

This component satisfies **O.MANAGE and O.DAC** because the TSF ensures the functions providing the restrictive default values for security attributes to be used by enforcing the DAC policy.

FMT_MSA.3(2) Static attribute initialization

This component satisfies **O.MANAGE, O.LABEL and O.MAC** because the TSF ensures the functions providing the restrictive default values for security attributes to be applied by enforcing the MAC policy.

FMT_MTD.1(1) Management of TSF data

This component satisfies **O.MANAGE** because the TSF ensures the functions restricting the ability to change, query, delete, and clear the audit data to the authorized administrator only.

FMT_MTD.1(2) Management of TSF data

This component satisfies **O.MANAGE** because the TSF ensures the functions restricting the ability to delete, and initialize the identification and authentication data to the authorized administrator only.

FMT_MTD.1(3) Management of TSF data

This component satisfies **O.MANAGE** because the TSF ensures the functions restricting the ability to modify the authentication data to the authorized administrator or the owner of data.

FMT_MTD.1(4) Management of TSF data

This component satisfies **O.MANAGE** because the TSF ensures the functions restricting the ability to change, query, delete, clear, and create the TSF data associated with security to the authorized administrator only.

FMT_REV.1(1) Revocation

This component satisfies **O.MANAGE and O.LABEL** because the TSF ensures the functions restricting the ability to revoke security attributes associated with the users within the TSC to the authorized administrator only.

FMT_REV.1(2) Revocation

This component satisfies **O.MANAGE and O.LABEL** because the TSF ensures the functions restricting the ability to revoke security attributes associated with the objects within the TSC to the authorized administrator only.

FMT_SMF.1 Specification of Management Functions

This component satisfies **O.MANAGE** because the TSF ensures the management functions of security attributes, security function and so on.

FMT_SMR.1 Security roles

This component satisfies **O.MANAGE** because the TSF ensures the functions maintaining the user roles and associating users with roles.

FPT_AMT.1 Abstract machine testing

This component satisfies **O.INTEGRITY and O.PROTECT** because the TSF ensures the functions running a suit of tests to demonstrate the correct operation.

FPT_ITT.1 Basic internal TSF data transfer protection

This component satisfies **O.MANAGE** because the TSF ensures the functions protecting TSF data from disclosure and modification when it is transmitted between separate parts of the TOE.

FPT_RVM.1 Non-bypass ability of the TSP

This component satisfies **O.PROTECT** because the TSF ensures the functions invoking and succeeding the TSP enforcement functions.

FPT_SEP.1 TSF domain separation

This component satisfies **O.PROTECT** because the TSF ensures the functions maintaining a security domain for its own execution that protects it from interference and tampering by distrusted subjects.

FPT_STM.1 Reliable time stamps

This component satisfies **O.AUDIT** because the TSF ensures the functions providing reliable time stamps for its own use.

FPT_TST.1 TSF testing

This component satisfies **O.INTEGRITY and O.PROTECT** because the TSF ensures the functions running a suite of self tests to demonstrate the correct operation of the TSF and providing the authorized users with the capability to verify the integrity of the TSF data and executable code.

FTA_SSL.1 TSF-initiated session locking

This component satisfies **O.INTEGRITY and O.PROTECT** because the TSF ensures the functions locking an interactive session after time interval of administrator inactivity.

FTP_ITC.1 Inter-TSF trusted channel

This component satisfies **O.MANAGE** because the TSF ensures the functions providing a communication channel between itself and a remote trusted IT product and providing assured identification of its end points.

8.2.2 Rationale for TOE Assurance Requirements

The assurance requirements of this ST are consist of the one of Common Criteria Part 3, and the target evaluation assurance level for the product is EAL3+. The augmented components in this ST are listed as below.

- ADV_IMP.2 Implementation of the TSF
- ADV_LLD.1 Descriptive low-level design
- ALC_TAT.1 Well-defined development tools
- ATE_DPT.2 Testing: low-level design
- AVA_VLA.2 Independent vulnerability analysis

The component **ADV_IMP.2** and **ATE_DPT.2** are augmented by the Protection Profile because the security objective **O.CODE** is required.

