KSignAccess V4.0

Security Target V1.2

KSign Co., Ltd.

KSIGN e-Security Leader

* The Security Target related to the certified TOE. This Security Target is written in Korean and translated from Korean into English.

18, Nonhyeon-ro 64-gil, Gangnam-gu, Seoul

TEL : 02-564-0182 FAX : 02-564-1627

http://www.ksign.com

KSign Co., LTD.

Rivision History

Version	Date	Detail	Created by	Reviewed by
V1.0	2017.11.20	Initial version	Park Byeong-il	Kim Jong-hoon
V1.1	2018.08.06	Update	Park Byeong-il	Uh Seong-Ryul
V1.2	2018.10.26	Update	Park Byeong-il	Uh Seong-Ryul

Contents

1. ST Introduction

1.1 ST reference

Title	KSignAccess V4.0 Security Target
Version	V1.2
Author	KSign Co., LTD.
Publictaion Date	2018. 10. 26
Common Criteria version	CC V3.1 r5
Evaluation Assurance Level	EAL 1+ (ATE_FUN.1)
Protection Profile	Korean National Protection Profile for Single Sign On V1.0
Keywords	Single Sign On, SSO

[Table 1-1] ST reference

1.2 TOE reference

Item			**Specification**
TOE			KSignAccess V4.0
Version			V4.0.2
Components	**KSignAccess Server**		KSignAccess Server V4.0.2
	KSignAccess Agent	**KSignAccess Agent for Linux**	KSignAccess Agent for Linux V4.0.2
		KSignAccess Agent for Solaris	KSignAccess Agent for Solaris V4.0.2
		KSignAccess Agent for HP-UX	KSignAccess Agent for HP-UX V4.0.2
		KSignAccess Agent for AIX	KSignAccess Agent for AIX V4.0.2
		KSignAccess Agent for Windows	KSignAccess Agent for Windows V4.0.2
Developer			KSign Co., LTD.

[Table 1-2] TOE reference

1.3 TOE overview

1.3.1 Single Sign On overview

The TOE is a used to enable the user to access various business systems and use the service through a single user login(Single Sign-On) without additional login action. The TOE performs user identification and authentication, authentication token(hereinafter referred to as "token") issue and validity verification according to the user authentication policy.

The TOE shall provide the user login capability using ID and password, issue a token during user login, and verify the issued token if accessing another business system after user login.

The primary security features provided by the TOE include user identification and authentication, token issue, storage, verification and destruction. The TOE must use a validated cryptographic module whose security and implementation conformance are validated by the Korea Cryptographic Module Validation Program (KCMVP).

The TOE uses the following validated cryptographic module.

- Cryptographic module name : KSignCASE64 v2.5
- Validation number : CM-103-2020.10
- Expiration date : Oct 05, 2020

1.3.2 TOE type and scpoe

The TOE defined by this Security Target is SSO that enables the user to access various business systems through a single user login, and the TOE is provided as software.

The KSignAccess Server and the KSignAccess Agent are the indispensable TOE component defined in this ST. The TOE is composed of the server that processes user login, manages the token, and sets the policy; and the agent that is API type installed in each business system performs the function of token issue and verification.

1.3.3 TOE usage and major security features

The TOE performs End user identification and authentication to enable the End user to access various business systems and use the service through a single user login without additional login action.

The TOE provides the security audit function that records and manages critical events as audit data when activating the security functionality and management function, function of protecting the data that stored in the TSF controlled repository, and TSF protection function such as TSF self-testing. In addition, the TOE provides authentication failure handling, identification and authentication functions including mutual authentication between the TOE components, cryptographic support function such as cryptographic key management and cryptographic operation for issuing a token, security management function for management of security functions behaviour and configuration, and the TOE access function to manage the authorized administrator's interacting session. In addition, the token provides confidentiality and integrity protection, and the TOE executable code provides integrity protection.

The end user identification and authentication process is divided into the initial authentication phase using the ID/password and the token-based authentication phase that accesses the business system using the token issued during the initial authentication procedure.

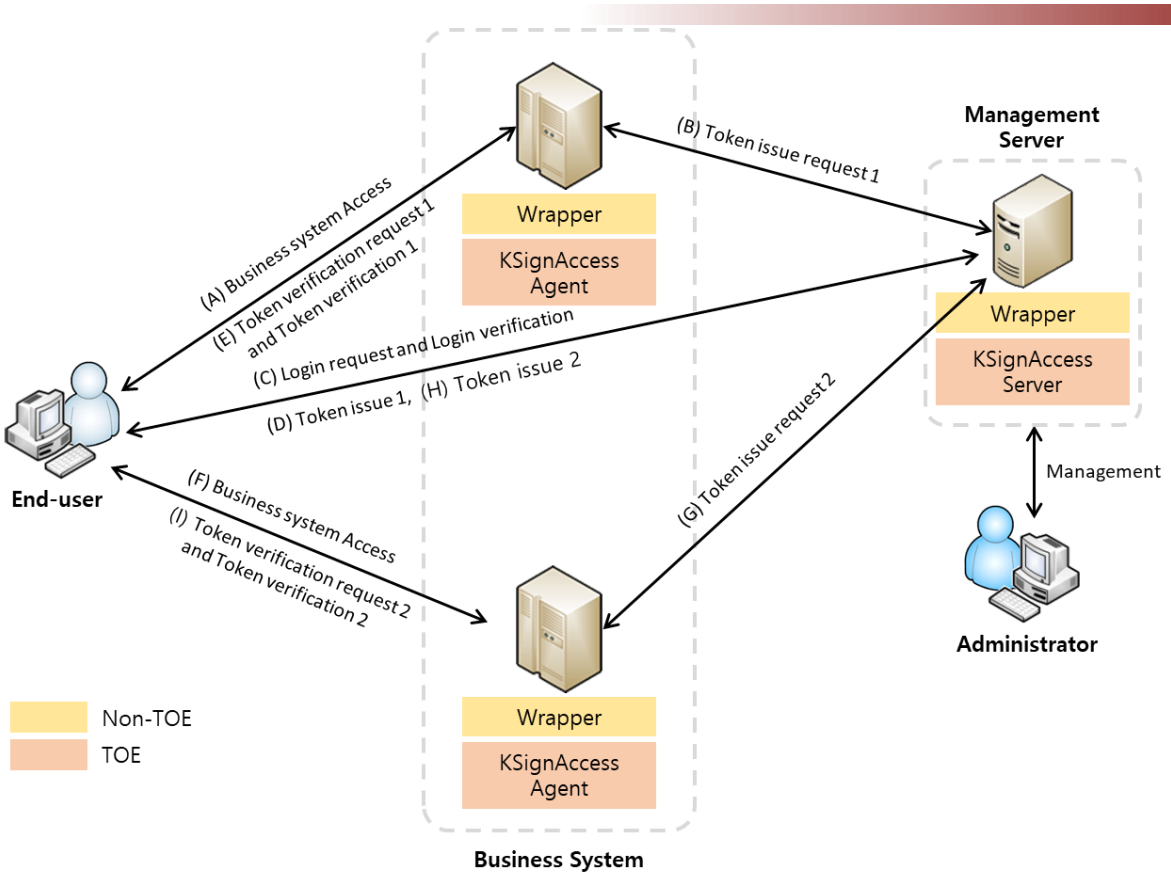
The execution procedure of the initial authentication phase is as follows.

End user has access to the business system, and the KSignAccess agent that receives the login request message sends a login verification request to the KSignAccess Server, which in turn checks the authorized user status.

Upon receiving the login verification request, the KSignAccess Server performs login verification directly using the End user information stored in the DBMS. The KSignAccess Server requests token issue to the KSignAccess Agent if the login verification result is valid. The KSignAccess Server or KSignAccess Agent transfers an issued token to the user.

The token-based authentication phase is performed only when the token has been normally issued in the initial authentication phase.

When the user utilizes business system services, the issued token is transferred to the KSignAccess Agent installed in the pertinent business system, and the KSignAccess Agent verifies the validity of the token by interfacing with the KSignAccess Server upon receiving the token.token.



[Figure 1-1] end-user identification and authentication procedure

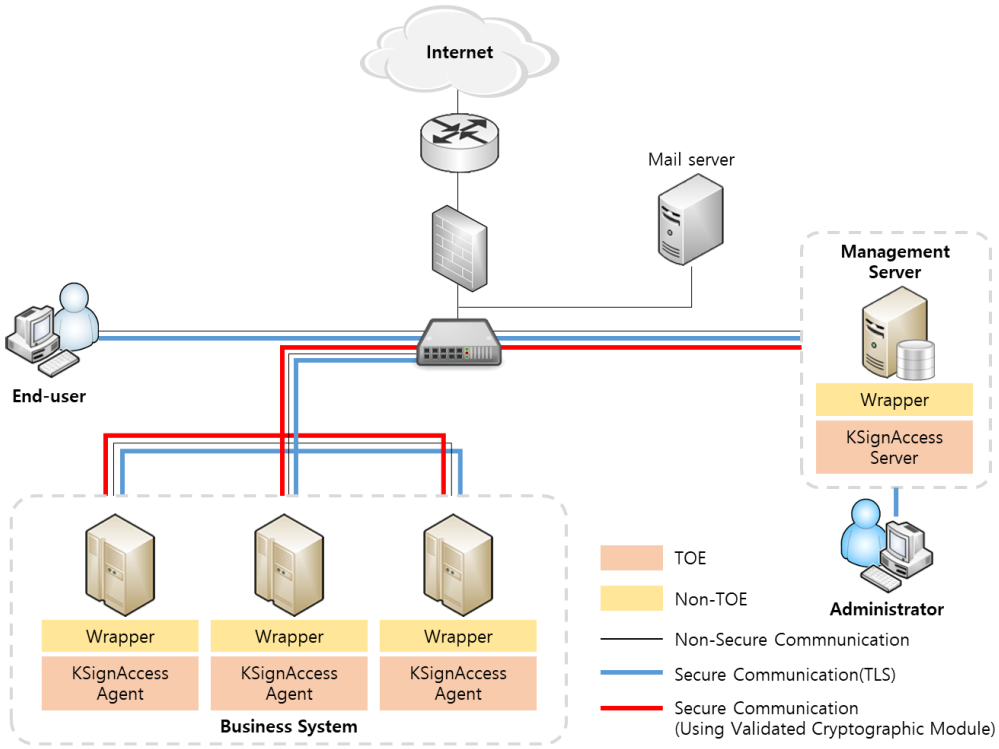
Authentication phase	Operation procedure
Initial authentication	(A) Business system Access – (B) Token issue request 1 – (C) Login request and Login verification – (D) Token issue 1 - (E) Token verification request 1 and Token verification 1
Token-based authentication	(F) Business system Access – (G) Token issue request 2 – (H) Token issue 2 - (I) Token verification request 2 and Token verification 2

[Table 1-3] operation procedure by authentication phase

- Authentication token issuer : KSignAccess Server
- Authentication token storage location : End User PC
- Authentication token validator : KSignAccess Agent

1.4 TOE operational environment

1.4.1 non-TOE and TOE operational environment



[Figure 1-2] TOE Operation environment

Figure 1-2 shows the general TOE operational environment. The TOE operational environment consists of KSignAccess Server and KSignAccess Agent. The KSignAccess Server verifies end user login attempts directly using the end user information stored in the DBMS, the token management, and the policy configuration, Provides security management interface functions for TOE management. The KSignAccess Agent, consisting of library files, performs end user login verification requests from KSignAccess Server, authentication token verification functions in each business system.