For dependency with **ADV_IMP.2**, the component **ADV_LLD.1** and **ALC_TAT.1** are augmented by the Protection Profile.

The component **AVA_VLA.2** is augmented by the Protection Profile because the vulnerability analysis should be performed not only by developer but also by evaluator.

8.2.3 Rationale for the IT environment requirements

[Table 8-4] Mapping objectives for environment to requirements

Objectives Requirement	OE.SSL	OE.TIME
FPT_STM.1		X
FPT_ITT.1	X	
FTP_ITC.1	X	

FPT_STM.1 Reliable time stamps

This component satisfies **OE.TIME** because it is assumed that IT Environment (Operating System) provides the reliable timestamp, which is used by the TSF.

FPT_ITT.1 Basic internal TSF data transfer protection

This component satisfies **OE.SSL** because it is assumed that IT Environment (SSL protocol) provides the secure communication between separate parts of the TOE.

FTP_ITC.1 Inter-TSF trusted channel

This component satisfies **OE.SSL** because it is assumed that IT Environment (SSL protocol) provides the secure communication between separate parts of the TOE.

8.3 Rationale for Dependencies

8.3.1 Dependencies between security functions

The following table shows that the dependencies between the components of security functional requirements.

[Table 8-5] Dependencies between functional components

No	Functional Component	Dependencies	Reference
1	FAU_ARP.1	FAU_SAA.1	4
2	FAU_GEN.1	FPT_STM.1	44
3	FAU_GEN.2	FAU_GEN.1, FIA_UID.1	2, 25
4	FAU_SAA.1	FAU_GEN.1	2
5	FAU_SAA.3	-	-
6	FAU_SAR.1	FAU_GEN.1	2
7	FAU_SAR.2	FAU_SAR.1	6
8	FAU_SAR.3	FAU_SAR.1	6
9	FAU_SEL.1	FAU_GEN.1 FMT_MTD.1	2 33, 34, 35, 36
10	FAU_STG.1	FAU_GEN.1	2
11	FAU_STG.3	FAU_STG.1	10
12	FAU_STG.4	FAU_STG.1	10
13	FDP_ACC.1	FDP_ACF.1	14
14	FDP_ACF.1	FDP_ACC.1, FMT_MSA.3	13, 31

15	FDP_IFC.1	FDP_IFF.1	16
16	FDP_IFF.2	FDP_IFC.1, FMT_MSA.3	15, 32
17	FDP_ITC.1	[FDP_IFC.1], FMT_MSA.3	15, 32
18	FDP_RIP.1	-	-
19	FIA_AFL.1	FIA_UAU.1	22
20	FIA_ATD.1	-	-
21	FIA_SOS.1	-	-
22	FIA_UAU.1	FIA_UID.1	25
23	FIA_UAU.4	-	-
24	FIA_UAU.7	FIA_UAU.1	22
25	FIA_UID.2	-	-
26	FIA_USB.1	FIA_ATD.1	20
27	FMT_MOF.1(1)	FMT_SMF.1, FMT_SMR.1	39, 40
28	FMT_MOF.1(2)	FMT_SMF.1, FMT_SMR.1	39, 40
29	FMT_MSA.1(1)	[FDP_ACC.1], FMT_SMF.1, FMT_SMR.1	13 39, 40
30	FMT_MSA.1(2)	[FDP_IFC.1], FMT_SMF.1, FMT_SMR.1	15 39, 40
31	FMT_MSA.3(1)	FMT_MSA.1, FMT_SMR.1	29, 40
32	FMT_MSA.3(2)	FMT_MSA.1, FMT_SMR.1	30, 40
33	FMT_MTD.1(1)	FMT_SMF.1, FMT_SMR.1	39, 40
34	FMT_MTD.1(2)	FMT_SMF.1, FMT_SMR.1	39, 40
35	FMT_MTD.1(3)	FMT_SMF.1, FMT_SMR.1	39, 40
36	FMT_MTD.1(4)	FMT_SMF.1, FMT_SMR.1	39, 40
37	FMT_REV.1(1)	FMT_SMR.1	40
38	FMT_REV.1(2)	FMT_SMR.1	40
39	FMT_SMF.1	-	-
40	FMT_SMR.1	FIA_UID.1	25
41	FPT_AMT.1	-	-
42	FPT_RVM.1	-	-
43	FPT_SEP.1	-	-
44	FPT_STM.1	-	-
45	FPT_TST.1	FPT_AMT.1	41
46	FTA_SSL.1	FIA_UAU.1	22

47	FTP_ITC.1	-	-
48	FPT_ITT.1	-	-

8.3.2 Dependencies between assurance requirements

The following table shows that the dependencies between the components of security functional requirements.

[Table 8-6] Dependencies between assurance components

No	Assurance Component	Dependencies	Reference
1	ADV_IMP.2	ADV_LLD.1 ADV_RCR.1 ALC_TAT.1	2 EAL3 3
2	ADV_LLD.1	ADV_HLD.2 ADV_RCR.1	EAL3 EAL3
3	ALC_TAT.1	ADV_IMP.1	1
4	ATE_DPT.2	ADV_HLD.2 ADV_LLD.1 ATE_FUN.1	EAL3 2 EAL3
5	AVA_VLA.2	ADV_FSP.1 ADV_HLD.2 ADV_IMP.1 ADV_LLD.1 AGD_ADM.1 AGD_USR.1	EAL3 EAL3 1 2 EAL3 EAL3

8.4 TOE Summary Specification Rationale

8.4.1 Conformance to TOE Security Functions

The following table shows that the TOE security functions specified in the TOE summary specification.

[Table 8-7] TOE Security Functions

ID	Security Function	ID	Security Function
Refer.1	System Call Interception	Admin.2	Hierarchical Category Management
Refer.2	Security Module Separation	Admin.3	Labeled Users Management
Ac_mac.1	Security Label Assignment	Admin.4	Labeled Objects Management
Ac_mac.2	Multi-Level based MAC	Admin.5	Labeled Processes Management
Ac_mac.3	Inheritance and Revocation	Admin.6	ACL Policies Management
Ac_dac.1	ACL based DAC	Admin.7	Allowed/Denied List Management
Ac_dac.2	Allowed/Denied List based DAC	Admin.8	Audit Configuration
Auth.1	ESM Authentication	Admin.9	ESM Users Management
Auth.2	SecureOS Authentication	Admin.10	Security Password Management
Audit.1	Audit Generation and Collection	Admin.11	Security Functions Configuration
Audit.2	Potential Violation Analysis	Admin.12	System Services Management
Audit.3	Audit Storage Management	Protect.1	Abstract Machine Testing
Audit.4	Audit Review	Protect.2	Integrity Functions
Audit.5	Simple Attack Prevention	Protect.3	ESM Screen Saving
Admin.1	Security Functions Management	Protect.4	Secure Communication

The following table shows that the IT security functions, as specified in the TOE summary specification, meet all security functional requirements for the TOE and work together to satisfy the TOE security functional requirements.

[Table 8-8] Mapping TOE SFR to TOE SS

TOE Security Functional Requirement			TOE Summary Specification
Class	Component	Element	
Security Audit	FAU_ARP.1	FAU_ARP.1.1	Audit.2
	FAU_GEN.1	FAU_GEN.1.1	Audit.1
		FAU_GEN.1.2	Audit.1
	FAU_GEN.2	FAU_GEN.2.1	Audit.1
	FAU_SAA.1	FAU_SAA.1.1	Audit.2
		FAU_SAA.1.2	Audit.2
	FAU_SAA.3	FAU_SAA.3.1	Audit.5