When the TOE Administrator accesses the TOE security management interface by entering the management server web address on the browser, the browser forms an HTTPS security channel.

There may exist various external entities necessary for the operation of the TOE, including email server to notify the authorized administrator in case of audit data loss. The mail server, which is an external entity other than TOE, corresponds to the TOE operational environment.

1.4.2 Requirements for non-TOE software, hardware, firmware

The requirements for hardware, software and operating system to install the TOE are as in the following.

TOE		Item	Specification
KSignAccess Server		CPU	Intel Xeon 3.5 GHz or higher
		Memory	16GB or higher
		HDD	Space required for installation of TOE 500MB or higher
		NIC	100/1000 Mbps x 1EA or higher
KSignAccess Agent	KSignAccess Agent for Linux	CPU	Intel Core i3 3.07 GHz or higher
		Memory	4 GB or higher
		HDD	Space required for installation of TOE 500MB or higher
		NIC	100/1000 Mbps x 1EA or higher
	KSignAccess Agent for Solaris	CPU	SUN SPARC 1.28 GHz or higher
		Memory	4 GB or higher
		HDD	Space required for installation of TOE 500MB or higher
		NIC	100/1000 Mbps x 1EA or higher
	KSignAccess Agent for HP-UX	CPU	Intel Itanium(IA64) 1.67 GHz or higher
		Memory	4 GB or higher
		HDD	Space required for installation of TOE 500MB or higher
		NIC	100/1000 Mbps x 1EA or higher
	KSignAccess Agent for	CPU	PowerPC POWER5 2.1 GHz or higher
		Memory	4 GB or higher

	AIX	HDD	Space required for installation of TOE 500MB or higher
		NIC	100/1000 Mbps x 1EA or higher
	KSignAccess Agent for Windows	CPU	Intel Core i3 3.30 GHz or higher
		Memory	4 GB or higher
		HDD	Space required for installation of TOE 500MB or higher
		NIC	100/1000 Mbps x 1EA or higher

[Table 1-4] Requirement hardware for non-TOE

The operating system on which the TOE operates is as in the following.

TOE		Operating system
KSignAccess Server		CentOS 6.8 kernel 2.6.32 (64 bit)
KSignAccess Agent	KSignAccess Agent for Linux	CentOS 6.8 kernel 2.6.32 (64 bit)
	KSignAccess Agent for Solaris	Solaris 5.10 (64bit)
	KSignAccess Agent for HP-UX	HP-UX 11.31 (64bit)
	KSignAccess Agent for AIX	AIX 7.1 (64bit)
	KSignAccess Agent for Windows	Windows Server 2008 R2 (64bit)
		Windows Server 2012 R2 (64bit)

[Table 1-5] Operating system for TOE support

The requirements for security management are as in the following.

Item	Sub item	Specification
Hardware	CPU	Intel Core i3 2.30GHz or higher
	Memory	4GB or higher
	HDD	300GB or higher
	NIC	100/1000 Mbps x 1EA or higher

Software	OS	Windows 7 Professional Service Pack 1 (64bit)
	Web Browser	Internet Explorer 11

[Table 1-6] Requirement for management system

The TOE uses the following validated cryptographic module.

TOE	S/W	Specification
KSignAccess Server	KSignCASE64 v2.5	Validation cryptographic modules for Cryptographic key generation, distribution, destruction and Cyryptographic operation, cryptographic commnunication between KSignAccess Server and KSign Access Agent.
KSignAccess Agent	KSignCASE64 v2.5	Validation cryptographic modules for Cryptographic key generation, distribution, destruction and Cyryptographic operation, cryptographic commnunication between KSignAccess Server and KSign Access Agent.

[Table 1-7] Validated cryptographic module information

The details of the validated cryptographic module included in the TOE are as following.

Item	Specification
Cryptographic module name	KSignCASE64 v2.5
Developer	KSign Co., Ltd.
Validation date	Oct 05, 2015
Validation level	VSL1
Validation number	CM-103-2020.10

[Table 1-8] details of Validated cryptographic module

Non-TOE software that is not within the TOE range but is required to operate normally is as following.

TOE	S/W	Specification
KSignAccess Server	Java(JDK) 1.8.0_191	Running and operating KSignAccess Server based on Java Application, security management, and Web Server Operation
	Apache Tomcat 8.5.34	Performing security management by authorized administrators. Java based operation of the Web Application Server (WAS)
	MySQL 5.6	TOE Policy Settings and Audit Data Stores
KSignAccess Agent	Java(JDK) 1.5.0_22	Running and operating KSignAccess Server based on Java Application, security management, and Web Server Operation
	Apache Tomcat 5.5.36	Java based operation of the Web Application Server (WAS)

[Table 1-9] Software for non-TOE

Operating the TOE requires the following additional systems in the IT environment.

Item	Specification
Mail Server (SMTP Server)	Send alert mail to administrators

[Table 1-10] Operation support IT environment for TOE

1.5 TOE description

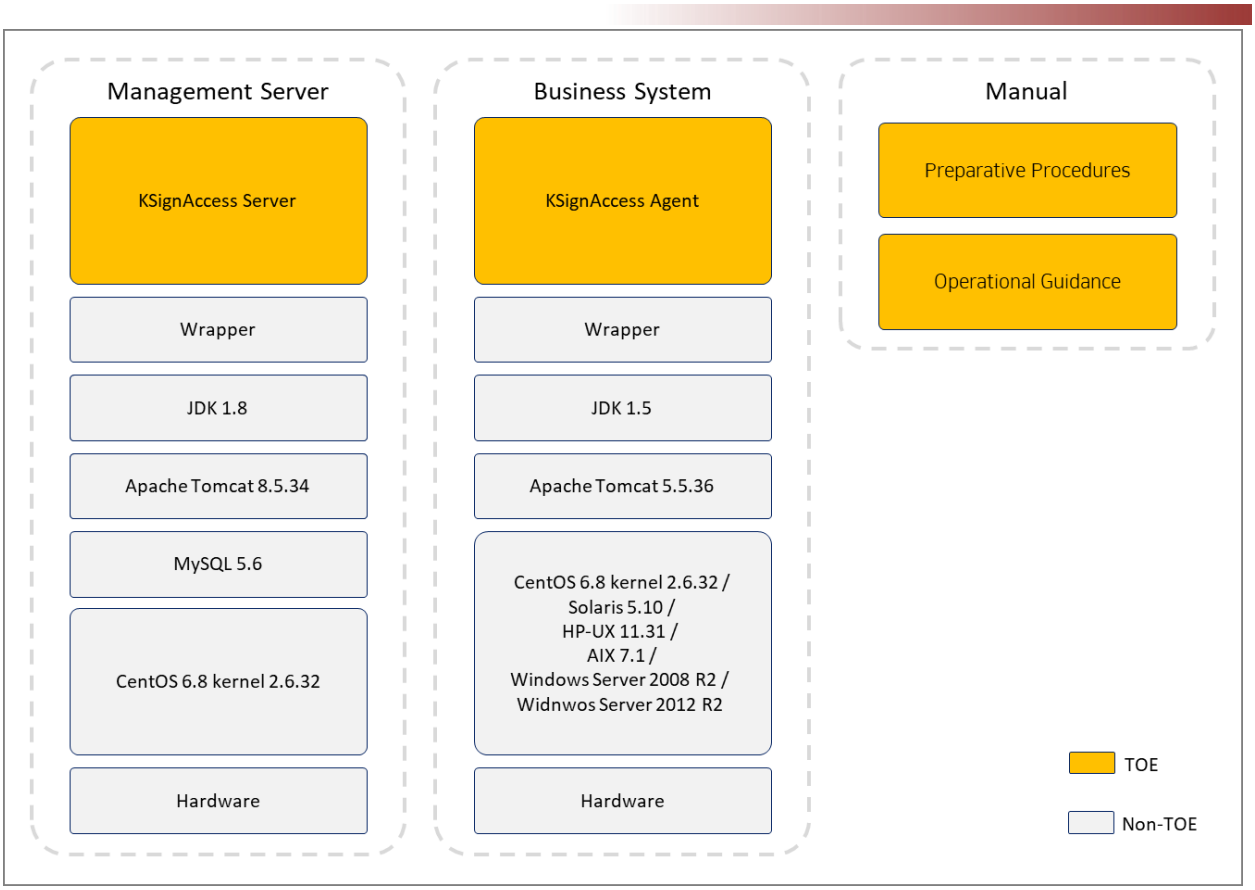
In this part, the physical scope of the TOE and guidelines are described and security features provided by the TOE are explained in detail in the logical scope of the TOE.

1.5.1 Physical scope of the TOE

The physical scope that makes up the TOE is KSignAccess Server, KSignAccess Agent and guidelines (User Operation Guide, Preparative Procedure) as shown in the below [Figure 1-11]. Validated Cryptographic Module (KSignCASE64 v2.5) is embedded in the TOE components.

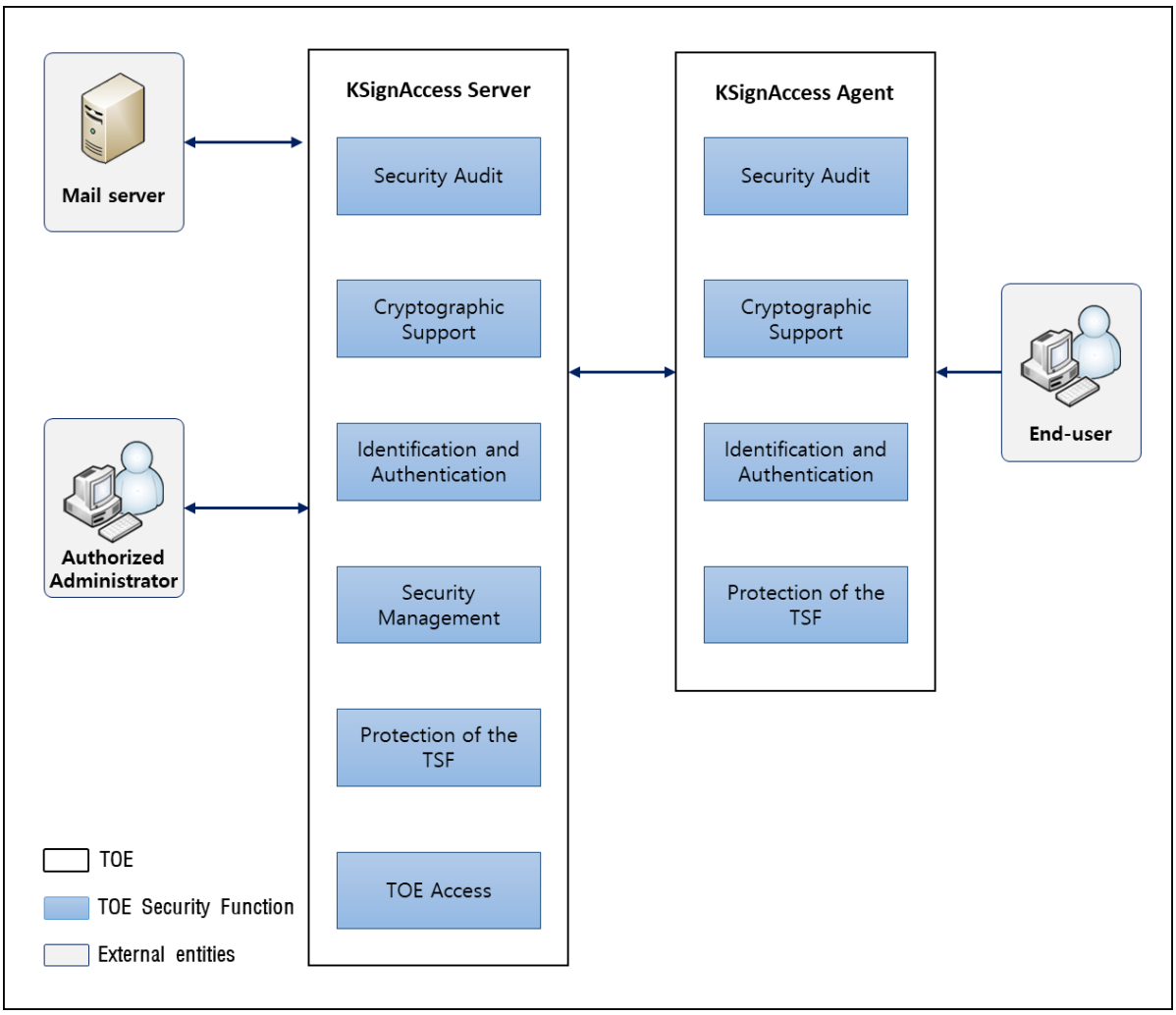
Scope		Distribution Status	Type	Distribute
TOE components	KSignAccess Server	KSignAccess Server V4.0.2 (KSignAccess_Server_V4.0.2.tar)	S/W	Distributed as a CD
	KSignAccess Agent	KSignAccess Agent for Linux V4.0.2 (KSignAccess_Agent_Linux_V4.0.2.tar)		
		KSignAccess Agent for Solaris V4.0.2 (KSignAccess_Agent_Solaris_V4.0.2.tar)		
		KSignAccess Agent for HP-UX V4.0.2 (KSignAccess_Agent_HP-UX_V4.0.2.tar)		
		KSignAccess Agent for AIX V4.0.2 (KSignAccess_Agent_AIX_V4.0.2.tar)		
		KSignAccess Agent for Windows V4.0.2 (KSignAccess_Agent_Windows_V4.0.2.zip)		
Manual	Preparative Procedure	KSignAccess V4.0 Preparative Procedure V1.2 (KSignAccess V4.0 Preparative Procedure V1.2.pdf)	File (PDF)	Distributed as a CD
	Operation Guide	KSignAccess V4.0 Operation Guide V1.2 (KSignAccess V4.0 Operation Guide V1.2.pdf)		

[Table 1-11] Physical scope of the TOE



[Figure 1-3] Physical scope of the TOE

1.5.2 Logical scope of the TOE



[Figure 1-4] Logical scope of the TOE

■ Security Audit

- KSignAccess Server provides audit information only to authorized administrators and understand it. Generate audit data in the event of an audited event, detect potential violations, and send an alert email to an authorized administrator. It also provides the ability to store and manage all generated audit data safely in an audit trail (DBMS), prevent unauthorized deletion of audit data, and protect the audit trail by ignoring audited events when the audit trail is possible audit data loss.

- The KSignAccess Agent records the audit trail of the KSignAccess Agent in the event of an audit of the end-user's identification and authentication success and failure, and the KSignAccess Agent's integrity verification.

■ Cryptographic support

- TOE is generates and discards all cryptographic keys used for product operation through KSignCASE64 v2.5, which is a validated cryptographic module whose security and implementation conformity is verified through the cryptographic module verification system. Token generation / verification.
- In addition, a cryptographic key is generated and exchanged through KSignCASE64 v2.5, a validated cryptographic module, for secure communication between physically separated KSignAccess Server and KSignAccess Agent.
  - Cryptographic key generation :
    - HASH_DRBG(SHA256, 256bit) : Symmetric key generate for data encryption/decryption.
    - RSAES(2048bit) : Asymmetric key generate for data encryption/decryption.
  - Symmetric Key Encryption/Decryption(SEED-CBC, 128bit) : Authentication Token Encryption/Decryption and important settings information encryption, secure communication channel between KSignAccess Server and KSignAccess Agent.
  - Asymmetric Key Encryption/Decryption(RSAES, 2048bit) : Authentication token, Symmetric key exchange for Authentication token request message encryption.
  - Digital signature generation and verification (RSA-PSS, 2048bit) : TOE Component mutual authentication.
  - MessageDigest(SHA-256) : generate of setting and module integrity data.
  - HMAC(HMAC-SHA256) : generate of TOE mutual authentication data.
  - Authenticate token generation : KSignAccess Server generation using validated cryptographic module.
  - Authenticate token verification : KSignAccess Agent verifiacation using validated cryptographic module.

■ Identification and authentication

- KSignAccess Server provides the function to identification and authentication the administrator who wants to use the security management function before every action and to protect the authentication feedback when entering the authentication data. In addition, it provides a secure identification and authentication function according to the authentication lock processing function in case of continuous authentication failure. It also blocks attempts to reuse authentication information for administrators logging in to KSignAccess Server.
- KSignAccess Agent identification and authentication to KSignAccess Agent for end-user to use single sign on function. It provides a function to protect authentication feedback when entering authentication data and provides secure identification and authentication function according to authentication lock processing function in case of consecutive authentication failure. It also blocks attempts to reuse authentication information for normal users logging in to KSignAccess Agent.
- The TOE provides a mechanism to verify the following enforce defined quality metrics for administrator and end-user password verification.
  - Min/Max length : 9 ~ 16 digits
  - Allowable characters : English letter (26 letters: a~z), Number (10 letters: 0~9), Special character that can be input by using a keyboard (27 letters: .-/+=_,:!@#$%^*()~{}[]|<>;&)
  - At least 1 Letter to English letter, Number, Special character
- When generating the authentication token used by the TOE, the authentication token is generated using the one-time authentication data (using the time stamp) through validated cryptographic module, the authentication token is overwritten with data 0x30 when the authentication token is destroyed.
- TOE performs mutual authentication through a self-implemented protocol between the KSignAccess Server and the KSignAccess Agent.

■ Security Management

- KSignAccess Server provides access control policy management, administrator management, security management function of KSignAccess Server configuration to authorized administrator, and authorized administrator performs security management through security management interface.

- The authorized administrator is the top administrator and the audit administrator. The top administrator can perform all the security management functions of the TOE through the security management interface, and the audit administrator can perform the audit data inquiry function.
- The authorized administrator forces the password to be changed at the first access to the security management interface. In case of the audit administrator, the password must be changed after the password is reset by the authorized administrator.
- The authorized administrator can change the password of the administrator or the end-user through the security management interface and verifies the validity of the password value according to the password policy when generating and changing the passwords of the end-user and the authorized administrator.

■ Protection of the TSF

- KSignAccess Server guarantees the confidentiality and integrity of TSF data transmitted for physically separated KSignAccess Agent through secure communication.
  KSignAccess Server maintains secure state and maintains security function at during intial start-up and regular interval The TOE performs periodic self-tests to check the status of the process and performs integrity checks on the TSF data and the TSF executable code that are subject to the integrity check.
- The integrity list that KSignAccess Server checks is as follows
  * KSignAccess-common-4.0.2.jar(KSignAccess common module)
  * KSignAccessServer-4.0.2.jar(KSignAccess Server module)
  * KSignAccessServer-core-4.0.2.jar(KSignAccess Server Core module)
  * KSignAccess Server Configure hashed data(KSignAccess Server configuration file)
  * KSignAccess Server DB Configure hased data(KSignAccess Server DB configuration file)
- KSignAccess Agent loads TSF data for secure communication and mutual authentication with KSignAccess Server at startup and receives integrity information from KSignAccess Server after mutual authentication succeeds and performs integrity check on TSF data and components.
- The integrity list that KSignAccess Server checks is as follows
  * KSignAccess-common-4.0.2.jar(KSignAccess common module)
  * KSignAccessAgent-4.0.2.jar(KsignAccess Agent module)
  * KSignAccess agent configure hashed data(KSignAccess Agent configuration file)

- The TOE securely stores and manages end-user and administrator authentication information, TOE integrity verification information, KSignAccess Server and KSignAccess Agent information in the DBMS to protect the TSF data. The authentication token is temporarily loaded into the PC memory through the browser of the end-user Discard immediately after use.