		FAU_SAA.3.2 FAU_SAA.3.3	Audit.5 Audit.5
	FAU_SAR.1	FAU_SAR.1.1 FAU_SAR.1.2	Audit.4 Audit.4
	FAU_SAR.2	FAU_SAR.2.1	Audit.4
	FAU_SAR.3	FAU_SAR.3.1	Audit.4
	FAU_SEL.1	FAU_SEL.1.1	Audit.1
	FAU_STG.1	FAU_STG.1.1 FAU_STG.1.2	Audit.3 Audit.3
	FAU_STG.3	FAU_STG.3.1	Audit.3
	FAU_STG.4	FAU_STG.4.1	Audit.3
User Data Protection	FDP_ACC.1	FDP_ACC.1.1	Ac_dac.1, Ac_dac.2
	FDP_ACF.1	FDP_ACF.1.1 FDP_ACF.1.2 FDP_ACF.1.3 FDP_ACF.1.4	Ac_dac.1 Ac_dac.1 Ac_dac.2 Ac_dac.2
	FDP_IFC.1	FDP_IFC.1.1	Ac_mac.1, Ac_mac.2
	FDP_IFF.2	FDP_IFF.2.1 FDP_IFF.2.2 FDP_IFF.2.3 FDP_IFF.2.4 FDP_IFF.2.5 FDP_IFF.2.6 FDP_IFF.2.7	Ac_mac.1, Ac_mac.2 Ac_mac.1, Ac_mac.2 Ac_mac.1, Ac_mac.2 Ac_mac.1, Ac_mac.2 Ac_mac.1, Ac_mac.2 Ac_mac.1, Ac_mac.2 Ac_mac.2
	FDP_ITC.1	FDP_ITC.1.1 FDP_ITC.1.2 FDP_ITC.1.3	Ac_mac.2, Ac_mac.3 Ac_mac.2, Ac_mac.3 Ac_mac.3
	FDP_RIP.1	FDP_RIP.1.1	Ac_mac.3
Identification and Authentication	FIA_AFL.1	FIA_AFL.1.1 FIA_AFL.1.2	Auth.1, Auth.2 Auth.1, Auth.2
	FIA_ATD.1	FIA_ATD.1.1	Admin.3
	FIA_SOS.1	FIA_SOS.1.1	Admin.9, Admin.10
	FIA_UAU.1	FIA_UAU.1.1 FIA_UAU.1.2	Auth.1, Auth.2 Auth.1, Auth.2

	FIA_UAU.4	FIA_UAU.4.1	Auth.2, Protect.4
	FIA_UAU.7	FIA_UAU.7.1	Auth.1, Auth.2
	FIA_UID.2	FIA_UID.2.1	Auth.2
	FIA_USB.1	FIA_USB.1.1 FIA_USB.1.2 FIA_USB.1.3	Ac_mac.1 Ac_mac.2 Ac_mac.3
Security Management	FMT_MOF.1	FMT_MOF.1.1	Admin.1
	FMT_MSA.1(1)	FMT_MSA.1.1	Admin.6, Admin.7
	FMT_MSA.1(2)	FMT_MSA.1.1	Admin.2, Admin.3, Admin.4, Admin.5
	FMT_MSA.3(1)	FMT_MSA.3.1 FMT_MSA.3.2	Admin.6, Admin.7 Admin.6, Admin.7
	FMT_MSA.3(2)	FMT_MSA.3.1 FMT_MSA.3.2	Admin.2, Admin.3, Admin.4 Admin.2, Admin.3, Admin.4
	FMT_MTD.1(1)	FMT_MTD.1.1	Admin.8
	FMT_MTD.1(2)	FMT_MTD.1.1	Admin.3, Admin.9
	FMT_MTD.1(3)	FMT_MTD.1.1	Admin.9, Admin.10
	FMT_MTD.1(4)	FMT_MTD.1.1	Admin.11, Protect.2, Protect.3
	FMT_REV.1(1)	FMT_REV.1.1 FMT_REV.1.2	Admin.3, Admin.9 Admin.3, Admin.9
	FMT_REV.1(2)	FMT_REV.1.1 FMT_REV.1.2	Admin.4 Admin.4
	FMT_SMF.1	FMT_SMF.1.1	Admin.1, Admin.2, Admin.3, Admin.4, Admin.5, Admin.6, Admin.7, Admin.8, Admin.9, Admin.10, Admin.11, Admin.12, Protect.1, Protect.2, Protect.3
	FMT_SMR.1	FMT_SMR.1.1 FMT_SMR.1.2	Admin.1, Admin.2, Admin.3, Admin.4, Admin.5, Admin.6, Admin.7, Admin.8, Admin.9, Admin.10, Admin.11, Admin.12 Admin.1, Admin.2, Admin.3, Admin.4, Admin.5, Admin.6, Admin.7, Admin.8, Admin.9, Admin.10, Admin.11, Admin.12