■ TOE access

- The maximum number of concurrent sessions is limited to 1 for management access session of administrator who can access to manage security management function of KSignAccess Server. If login is performed with same account or same privilege from other administrator PC after login of authorized administrator, and provides a function of terminating an existing connection. Also, if the administrator session exceeds the set inactivity time, the administrator session is terminated.
- In the case of the audit administrator, the access session is restricted according to the access permission IP rule and the audit data on the result of the session restriction of the security management interface is generated.

1.6 Terms and definitions

Terms used in this ST, which are the same as in the CC, must follow those in the CC.

Application Programming Interface (API)

A set of software libraries that exist between the application layer and the platform system layer and facilitate the development of applications that run on the platform

Approved cryptographic algorithm

A cryptographic algorithm selected by Korean Cryptographic Module Validation Authority for block cipher, secure hash algorithm, message authentication code, random bit generation, key agreement, public key cipher, digital signatures cryptographic algorithms considering safety, reliability and interoperability

Approved mode of operation

The mode of cryptographic module using approved cryptographic algorithm

Assets

Entities that the owner of the TOE presumably places value upon

Assignment

The specification of an identified parameter in a component (of the CC) or requirement

Attack potential

Measure of the effort to be expended in attacking a TOE expressed as an attacker's expertise, resources and motivation

Augmentation

Addition of one or more requirement(s) to a package

Authentication Data

Information used to verify a user's claimed identity

Authentication token

Authentication data that authorized end-users use to access the business system

Authorized Administrator

Authorized user to securely operate and manage the TOE

Top Administrator

The authorized administrator who has the highest authority to perform all security management functions in the security management interface

Audit Administrator

An authorized administrator who can perform the audit record retrieval function in the security management interface

Authorized User

The TOE user who may, in accordance with the SFRs, perform an operation

Business System

An application server that authorized end-users access through 'SSO' Can/could The 'can' or 'could' presented in Application notes indicates optional requirements applied to the TOE by ST author's choice

Class

Set of CC families that share a common focus

Client

Application program that can access the services of SSO server or SSO agent through network

Component

Smallest selectable set of elements on which requirements may be based

Critical Security Parameters (CSP)

Information related to security that can erode the security of the encryption module if exposed or changed (e.g., verification data such as secret key/private key, password, or Personal Identification Number)

Database Management System (DBMS)

A software system composed to configure and apply the database.

Decryption

The act that restoring the ciphertext into the plaintext using the decryption key

Dependency

Relationship between components such that if a requirement based on the depending component is included in a PP, ST or package, a requirement based on the component that is depended upon must normally also be included in the PP, ST or package

Element

Indivisible statement of a security need

Encryption

The act that converting the plaintext into the ciphertext using the encryption key

end-user

Users of the TOE who want to use the business system, not the administrators of the TOE

Evaluation Assurance Level (EAL)

Set of assurance requirements drawn from CC Part 3, representing a point on the CC predefined assurance scale, that form an assurance package

External Entity

Human or IT entity possibly interacting with the TOE from outside of the TOE boundary

Family

Set of components that share a similar goal but differ in emphasis or rigour

Identity

Representation uniquely identifying entities (e.g. user, process or disk) within the context of the TOE

Iteration

Use of the same component to express two or more distinct requirements

Korea Cryptographic Module Validation Program (KCMVP)

A system to validate the security and implementation conformance of cryptographic modules used for protection of important but not classified information among the data communicated through the information and communication network of the government and public institutions

Management access

The access to the TOE by using the HTTPS, SSH, TLS, etc to manage the TOE by administrator, remotely

Management Console

Application program that provides GUI, CLI, etc. to the administrator and provides system management and configuration

Object

Passive entity in the TOE containing or receiving information and on which subjects perform operations

Operation(on a component of the CC))

Modification or repetition of a component. Allowed operations on components are assignment, iteration, refinement and selection

Operation(on a subject)

Specific type of action performed by a subject on an object

Private Key

A cryptographic key which is used in an asymmetric cryptographic algorithm and is uniquely associated with an entity(the subject using the private key), not to be disclosed

Protection Profile (PP)

Implementation-independent statement of security needs for a TOE type

Public Key

A cryptographic key which is used in an asymmetric cryptographic algorithm and is associated with an unique entity(the subject using the public key), it can be disclosed

Public Key(asymmetric) cryptographic algorithm

A cryptographic algorithm that uses a pair of public and private key

Public Security Parameters (PSP)

security related public information whose modification can compromise the security of a cryptographic module

Random bit generator (RBG)

A device or algorithm that outputs a binary sequence that is statistically independent and is not biased. The RBG used for cryptographic application generally generates 0 and 1 bit string, and the sequence can be combined into a random bit block. The RBG is classified into the deterministic and non-deterministic type. The deterministic type RBG is composed of an algorithm that generates bit strings from the initial value called a "seed key," and the non-deterministic type RBG produces output that depends on the unpredictable physical source.

Recommend/be recommended

The 'recommend' or 'be recommended' presented in Application notes is not mandatorily recommended, but required to be applied for secure operations of the TOE

Refinement

Addition of details to a component

Role

Predefined set of rules on permissible interactions between a user and the TOE

Secret Key

The cryptographic key which is used in symmetric cryptographic algorithm and is associated with on or more entity, it is not allowed to release

Secure Sockets Layer (SSL)

This is a security protocol proposed by Netscape to ensure confidentiality, integrity and security over a computer network

Security Policy Document

Document uploaded to the list of the validated cryptographic module with the module's name and specifying the summary for the cryptographic algorithms and operational environments of the TOE

Security Target (ST)

Implementation-dependent statement of security needs for a specific identified TOE

Selection

Specification of one or more items from a list in a component

Self-test

Pre-operational or conditional test executed by the cryptographic module

Sensitive Security Parameters (SSP)

critical security parameters (CSP) and public security parameters (PSP)

Shall/must

The 'shall' or 'must' presented in Application notes indicates mandatory requirements applied to the TOE

Subject

Active entity in the TOE that performs operations on objects

Symmetric cryptographic technique

Encryption scheme that uses the same secret key in mode of encryption and decryption, also known as secret key cryptographic technique

Target of Evaluation (TOE)

Set of software, firmware and/or hardware possibly accompanied by guidance

Threat Agent

Entity that can adversely act on assets

TOE Security Functionality (TSF)

Combined functionality of all hardware, software, and firmware of a TOE that must be relied upon for the correct enforcement of the SFRs

Transport Layer Security (TLS)

This is a cryptographic protocol between a SSL-based server and a client and is described in RFC 2246

TSF Data

Data for the operation of the TOE upon which the enforcement of the SFR relies

User

Refer to "External entity", authorized administrator and authorized end-user in the TOE

Validated Cryptographic Module

A cryptographic module that is validated and given a validation number by validation authority

Wrapper

Interfaces for interconnection between the TOE and various types of business systems or authentication systems

1.7 Conventions

The notation, formatting and conventions used in this ST are consistent with the Common Criteria for Information Technology Security Evaluation.

The CC allows several operations to be performed for functional requirements: iteration, assignment, selection and refinement. Each operation is used in this PP.

Iteration

Iteration is used when a component is repeated with varying operations. The result of iteration is maked with an iteration number in parenthesis following the component identifier, i.e., denoted as (iteration No.).

Assignment

This is used to assign specific values to unspecified parameters (e.g., password length). The result of assignment is indicated in square brackets like [assignment_value].

Selection

This is used to select one or more options provided by the CC in stating a requirement. The result of selection is shown as *underlined and italicized.*

Refinement

This is used to add details and thus further restrict a requirement. The result of refinement is shown in **bold text**.

2. Conformance claim

2.1 CC conformance claim

Common Criteria (CC)		Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5 - Common Criteria for Information Technology Security Evaluation. Part 1: Introduction and General Model, Version 3.1, Revision 5 (CCMB-2017-04-001, April, 2017) - Common Criteria for Information Technology Security Evaluation. Part 2: Security Functional Components, Version 3.1, Revision 5 (CCMB-2017-04-002, April, 2017) - Common Criteria for Information Technology Security Evaluation. Part 3: Security Assurance Components, Version 3.1, Revision 5 (CCMB-2017-04-003, April, 2017)
Protection Profile (PP)		Korean National Protection Profile for Single Sign On V1.0
Conformance Claim	**Part 2 Security functional components**	Extended : FCS_RBG.1, FIA_IMA.1, FIA_SOS.3, FMT_PWD.1, FPT_PST.1, FTA_SSL.5
	Part 3 Security assurance components	Conformant
	Package	Augmented : EAL1 augmented (ATE_FUN.1)

[Table 2-1] Conformance Claim

2.2 PP conformance claim

This ST claim conformance the following PP.

- Korean National Protection Profile for Single Sign On V1.0

2.3 Package conformance claim

This ST claims conformance to assurance package EAL1 augmented with ATE_FUN.1.

2.4 Conformance claim rationale

This ST claims conformance to security objectives and security requirements by "strict PP conformance" adherence to 'Korean National Protection Profile for Single Sign On V1.0'.

3. Security objectives

3.1 Security objectives for the operational environment

OE.PHYSICAL_CONTROL

The place where SSO agent and SSO server among the TOE components are installed and operated shall be equipped with access control and protection facilities so that only authorized administrator can access.

OE.TRUSTED_ADMIN

The authorized administrator of the TOE shall be non-malicious users, have appropriately trained for the TOE management functions and accurately fulfill the duties in accordance with administrator guidelines.

OE.LOG_BACKUP

The authorized administrator shall periodically checks a spare space of audit data storage in case of the audit data loss, and carries out the audit data backup (external log server or separate storage device, etc.) to prevent audit data loss.

OE.OPERATION_SYSTEM_REINFORCEMENT

The authorized administrator of the TOE shall ensure the reliability and security of the operating system by performing the reinforcement on the latest vulnerabilities of the operating system in which the TOE is installed and operated.

OE.SECURE_DEVELOPMENT

The developer who uses the TOE to interoperate with the end-user identification and authentication function in the operational environment of the business system shall ensure that the security functions of the TOE are securely applied in accordance with the requirements of the manual provided with the TOE.

OE.TIME STAMP

The TOE accurately records incidents related to security by receiving reliable time stamps provided by the TOE operating environment.

OE.DBMS

DBMS that saves the TSF data and audit data is operated in a physically safe environment.

[Table 3-1] Security objectives for the operational environment

4. Extended components definition

4.1 Cryptographic support (FCS)

4.1.1 Random bit generation

Family Behaviour

This family defines requirements for the TSF to provide the capability that generates random bits required for TOE cryptographic operation.

Component leveling

FCS_RBG.1 random bit generation, requires TSF to provide the capability that generates random bits required for TOE cryptographic operation.

Management: FCS_RBG.1

There are no management activities foreseen.

Audit: FCS_RBG.1

There are no auditable events foreseen.

4.1.1.1 FCS_RBG.1 Randombit generation

Hierarchical to No other components.

Dependencies No dependencies.

FCS_RBG.1.1 The TSF shall generate random bits required to generate a cryptographic key using the specified random bit generator that meets the following [assignment: *list of standards*].

4.2 Identification and authentication (FIA)

4.2.1 TOE Internal mutual authentication

Family Behaviour

This family defines requirements for providing mutual authentication between TOE components in the process of user identification and authentication.

Component leveling

FIA_IMA.1 TOE Internal mutual authentication requires that the TSF provides mutual authentication function between TOE components in the process of user identification and authentication.

Management: FIA_IMA.1

There are no management activities foreseen.

Audit: FIA_IMA.1

The following actions are recommended to record if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimum : Success and failure of mutual authentication

4.2.1.1 FIA_IMA.1 TOE Internal mutual authentication

Hierarchical to No other components.

Dependencies No dependencies.

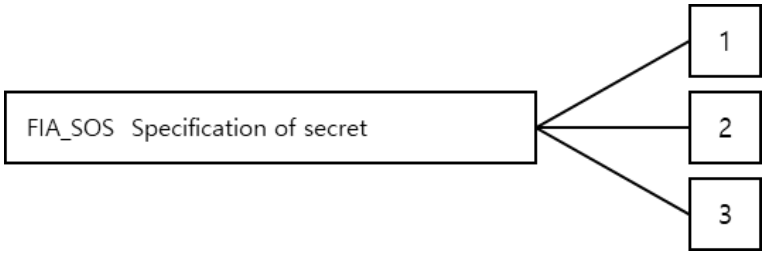
FIA_IMA.1.1 The TSF shall perform mutual authentication between [assignment: *different parts of TOE*] using the [assignment: *authentication protocol*] that meets the following [*assignment: list of standards*].

4.2.2 Specification of Secrets

Family Behaviour

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

Component leveling



The specification of secrets family in CC Part 2 is composed of 2 components. It is now composed of three components, since this PP adds one more component as below.

※ The description on two components included in CC Part 2 is omitted.

FIA_SOS.3 Destruction of secrets requires, that the secret information be destroyed according to the specified destruction method, which can be based on the assigned standard.

Management: FIA_SOS.3

There are no management activities foreseen.

Audit: FIA_SOS.3

The following actions are recommended to record if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimum: Success and failure of the activity

4.2.2.1 FIA_SOS.3 Destruction of Secrets

Hierarchical to No other components.

Dependencies FIA_SOS.2 TSF Generation of secrets

FIA_SOS.3.1 The TSF shall destroy secrets in accordance with a specified secrets destruction method [assignment: *secret destruction method*] that meets the following: [assignment: *list of standards*].

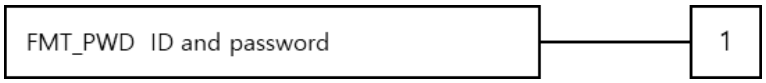
4.3 Security Management (FMT)

4.3.1 ID and pasword

Family Behaviour

This family defines the capability that is required to control ID and password management used in the TOE, and set or modify ID and/or password by authorized users.

Component leveling



FMT_PWD.1 ID and password management, requires that the TSF provides the management function of ID and password.

Management: FMT_PWD.1

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

a) Management of ID and password configuration rules.

Audit: FMT_PWD.1

The following actions are recommended to record if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimum: All changes of the password

4.3.1.1 FMT_PWD.1 Management of ID and password

Hierarchical to No other components.

Dependencies FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

FMT_PWD.1.1 The TSF shall restrict the ability to manage the password of [assignment: list of functions] to [assignment: the authorized identified roles].
1. [assignment: *password combination rules and/or length*]
2. [assignment: *other management such as management of special characters unusable for password, etc.*]

FMT_PWD.1.2 The TSF shall restrict the ability to manage the ID of [assignment: *list of functions*] to [assignment: *the authorized identified roles*].
1. [assignment: *ID combination rules and/or length*]
2. [assignment: *other management such as management of special characters unusable for ID, etc.*]

FMT_PWD.1.3 The TSF shall provide the capability for [selection, choose one of: *setting ID and password when installing, setting password when installing, changing the ID and password when the authorized administrator accesses for the first time, changing the password when the authorized administrator accesses for the first time*].

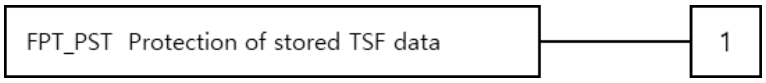
4.4 Protection of the TSF (FPT)

4.4.1 Protection of stored TSF data

Family Behaviour

This family defines rules to protect TSF data stored within containers controlled by the TSF from the unauthorized modification or disclosure.

Component leveling



FPT_PST.1 Basic protection of stored TSF data, requires the protection of TSF data stored in containers controlled by the TSF.

Management: FPT_PST.1

There are no management activities foreseen.

Audit: FPT_PST.1

There are no auditable events foreseen.

4.4.1.1 FPT_PST.1 basic protection of stored TSF data

Hierarchical to No other components.

Dependencies No dependencies.

FPT_PST.1.1 The TSF shall protect [assignment: *TSF data*] stored in containers controlled by the TSF from the unauthorized [selection: *disclosure, modification*].

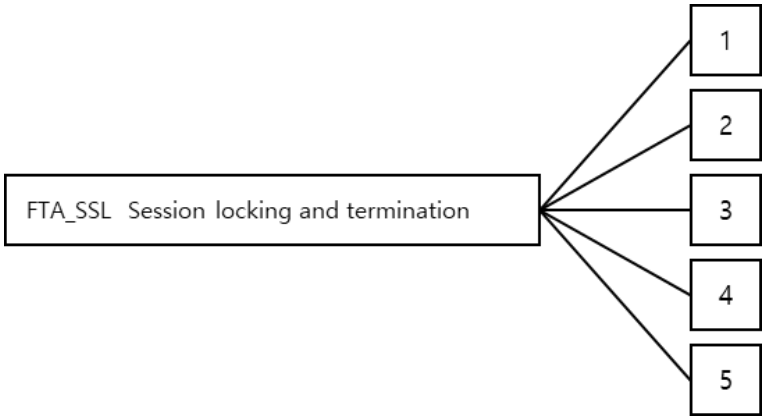
4.5 TOE Access (FTA)

4.5.1 Session locking and termination

Family Behaviour

This family defines requirements for the TSF to provide the capability for TSF-initiated and userinitiated locking, unlocking, and termination of interactive sessions.

Component leveling



In CC Part 2, the session locking and termination family consists of four components. In this PP, it consists of five components by extending one additional component as follows.

※ The relevant description for four components contained in CC Part 2 is omitted.

FTA_SSL.5 The management of TSF-initiated sessions, provides requirements that the TSF locks or terminates the session after a specified time interval of user inactivity.

Management: FTA_SSL.5

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

a) Specification for the time interval of user inactivity that is occurred the session locking and termination for each user

b) Specification for the time interval of default user inactivity that is occurred the session locking and termination

Audit: FTA_SSL.5

The following actions are recommended to record if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimum: Locking or termination of interactive session

4.5.1.1 FTA_SSL.5 Management of TSF-initiated sessions

Hierarchical to No other components.

Dependencies FIA_UAU.1 authentication or No dependencies.

FTA_SSL.5.1 FTA_SSL.5.1The TSF shall [Selection:

- *lock the session and re-authenticate the user before unlocking the session,*
- *terminate] an interactive session after a [assignment: time interval of user inactivity*].

5. Security requirements

The security requirements specify security functional requirements and assurance requirements that must be satisfied by the TOE.

The security functional requirements included in this PP are derived from CC Part 2 and Chapter 4 Extended Components Definition.

5.1 Security functional requirements

The following table summarizes the security functional requirements used in the ST.

Security functional class	Security functional component	
FAU	FAU_ARP.1	Security alarms
	FAU_GEN.1	Audit data generation
	FAU_SAA.1	Potential violation analysis
	FAU_SAR.1	Audit review
	FAU_SAR.3	Selectable audit review
	FAU_STG.3	Action in case of possible audit data loss
	FAU_STG.4	Prevention of audit data loss
FCS	FCS_CKM.1	Cryptographic key generation
	FCS_CKM.2	Cryptographic key distribution
	FCS_CKM.4	Cryptographic key destruction
	FCS_COP.1(1)	Cryptographic operation(Symmetric key)
	FCS_COP.1(2)	Cryptographic operation(Asymmetric key)
	FCS_COP.1(3)	Cryptographic operation(Digital signature generation)
	FCS_COP.1(4)	Cryptographic operation(Digital signature verification)
	FCS_COP.1(5)	Cryptographic operation(HMAC)
	FCS_COP.1(6)	Cryptographic operation(Hash)

	FCS_RBG.1(Extended)	Random bit generatation
FIA	FIA_AFL.1(1)	Authentication failure handling (End-user)
	FIA_AFL.1(2)	Authentication failure handling (Authorized Administrator)
	FIA_IMA.1(Extended)	TOE Internal mutual authentication
	FIA_SOS.1	Verification of secrets
	FIA_SOS.2	TSF Generation of secrets
	FIA_SOS.3(Extended)	Destruction of secrets
	FIA_UAU.2(1)	User authentication before any action (End-user)
	FIA_UAU.2(2)	User authentication before any action (Authorized Administrator)
	FIA_UAU.4(1)	Single-use authentication mechanisms (End-user)
	FIA_UAU.4(2)	Single-use authentication mechanisms (Authorized Administrator)
	FIA_UAU.7(1)	Protected authentication feedback (End-user)
	FIA_UAU.7(2)	Protected authentication feedback (Authorized Administrator)
	FIA_UID.2(1)	User identification before any action (End-user)
	FIA_UID.2(2)	User identification before any action (Authorized Administrator)
FMT	FMT_MOF.1	Management of security functions behaviour
	FMT_MTD.1	Management of TSF data
	FMT_PWD.1(Extended)	Management of ID and password (Authorized Administrator)
	FMT_SMF.1	Specification of management functions

	FMT_SMR.1	Security roles
FPT	FPT_ITT.1	Basic internal TSF data transfer protection
	FPT_PST.1(Extended)	Basic protection of stored TSF data
	FPT_TST.1	TSF testing
FTA	FTA_MCS.2	Per user attribute Limitation on multiple concurrent sessions
	FTA_SSL.5(Extended)	Management of TSF-initiated sessions
	FTA_TSE.1	TOE session establishment

[Table 5-1] Security functional requirements

5.1.1 Security audit (FAU)

5.1.1.1 FAU_ARP.1 Security alarms

Hierarchical to No other components.