Protection of the TSF	FPT_AMT.1	FPT_AMT.1.1	Protect.1
	FPT_ITT.1	FPT.ITT.1.1	Protect.4
	FPT_RVM.1	FPT_RVM.1.1	Refer.1
	FPT_SEP.1	FPT_SEP.1.1 FPT_SEP.1.2	Refer.2 Refer.2
	FPT_STM.1	FPT_STM.1.1	Audit.1
	FPT_TST.1	FPT_TST.1.1 FPT_TST.1.2 FPT_TST.1.3	Protect.1, Protect.2 Protect.2 Protect.2
TOE access	FTA_SSL.1	FTA_SSL.1.1 FTA_SSL.1.2	Protect.3 Protect.3
Trusted path/channels	FTP_ITC.1	FTP_ITC.1.1 FTP_ITC.1.2 FTP_ITC.1.3	Protect.4 Protect.4 Protect.4

Refer.1 - System Call Interception

This satisfies the **FPT_RVM.1** because the Refer.1 ensures that all controlled system calls are intercepted.

Refer.2 - Security Module Separation

This satisfies the **FPT_SEP.1** because the Refer.2 maintains a secure domain within the commercial operating system for trusted execution.

Ac_mac.1 - Security Label Assignment

This satisfies the **FDP_IFC.1** because the Ac_mac.1 applies MAC policy between creating subject and created subject when performing subject creation on behalf of the user.

This satisfies the **FDP_IFF.2** because the Ac_mac.1 provides the functions applying policies of multi-level based MAC when new subject is created, and configuring security attributes for subject in case of allowed enforcement.

This satisfies the **FIA_USB.1** because the Ac_mac.1 provides the functions configuring security attributes on the basis of security attributes of users for performing subject on behalf of the user.

Ac_mac.2 Multi-Level based MAC

This satisfies the **FDP_IFC.1** because the Ac_mac.2 applies MAC policy over mandatory operation when a subject accesses to an object.

This satisfies the **FDP_IFF.2** because the Ac_mac.2 applies MAC by multi-level based security attributes of subject and object when a subject accesses to an object.

This satisfies the **FDP_ITC.1** because the Ac_mac.2 ensures the functions enforcing the MAC policy for importing user data without security attributes.

Ac_mac.3 Inheritance and Revocation

This satisfies the **FDP_ITC.1** because the Ac_mac.3 inherits security attributes based on subject for the imported data from outside of the TSC.

This satisfies the **FDP_RIP.1** because the Ac_mac.3 provides the functions to inherit attributes of subject when an object is created, and revokes security attributes and previous information content when an object is deleted to ensure that the previous information of a resource is no longer available.

Ac_dac.1 ACL based DAC

This satisfies the **FDP_ACC.1** because the Ac_dac.1 provides the functions enforcing DAC based on ACL policies when a subject accesses to an object through operation of DAC.

This satisfies the **FDP_ACF.1** because the Ac_dac.1 provides the functions enforcing DAC based on ACL policy including the security attributes of subject(identity of subject, group memberships of subject, process of subject) and the security attributes of object(access permission operation).

Ac_dac.2 Allowed/Denied List based DAC

This satisfies the **FDP_ACC.1** because the Ac_dac.2 provides the functions enforcing DAC based on the allowed rule of setuid, the allowed rule for su operation, the denied rule of the restricting command execution, the denied rule of the controlling kill signal when a subject try to access to an object through the DAC operation.

This satisfies the **FDP_ACF.1** because the Ac_dac.2 provides the function enforcing DAC on the basis of security attributes of subject (security role status of subject).

Auth.1 ESM Authentication

This satisfies the **FIA_AFL.1** because the Auth.1 provides the function to terminate the ESM if ESM's authentication of authorized administrator fails 5 times in succession.

This satisfies the **FIA_UAU.1** because the Auth.1 provides user authentication function for ESM.

This satisfies the **FIA_UAU.7** because the Auth.1 provides only '*' or space by appointed authentication feedback to the user of ESM while the authentication is in progress.

Auth.2 SecureOS Authentication

This satisfies the **FIA_AFL.1** because the Auth.2 provides the function to disconnect with Secure OS if the authentication of authorized administrator fails 5 times in succession.