Dependencies FAU_SAA.1 Potential violation analysis

FAU_ARP.1.1 The TSF shall take [Send e-mail to authorized administrator] upon detection of a potential security violation.

5.1.1.2 FAU_GEN.1 Audit data generation

Hierarchical to No other components.

Dependencies FPT_STM.1 Reliable time stamps

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) [Refer to the "auditable events" in [Table 5-2] Audit events, [none]]

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the ST [Refer to the contents of "additional audit record" in [Table 5-2] Audit events, [none]]

Security functional component	Auditable event	Additional audit record
FAU_ARP.1	Actions taken due to potential security violations	
FAU_SAA.1	Enabling and disabling of any of the analysis mechanisms, Automated responses performed by the tool	
FAU_STG.3	Actions taken due to exceeding of a threshold	
FAU_STG.4	Actions taken due to the audit storage failure	
FCS_CKM.1	Success and failure of the activity	
FCS_CKM.2	Success and failure of the activity (only applying to key distribution related to the TSF data encryption/decryption)	
FCS_CKM.4	Success and failure of the activity (only applying to key destruction related to the TSF data encryption/decryption)	
FCS_COP.1(1) FCS_COP.1(2) FCS_COP.1(3) FCS_COP.1(4) FCS_COP.1(5) FCS_COP.1(6)	Success and failure, and the type of cryptographic operation(only applying to items related to the issue, storing, verification, and destruction of a token)	

FIA_AFL.1(1) FIA_AFL.1(2)	The reaching of the threshold for the unsuccessful authentication attempts and the actions taken, and the subsequent, if appropriate, restoration to the normal state	
FIA_SOS.2	Rejection by the TSF of any tested secret	
FIA_SOS.3 (Extended)	Success and failure of the activity(applicable to the destruction of SSO token only)	
FIA_UAU.2(1) FIA_UAU.2(2)	All use of the authentication mechanism	
FIA_UAU.4(1) FIA_UAU.4(2)	Attempts to reuse authentication data	
FIA_UID.2(1) FIA_UID.2(2)	All use of the administrator identification mechanism, including the administrator identity provided	
FMT_MOF.1	All modifications in the behaviour of the functions in the TSF	
FMT_MTD.1	All modifications to the values of TSF data	Modified values of TSF data
FMT_PWD.1 (Extended)	All changes of the password	Modified values of TSF data
FMT_SMF.1	Use of the management functions	
FMT_SMR.1	Modifications to the user group of rules divided	
FPT_TST.1	Execution of the TSF self tests and the results of the tests	Modified TSF data in case of integrity violation
FTA_MCS.2	Denial of a new session based on the limitation of multiple concurrent sessions	
FTA_SSL.5 (Extended)	Locking or termination of interactive session	
FTA_TSE.1	Denial of a session establishment due to the session establishment mechanism All attempts at establishment of a user session	

[Table 5-2] Audit events

5.1.1.3 FAU_SAA.1 Potential violation analysis

Hierarchical to No other components.

Dependencies FAU_GEN.1 Audit data generation

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

FAU_SAA.1.2 The TSF shall enforce the following rules for monitoring audited events.
a) Accumulation or combination of [
- Start-up and shutdown of the audit functions
- Administrator authentication failure
- Administrator and end-user authorization fail exceeds allowed number
- Audit storage capacity exceeded
- Administrator access control policy violation
- self-test failure of the validated cryptographic module
- Integrity test fail of the TOE
- License verification failure
] known to indicate a potential security violation
b) [none]

5.1.1.4 FAU_SAR.1 Audit review

Hierarchical to No other components.

Dependencies FAU_GEN.1 Audit data generation

FAU_SAR.1.1 The TSF shall provide [authorized administrator] with the capability to read [all the audit data] 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.

5.1.1.5 FAU_SAR.3 Selectable audit review

Hierarchical to No other components.

Dependencies FAU_SAR.1 Audit review

FAU_SAR.3.1 The TSF shall provide the ability to apply [Session ID, Detail, identify of subject(End-user or Administrator ID, End-user or Administrator IP, TOE Components), Date and time of the event, case type, case results (success or fail) of AND condition] of audit data based on [default sort by date and time. Optionally, sort descending and ascending by OID, Audit code, requestor, requestip, result, date, session id].

5.1.1.6 FAU_STG.3 Action in case of possible audit data loss

Hierarchical to No other components.

Dependencies FAU_STG.1 Protected audit trail storage

FAU_STG.3.1 The TSF shall [Notification to the authorized administrator, [none] if the audit trail exceeds [the threshold set by the authorized administrator (default value 50%, the range of values that the authorized administrator is able to set 50~80%)].

5.1.1.7 FAU_STG.4 Prevention of audit data loss

Hierarchical to FAU_STG.3 Action in case of possible audit data loss

Dependencies FAU_STG.1 Protected audit trail storage

FAU_STG.4.1 The TSF shall *ignore audited events* and [Notification to the authorized administrator, [None]] if the audit trail is full.

Application notes

The percentage of threshold saturation of the audited storage capacity set by an authorized administrator (the range of values set by the authorized administrator 81% to 100%), if threshold saturation defaults exceed 80%, ignore the audited event and send a mail to the authorized administrator.

5.1.2 Cryptographic support (FCS)

5.1.2.1 FCS_CKM.1(1) Cryptographic key generation(1)

Hierarchical to No other components.

Dependencies [FCS_CKM.2 Cryptographic key distribution, or
FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [HASH_DRBG] and specified cryptographic key sizes [256bit] that meet the following: [ISO/IEC 18031(2011), NIST SP 800-90].

5.1.2.2 FCS_CKM.1(2) Cryptographic key generation(2)

Hierarchical to No other components.

Dependencies [FCS_CKM.2 Cryptographic key distribution, or
FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [RSAES] and specified cryptographic key sizes [2048bit] that meet the following: [ISO/IEC 18033-2(2006)].

5.1.2.3 FCS_CKM.2 Cryptographic key distribution

Hierarchical to No other components.

Dependencies [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [RSAES 2048] that meets the following: [ISO/IEC 18033-2(2006)].

5.1.2.4 FCS_CKM.4 Cryptographic key destruction

Hierarchical to No other components.

Dependencies [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [overwrite with 0x00] that meets the following: [none].

5.1.2.5 FCS_COP.1(1) Cryptographic operation (Symmetric key)

Hierarchical to No other components.

Dependencies [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [Encryption Between Components] in accordance with a specified cryptographic algorithm [SEED-CBC] and cryptographic key sizes [128 Bit] that meet the following: [KO-12.0004/R1(2005), TTAS.KO-12.0025(2003)].

5.1.2.6 FCS_COP.1(2) Cryptographic operation (Asymmetric key)

Hierarchical to No other components.

Dependencies [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [Public key Cryptographic operation] in accordance with a specified cryptographic algorithm [RSAES] and cryptographic key sizes [2048 Bit] that meet the following: [ISO/IEC 18033-2(2006)].

5.1.2.7 FCS_COP.1(3) Cryptographic operation (Digital signature generation)

Hierarchical to No other components.

Dependencies [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [Digital Signature generation] in accordance with a specified cryptographic algorithm [RSA-PSS 2048] and cryptographic key sizes [2048 Bit] that meet the following: [ISO/IEC 14888-2(2008)].

5.1.2.8 FCS_COP.1(4) Cryptographic operation (Digital signature verification)

Hierarchical to No other components.

Dependencies [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [Digital Signature verification] in accordance with a specified cryptographic algorithm [RSA-PSS 2048] and cryptographic key sizes [2048 Bit] that meet the following: [ISO/IEC 14888-2(2008)].

5.1.2.9 FCS_COP.1(5) Cryptographic operation (HMAC)

Hierarchical to No other components.

Dependencies [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [Generate message authentication code, Mutual authentication between the TOE components, Verification of authentication token] in accordance with a specified cryptographic algorithm [HMAC-SHA256] and cryptographic key sizes [256 Bit] that meet the following: [ISO/IEC 9797-2(2011)].

5.1.2.10 FCS_COP.1(6) Cryptographic operation (Hash)

Hierarchical to No other components.

Dependencies [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [TOE Server and TOE Agent module integrity verification] in accordance with a specified cryptographic algorithm [SHA-256] and cryptographic key sizes [256 Bit] that meet the following: [ISO/IEC 10118-3(2004), ISO/IEC 10118-3 Amd 1 (2006)].

5.1.2.11 FCS_RBG.1(Extended) Random bit generation

Hierarchical to No other components.

Dependencies No dependencies.

FCS_RBG.1.1 The TSF shall generate random bits required to generate an cryptographic key using the specified random bit generator that meets the following [ISO/IEC 18031(2011), NIST SP 800-90].

5.1.3 Identification and authentication (FIA)

5.1.3.1 FIA_AFL.1(1) Authentication failure handling(End-user)

Hierarchical to No other components.

Dependencies FIA_UAU.1 Timing of authentication

FIA_AFL.1.1 The TSF shall detect when *[5]* unsuccessful authentication attempts occur related to [authentication of end-user].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been *met,* the TSF shall [lock account for disabled 5 minutes].

5.1.3.2 FIA_AFL.1(2) Authentication failure handling(Authorized Administrator)

Hierarchical to No other components.

Dependencies FIA_UAU.1 Timing of authentication

FIA_AFL.1.1 The TSF shall detect when *[5]* unsuccessful authentication attempts occur related to [authentication of administrator].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been *met,* the TSF shall [lock account for disabled 5 minutes].

5.1.3.3 FIA_IMA.1 Internal mutual authentication

Hierarchical to No other components.

Dependencies No dependencies.

FIA_IMA.1.1 The TSF shall perform mutual authentication between [KSignAccess Server and KSignAccess Agent] using the [own authentication protocol] that meets the following [None].

5.1.3.4 FIA_SOS.1 Verification of secrets

Hierarchical to No other components.

Dependencies No dependencies.

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

a) Allowable characters

- English letter (26 letters: a~z)
- Number (10 letters: 0~9)
- Special character that can be input by using a keyboard (27 letters: .-/+=_,:!@#$%^*()~{}[]|<>;&)

b) Min/Max length

- 9 ~ 16 digits

c) Combination rules

- At least 1 Letter to English letter, Number, Special character

5.1.3.5 FIA_SOS.2 TSF Generation of secrets

Hierarchical to No other components.

Dependencies No dependencies.

FIA_SOS.2.1 TSF shall provide a mechanism to generate **an authentication token** that meet [authentication time (time stamp), end-user IP, authentication method, UID, HMAC].

FIA_SOS.2.2 TSF shall be able to enforce the use of TSF-generated **authentication token** for [end-user login].

5.1.3.6 FIA_SOS.3(Extended) Destruction of secrets

Hierarchical to No other components.

Dependencies FIA_SOS.2 TSF Generation of secrets

FIA_SOS.3.1 The TSF shall destroy **authentication tokens** in accordance with a specified **authentication token** destruction method [overwrite data with 0x30] that meets the following: [None].

5.1.3.7 FIA_UAU.2(1) User authentication before any action (End-user)

Hierarchical to FIA_UAU.1 Timing of authentication

Dependencies FIA_UID.1 Timing of identification

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

5.1.3.8 FIA_UAU.2(2) User authentication before any action (Authorized Administrator)

Hierarchical to FIA_UAU.1 Timing of authentication

Dependencies FIA_UID.1 Timing of identification

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

5.1.3.9 FIA_UAU.4(1) Single-use authentication mechanisms (End-user)

Hierarchical to No other components.

Dependencies No dependencies.

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to [end-user session information, authentication token information].

5.1.3.10 FIA_UAU.4(2) Single-use authentication mechanisms (Authorized Administrator)

Hierarchical to No other components.

Dependencies No dependencies.

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to [authorized administrator session information].

5.1.3.11 FIA_UAU.7(1) Protected authentication feedback (End-user)

Hierarchical to No other components.

Dependencies FIA_UAU.1 Timing of authentication

FIA_UAU.7.1 The TSF shall provide only [the following list of feedback] to the **End-user** while the authentication is in progress.

a) Password being entered are masked (password masking with ●) to prevent them from being disclosed on the screen.

b) In case of failure of identification and authentication, feedbacks on the reason for the failure are not provided.

5.1.3.12 FIA_UAU.7(2) Protected authentication feedback (Authorized Administrator)

Hierarchical to No other components.

Dependencies FIA_UAU.1 Timing of authentication

FIA_UAU.7.1 The TSF shall provide only [the following list of feedback] to the **Authorized Administrator** while the authentication is in progress.

a) Password being entered are masked (password masking with ●) to prevent them from being disclosed on the screen.

b) In case of failure of identification and authentication, feedbacks on the reason for the failure are not provided.

5.1.3.13 FIA_UID.2(1) User identification before any action (End-user)

Hierarchical to FIA_UID.1 Timing of identification

Dependencies No dependencies.

FIA_UID.2.1 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that **end-user**.

5.1.3.14 FIA_UID.2(2) User identification before any action (Authorized Administrator)

Hierarchical to FIA_UID.1 Timing of identification

Dependencies No dependencies.

FIA_UID.2.1 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that **authorized administrator**.

5.1.4 Security management (FMT)

5.1.4.1 FMT_MOF.1 Management of security functions behaviour

Hierarchical to No other components.

Dependencies FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MOF.1.1 The TSF shall restrict the ability to conduct ***management actions*** the functions in [[Table 5-3] list of security functions management] to [assignment: the authorised identified roles].

SFR Component	Security Function	Administrator Type
FAU_ARP.1	Management of actions (addition, removal, modification) to be taken	Top Administrator
FAU_SAA.1	Maintenance of the rules (addition, removal and modification of the rules in the rule group)	Top Administrator
FAU_SAR.1	Maintenance (deletion, modification, addition) of the group of users with read access right to the audit records	Top Administrator, Audit Administrator
FAU_STG.3	Maintenance (deletion, modification, addition) of actions to be taken in case of imminent audit storage failurer	Top Administrator
FAU_STG.4	Maintenance (deletion, modification, addition) of actions to be taken in case of audit storage failure.	Top Administrator
FIA_AFL.1(1)	Management of the threshold for unsuccessful authentication attempts	Top Administrator
FIA_AFL.1(2)	Management of actions to be taken in the event of an authentication failure	Top Administrator
FIA_SOS.1	Management of the metric used to verify the secrets	Top Administrator
FIA_UAU.2	Management of the authentication data by an administrator	Top Administrator
FIA_UID.2	Management of the administrator and end-user identities	Top Administrator
FMT_MOF.1	Management of the group of roles that can interact with the functions in the TSF	Top Administrator
FMT_MTD.1	Management of the group of roles that can interact	Top Administrator

	with the TSF data	
FMT_SMR.1	Management of the group of users that are part of a role.	Top Administrator
FPT_ITT.1	Management of the mechanism used to provide the protection of the data in transit between different parts of the TSF	Top Administrator
FPT_TST.1	Management of the conditions under which TSF self testing occurs, such as during intial start-up. Regular interval, or under specified conditions Management of the time interval if appropriate	Top Administrator

[Table 5-3] List of security functions management

5.1.4.2 FMT_MTD.1 Management of TSF data

Hierarchical to No other components.

Dependencies FMT_SMF.1 Specification of Management Functions
FMT_SMR.1 Security roles

FMT_MTD.1.1 The TSF shall restrict the ability to ***manage*** the [[Table 5-4] list of TSF data] to [authorized administrator].

SFR Component	Management Function	Administrator Type
FAU_STG.3	Maintenance (deletion, modification, addition) of actions to be taken in case of imminent audit storage failure	Top Administrator
FAU_STG.4	Maintenance (deletion, modification, addition) of actions to be taken in case of audit storage failure.	Top Administrator
FIA_AFL.1	Management of the threshold for unsuccessful authentication attempts	Top Administrator
FIA_UAU.2	Management of the authentication data by an administrator,	Top Administrator

	Management of the authentication data by the associated end-user	
FIA_UID.2	Management of the administrator and end-user identities	Top Administrator
FMT_MOF.1	Management of the group of roles that can interact with the functions in the TSF	Top Administrator
FMT_MTD.1	Management of the group of roles that can interact with the TSF data	Top Administrator
FMT_SMR.1	Management of the group of users that are part of a role.	Top Administrator
FPT_ITT.1	Management of the mechanism used to provide the protection of the data in transit between different parts of the TSF	Top Administrator
FTA_TSE.1	Management of the conditions under which TSF self testing occurs, such as during intial start-up. regular interval, or under specified conditions	Top Administrator

[Table 5-4] list of TSF data

5.1.4.3 FMT_PWD.1(Extended) Management of ID and password (Authorized Administrator)

Hierarchical to No other components.

Dependencies FMT_SMF.1 Specification of Management Functions
FMT_SMR.1 Security roles

FMT_PWD.1.1 The TSF shall restrict the ability to manage the password of [N/A] to [the authorized administrator].

FMT_PWD.1.2 The TSF shall restrict the ability to manage the ID of [N/A] to [the authorized administrator].

FMT_PWD.1.3 The TSF shall provide the capability for *changing the password when the authorized administrator accesses for the first time*.

5.1.4.4 FMT_SMF.1 Specification of Management Functions

Hierarchical to No other components.

Dependencies No dependencies.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:
[Management of functions of the TSF: Management functions specified in FMT_MOF.1
Management of TSF data: Management functions spectified in FMT_MTD.1]

5.1.4.5 FMT_SMR.1 Security role

Hierarchical to No other components.

Dependencies FIA_UID.1 Timing of identification

FMT_SMR.1.1 The TSF shall maintain the roles [as the authorized identified roles]
- Top Administrator
- Audit Administrator

FMT_SMR.1.2 The TSF shall be able to associate users and their **roles defined in FMT_SMR.1.1**.

5.1.5 Protection of the TSF (FPT)

5.1.5.1 FPT_ITT.1 Basic Internal TSF data transfer protection

Hierarchical to No other components.

Dependencies No dependencies.

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

5.1.5.2 FPT_PST.1(Extended) Basic protection of stored TSF data

Hierarchical to No other components.

Dependencies No dependencies.

The TSF should protect the [TSF data] stored in the repository, which is controlled by the TSF, from unauthorized *exposure and modification*.

5.1.5.3 FPT_TST.1 TSF testing

Hierarchical to No other components.

Dependencies No dependencies.

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

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

FPT_TST.1.3 The TSF shall provide **authorized administrators** with the capability to verify the integrity of *TSF*.

5.1.6 TOE access (FTA)

5.1.6.1 FTA_MCS.2 Per user attribute limitation on multiple concurrent sessions

Hierarchical to FTA_MCS.1 Basic limitation on multiple concurrent sessions

Dependencies FIA_UID.1 Timing of identification

FTA_MCS.2.1 The TSF shall restrict the maximum number of concurrent sessions that belong to the same user according to the rules [restriction to one for the maximum number of concurrent sessions for administrator management access session, prohibition of same administrator both concurrent connections of management access session and local access session that belong to the same user]

FTA_MCS.2.2 The TSF shall enforce, by default, a limit of [1] sessions per user.

5.1.6.2 FTA_SSL.5(Extended) Management of TSF-initiated sessions

Hierarchical to No other components.

Dependencies FIA_UAU.1 Authentication
or No dependencies.

FTA_SSL.5.1 The TSF shall [terminate] an interactive session after a [10 minutes].

5.1.6.3 FTA_TSE.1 TOE session establishment

Hierarchical to No other components.

Dependencies No dependencies.

FTA_TSE.1.1 The TSF shall be able to deny **administrator's management access** session establishment based on [connection IP, *whether or not to activate the management access session of administrator account with the same privilege*].