This satisfies the **FIA_UAU.1** because the Auth.2 provides user authentication function for SecureOS.

This satisfies the **FIA_UAU.4** because the Auth.2 ensures that SecureOS authentication data can not be reused by using SSL protocol.

This satisfies the **FIA_UAU.7** because the Auth.2 provides only '*' or space by appointed authentication feedback to the user while the authentication is in progress.

This satisfies the **FIA_UID.2** because the Auth.2 ensures the function identifying whether a connection is allowed on the basis of administrator's identification and ESM's connection IP before authenticating an administrator.

Audit.1 Audit Generation and Collection

This satisfies the **FAU_GEN.1** because the Audit.1 provides function generating an audit record for the auditable events.

This satisfies the **FAU_GEN.2** because the Audit.1 ensures the function associating each auditable event with the identity of the user that caused the event.

This satisfies the **FAU_SEL.1** because the Audit.1 ensures the function including or excluding auditable events from the set of audited events.

This satisfies the **FPT_STM.1** because the Audit.1 provides reliable time stamps requiring at audit generation and collection in sequence from the ESM.

Audit.2 Potential Violation Analysis

This satisfies the **FAU_ARP.1** because the Audit.2 provides corresponding action in case of potential violation detection.

This satisfies the **FAU_SAA.1** because the Audit.2 ensures the function indicating as a potential violation when accumulation and combination of security violation attained to the configured value.

Audit.3 Audit Storage Management

This satisfies the **FAU_STG.1** because the Audit.3 provides the functions protecting the stored audit records from the unauthorized deletion and modification.

This satisfies the **FAU_STG.3** because the Audit.3 ensures the protection functions of possible audit data loss taking the corresponding actions notifying to administrator if the audit trail exceeds the audit space defined by the authorized administrator.

This satisfies the **FAU_STG.4** because the Audit.3 provides the preventive function of the auditable events deferring the TSF to be called by all users except the authorized administrator if the audit trail is full.

Audit.4 Audit Review

This satisfies the **FAU_SAR.1** because the Audit.4 provides function that authorized administrator is able to review audit data.

This satisfies the **FAU_SAR.2** because the Audit.4 ensures the function prohibiting all users read access to the audit records except those the authorized administrators.

This satisfies the **FAU_SAR.3** because the Audit.4 ensures the function providing the ability to perform searching and sorting of audit data.

Audit.5 Simple Attack Prevention

This satisfies the **FAU_SAA.3** because the Audit.5 provides the functions detecting the signature events from the auditable events and indicating a potential violation.

Admin.1 Security Functions Management

This satisfies the **FMT_MOF.1** because the Admin.1 provides functions that authorized administrator is able to start or stop security functions.

This satisfies the **FMT_SMF.1** because the Admin.1 provides GUI and CLI to start or stop security functions.

This satisfies the **FMT_SMR.1** because the Admin.1 ensures that associating a user with the role of authorized administrator.

Admin.2 Hierarchical Category Management

This satisfies the **FMT_MSA.1(2)** because the Admin.2 provides the function that authorized administrator manages non-hierarchical category.

This satisfies the **FMT_MSA.3(2)** because the Admin.2 provides initial value of non-hierarchical category.

This satisfies the **FMT_SMF.1** because the Admin.2 provides GUI and CLI to manage security category.

This satisfies the **FMT_SMR.1** because the Admin.2 ensures that associating a user with the role of authorized administrator.

Admin.3 Labeled Users Management

This satisfies the **FIA_ATD.1** because the Admin.3 defines security attributes of labeled users.

This satisfies the **FMT_MSA.1(2)** because the Admin.3 provides functions that authorized administrator can assign, modify, and delete the security attributes of users.

This satisfies the **FMT_MSA.3(2)** because the Admin.3 provides initial value when an authorized administrator assigns security attributes to users.

This satisfies the **FMT_MTD.1(2)** because the Admin.3 provides the functions initializing and deleting audit data of this user when an authorized administrator assigns or revokes security attributes of user.

This satisfies the **FMT_REV.1(1)** because the Admin.3 assigns ability to revoke security attributes of labeled users to the authorized user only.

This satisfies the **FMT_SMF.1** because the Admin.3 provides GUI and CLI to manage security attributes of labeled user.

This satisfies the **FMT_SMR.1** because the Admin.3 ensures associating a user with the role of authorized administrator.

Admin.4 Labeled Objects Management

This satisfies the **FMT_MSA.1(2)** because the Admin.4 provides functions that authorized administrator can assign and delete security attributes of object(files).

This satisfies the **FMT_MSA.3(2)** because the Admin.4 provides initial value when an authorized administrator assigns security attributes to object(files).

This satisfies the **FMT_REV.1(2)** because the Admin.4 ensures an exclusive ability for the authorized administrator to revoke security attributes of the object(files).

This satisfies the **FMT_SMF.1** because the Admin.4 provides GUI and CLI to manage security attributes of object (file).

This satisfies the **FMT_SMR.1** because the Admin.4 ensures associating a user with the role of authorized administrator.

Admin.5 Labeled Processes Management

This satisfies the **FMT_MSA.1(2)** because the Admin.5 provides the function that authorized administrator can query the security attributes of subject(processes).

This satisfies the **FMT_SMF.1** because the Admin.5 provides GUI and CLI to query security attributes of subject.

This satisfies the **FMT_SMR.1** because the Admin.5 ensures associating a user with the role of authorized administrator.

Admin.6 ACL Policies Management

This satisfies the **FMT_MSA.1(1)** because the Admin.6 provides functions that authorized administrator query, add, modify, and delete policies of ACL based DAC.

This satisfies the **FMT_MSA.3(1)** because the Admin.6 provides initial value when an authorized administrator adds ACL policy.

This satisfies the **FMT_SMF.1** because the Admin.6 provides GUI and CLI to manage ACL policies.

This satisfies the **FMT_SMR.1** because the Admin.6 ensures associating a user with the role of authorized administrator.

Admin.7 Allowed/Denied List Management

This satisfies the **FMT_MSA.1(1)** because the Admin.7 provides functions that authorized administrator can query, add, and delete allowed/denied list of DAC policies.

This satisfies the **FMT_MSA.3(1)** because the Admin.7 provides initial value when an authorized administrator adds allowed/denied list.

This satisfies the **FMT_SMF.1** because the Admin.7 provides GUI and CLI to manage allowed/denied list.

This satisfies the **FMT_SMR.1** because the Admin.7 ensures associating a user with the role of authorized administrator.

Admin.8 Audit Configuration

This satisfies the **FMT_MTD.1(1)** because the Admin.8 provides ability that authorized administrator can manage the configuration of audit data environment and alarms.

This satisfies the **FMT_SMF.1** because the Admin.8 provides GUI and CLI to manage the configuration of audit data environment and alarms.

This satisfies the **FMT_SMR.1** because the Admin.8 ensures associating a user with the role of authorized administrator.

Admin.9 ESM Users Management

This satisfies the **FIA_SOS.1** because the Admin.9 ensures the functions providing a mechanism to verify whether the ESM user's registered password satisfies the defined quality criteria.

This satisfies the **FMT_MTD.1(2)** because the Admin.9 provides the function to delete authentication data of the user when the ESM user is deleted.

This satisfies the **FMT_MTD.1(3)** because the Admin.9 provides the function that only authorized administrator and allowed user can modify identification and authentication data of ESM users.

This satisfies the **FMT_REV.1(1)** because the Admin.9 assigns revocable ability of administrator to authorized user.

This satisfies the **FMT_SMF.1** because the Admin.9 provides GUI to manage user of ESM.

This satisfies the **FMT_SMR.1** because the Admin.9 ensures associating a user with the role of authorized administrator.

Admin.10 Security Password Management

This satisfies the **FIA_SOS.1** because the Admin.10 ensures the functions providing a mechanism to verify whether the Secure OS user's registering security password satisfies the defined quality.

This satisfies the **FMT_MTD.1(3)** because the Admin.10 provides ability that only authorized administrator and allowed user can modify identification and authentication data of SecureOS users.

This satisfies the **FMT_SMF.1** because the Admin.10 provides GUI to manage Secure OS users.

This satisfies the **FMT_SMR.1** because the Admin.10 ensures associating a user with the role of authorized administrator.

Admin.11 Security Functions Configuration

This satisfies the **FMT_MTD.1(4)** because the Admin.11 provides the function for an authorized administrator to manage operating environment of security functions.

This satisfies the **FMT_SMF.1** because the Admin.11 provides GUI and CLI to configure operating environment of security functions.

This satisfies the **FMT_SMR.1** because the Admin.11 ensures associating a user with the role of authorized administrator.

Admin.12 System Services Management

This satisfies the **FMT_SMF.1** because the Admin.12 provides GUI of system services that is able to configure and to manage policies of IP Filter, system monitoring, and system account management existing outside of the TOE.

This satisfies the **FMT_SMR.1** because the Admin.12 ensures associating a user with the role of authorized administrator.

Protect.1 Abstract Machine Testing

This satisfies the **FMT_SMF.1** because the Protect.1 provides GUI to indicate the result of abstract machine testing.

This satisfies the **FPT_AMT.1** because the Protect.1 ensures the functions running a suite of tests when administrator requests during start-up, periodically during operation to demonstrate the correct operation of the abstract machine that underlies the TSF.

This satisfies the **FPT_TST.1** because the Protect.1 ensures the functions running a suite of self tests when administrator requests during start-up, periodically during operation to demonstrate the correct operation of the TSF.

Protect.2 Integrity Functions

This satisfies the **FMT_MTD.1(4)** because the Protect.2 provides the function for an authorized administrator to manage target items and performance of integrity check.

This satisfies the **FMT_SMF.1** because the Protect.2 provides GUI to perform integrity check and managing target items.

This satisfies the **FPT_TST.1** because the Protect.2 ensures the functions performing integrity checking for executable files during start-up, periodically during operation to demonstrate the correct operation of the TSF and providing the authorized users with the capability to verify the integrity of the TSF data and executable code.

Protect.3 ESM Screen Saving

This satisfies the **FMT_MTD.1(4)** because the Protect.3 provides the function for an authorized administrator to manage inactive time interval of ESM.

This satisfies the **FMT_SMF.1** because the Protect.3 provides GUI to manage inactive time interval of ESM.

This satisfies the **FTA_SSL.1** because the Protect.3 ensures the functions locking an interactive session after inactive time interval of ESM user, and requiring events to initiate unlocking the session.

Protect.4 Secure Communication

This satisfies the **FIA_UAU.4** because the Protect.4 ensures preventing reuse of transmitted data by utilizing SSL version 3 protocol.

This satisfies the **FTP_ITC.1** and the **FPT_ITT.1** because the Protect.4 ensures that secure channel is configured by SSL version 3 protocol, when an authorized administrator tries to manage TOE from a remote place.

8.4.2 Justification for Compliance of Assurance Measures

The TOE summary specification in section 6.2 includes a justification that each TOE security assurance requirement is met by appropriate assurance measures.

8.5 Rationale for Strength of Function

The information to be protected by this TOE is the general data of government and the value is medium. The assumed level of expertise of the attacker for all the threats is unsophisticated, with access to standard equipment and public information. In the Common Criteria[1)] recommend to provide the security functions higher than minimum SOF-basic against the threats which have the low level of the attack potential.

Therefore, this is done in accordance with the SOF-medium for the strength of function.

This TOE satisfies the SOF-medium for the following security functional requirements to meet the threats with **T.UAUTH and T.BYPASS**.

- FIA_UAU.1 Timing of authentication
- FIA_UAU.4 Single-use authentication mechanisms

This TOE satisfies the SOF-medium for the following security functional requirements to meet the security objectives with **O.IA**.

- FIA_UAU.1 Timing of authentication
- FIA_UAU.4 Single-use authentication mechanisms

This TOE satisfies the SOF-medium for the following security functional requirements.

[1)] In the Annex B.8 Strength of function and vulnerability analysis of the Common Evaluation Methodology for Information Technology Security, Part 2, the minimum strength of function based on the method of calculating the attack potential is defined.

- FIA_UAU.1 Timing of authentication
- FIA_UAU.4 Single-use authentication mechanisms