5.2 Security assurance requirements

Assurance requirements of this Protection Profile are comprised of assurance components in CC part 3, and the evaluation assurance level is EAL1+. The following table summarizes assurance components.

Security assurance class	Security assurance component	
Security Target evaluation	ASE_INT.1	ST introduction
	ASE_CCL.1	Conformance claims
	ASE_OBJ.1	Security objectives for the operational environment
	ASE_ECD.1	Extended components definition
	ASE_REQ.1	Stated security requirements
	ASE_TSS.1	TOE summary specification
Development	ADV_FSP.1	Basic functional specification
Guidance documents	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
Life-cycle support	ALC_CMC.1	Labelling of the TOE
	ALC_CMS.1	TOE CM coverage
Tests	ATE_FUN.1	Functional testing
	ATE_IND.1	Independent testing - conformance
Vulnerability assessment	AVA_VAN.1	Vulnerability survey

[Table 5-5] Security assurance requirements

5.2.1 Security Target evaluation

5.2.1.1 ASE_INT.1 ST introduction

Dependencies No dependencies.

Developer action elements

ASE_INT.1.1D The developer shall provide an ST introduction.

Content and presentation elements

ASE_INT.1.1C The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description.

ASE_INT.1.2C The ST reference shall uniquely identify the ST.

ASE_INT.1.3C The TOE reference shall uniquely identify the TOE.

ASE_INT.1.4C The TOE overview shall summarise the usage and major security features of the TOE.

ASE_INT.1.5C The TOE overview shall identify the TOE type.

ASE_INT.1.6C The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE.

ASE_INT.1.7C The TOE description shall describe the physical scope of the TOE.

ASE_INT.1.8C The TOE description shall describe the logical scope of the TOE.

Evaluator action elements

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

ASE_INT.1.2E The evaluator shall confirm that the TOE reference, the TOE overview, and the TOE description are consistent with each other.

5.2.1.2 ASE_CCL.1 Conformance claims

Dependencies ASE_INT.1 ST Introduction
ASE_ECD.1 Extended components definition
ASE_REQ.1 Stated security requirements

Developer action elements

ASE_CCL.1.1D The developer shall provide a conformance claim.

ASE_CCL.1.2D The developer shall provide a conformance claim rationale.

Content and presentation elements

ASE_CCL.1.1C The conformance claim shall contain a CC conformance claim that identifies the version of the CC to which the ST and the TOE claim conformance.

ASE_CCL.1.2C The CC conformance claim shall describe the conformance of the ST to CC Part 2 as either CC Part 2 conformant or CC Part 2 extended.

ASE_CCL.1.3C The CC conformance claim shall describe the conformance of the ST to CC Part 3 as either CC Part 3 conformant or CC Part 3 extended.

ASE_CCL.1.4C The CC conformance claim shall be consistent with the extended components definition.

ASE_CCL.1.5C The conformance claim shall identify all PPs and security requirement packages to which the ST claims conformance.

ASE_CCL.1.6C The conformance claim shall describe any conformance of the ST to a package as either package-conformant or package-augmented.

ASE_CCL.1.7C The conformance claim rationale shall demonstrate that the TOE type is consistent with the TOE type in the PPs for which conformance is being claimed.

ASE_CCL.1.8C The conformance claim rationale shall demonstrate that the statement of the security problem definition is consistent with the statement of the security problem definition in the PPs for which conformance is being claimed.

ASE_CCL.1.9C The conformance claim rationale shall demonstrate that the statement of security objectives is consistent with the statement of security objectives in the PPs for which conformance is being claimed.

ASE_CCL.1.10C The conformance claim rationale shall demonstrate that the statement of security requirements is consistent with the statement of security requirements in the PPs for which conformance is being claimed.

Evaluator action elements

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

5.2.1.3 ASE_OBJ.1 Security objectives for the operational environment

Dependencies No dependencies.

Developer action elements

ASE_OBJ.1.1D The developer shall provide a statement of security objectives.

Content and presentation elements

ASE_OBJ.1.1C The statement of security objectives shall describe the security objectives for the operational environment.

Evaluator action elements

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

5.2.1.4 ASE_ECD.1 Extended components definition

Dependencies No dependencies.

Developer action elements

ASE_ECD.1.1D The developer shall provide a statement of security requirements.

ASE_ECD.1.2D The developer shall provide an extended components definition.

Content and presentation elements

ASE_ECD.1.1C The statement of security requirements shall identify all extended security requirements.

ASE_ECD.1.2C The extended components definition shall define an extended component for each extended security requirement.

ASE_ECD.1.3C The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes.

ASE_ECD.1.4C The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation.

ASE_ECD.1.5C The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements can be demonstrated.

Evaluator action elements

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

ASE_ECD.1.2E The evaluator shall confirm that no extended component can be clearly expressed using existing components.

5.2.1.5 ASE_REQ.1 Stated security requirements

Dependencies ASE_ECD.1 Extended components definition

Developer action elements

ASE_REQ.1.1D The developer shall provide a statement of security requirements.

ASE_REQ.1.2D The developer shall provide a security requirements rationale.

Content and presentation elements

ASE_REQ.1.1C The statement of security requirements shall describe the SFRs and the SARs.

ASE_REQ.1.2C All subjects, objects, operations, security attributes, external entities and other terms that are used in the SFRs and the SARs shall be defined.

ASE_REQ.1.3C The statement of security requirements shall identify all operations on the security requirements.

ASE_REQ.1.4C All operations shall be performed correctly.

ASE_REQ.1.5C Each dependency of the security requirements shall either be satisfied, or the security requirements rationale shall justify the dependency not being satisfied.

ASE_REQ.1.6C The statement of security requirements shall be internally consistent.

Evaluator action elements

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

5.2.1.6 ASE_TSS.1 TOE summary specification

Dependencies ASE_INT.1 ST introduction
ASE_REQ.1 Stated security requirements
ADV_FSP.1 Basic functional specification

Developer action elements

ASE_TSS.1.1D The developer shall provide a TOE summary specification.

Content and presentation elements

ASE_TSS.1.1C The TOE summary specification shall describe how the TOE meets each SFR.

Evaluator action elements

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

ASE_TSS.1.2E The evaluator shall confirm that the TOE summary specification is consistent with the TOE overview and the TOE description.

5.2.2 Development

5.2.2.1 ADV_FSP.1 Basic functional specification

Dependencies No dependencies.

Developer action elements

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

ADV_FSP.1.2D The developer shall provide a tracing from the functional specification to the SFRs.

Content and presentation elements

ADV_FSP.1.1C The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.2C The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.3C The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.

ADV_FSP.1.4C The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

Evaluator action elements

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

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

5.2.3 Guidance documents

5.2.3.1 AGD_OPE.1 Operational user guidance

Dependencies ADV_FSP.1 Basic functional specification

Developer action elements

AGD_OPE.1.1D The developer shall provide operational user guidance.

Content and presentation elements

AGD_OPE.1.1C The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2C The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3C The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4C The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5C The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AGD_OPE.1.6C The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7C The operational user guidance shall be clear and reasonable.

Evaluator action elements

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

5.2.3.2 AGD_PRE.1 Preparative procedures

Dependencies No dependencies.

Developer action elements

AGD_PRE.1.1D The developer shall provide the TOE including its preparative procedures.

Content and presentation elements

AGD_PRE1.1C The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE1.2C The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

Evaluator action elements

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

AGD_PRE.1.2E The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

5.2.4 Life-cycle support

5.2.4.1 ALC_CMC.1 TOE Labelling of the TOE

Dependencies ALC_CMS.1 TOE CM coverage

Developer action elements

ALC_CMC.1.1D The developer shall provide the TOE and a reference for the TOE.

Content and presentation elements

ALC_CMC.1.1C The TOE shall be labelled with its unique reference.

Evaluator action elements

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

5.2.4.2 ALC_CMS.1 TOE CM coverage

Dependencies No dependencies.

Developer action elements

ALC_CMS.1.1D The developer shall provide a configuration list for the TOE.

Content and presentation elements

ALC_CMS.1.1C The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.

ALC_CMS.1.2C The configuration list shall uniquely identify the configuration items.

Evaluator action elements

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

5.2.5 Tests

5.2.5.1 ATE_FUN.1 Functional testing

Dependencies ATE_COV.1 Evidence of coverage

Developer action elements

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

ATE_FUN.1.2D The developer shall provide test documentation.

Content and presentation elements

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

ATE_FUN.1.2C The test plans shall identify the tests to be performed and describe the scenarios for performing each test. These scenarios shall include any ordering dependencies on the results of other tests.

ATE_FUN.1.3C The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE_FUN.1.4C The actual test results shall be consistent with the expected test results.

Evaluator action elements

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

5.2.5.2 ATE_IND.1 Independent testing - conformance

Dependencies ADV_FSP.1 Basic functional specification
AGD_OPE.1 Operational user guidance
AGD_PRE.1 Preparative procedures

Developer action elements

ATE_IND.1.1D The developer shall provide the TOE for testing.

Content and presentation elements

ATE_IND.1.1C The TOE shall be suitable for testing.

Evaluator action elements

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

ATE_IND.1.2E The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

5.2.6 Vulnerability assessment

5.2.6.1 AVA_VAN.1 Vulnerability survey

Dependencies ADV_FSP.1 Basic functional specification
AGD_OPE.1 Operational user guidance
AGD_PRE.1 Preparative procedures

Developer action elements

AVA_VAN.1.1D The developer shall provide the TOE for testing.

Content and presentation elements

AVA_VAN.1.1C The TOE shall be suitable for testing.

Evaluator action elements

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

AVA_VAN.1.2E The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.1.3E The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

5.3 Security requirements rationale

5.3.1 Dependency rationale of security functional requirements

No.	Security functional requirements	Dependency	Reference No.
1	FAU_ARP.1	FAU_SAA.1	3
2	FAU_GEN.1	FPT_STM.1	OE.Time stamp
3	FAU_SAA.1	FAU_GEN.1	2
4	FAU_SAR.1	FAU_GEN.1	2
5	FAU_SAR.3	FAU_SAR.1	4
6	FAU_STG.3	FAU_STG.1	OE.DBMS
7	FAU_STG.4	FAU_STG.1	OE.DBMS
8	FCS_CKM.1(1)	[FCS_CKM.2 or FCS_COP.1]	10, 12, 16
		FCS_CKM.4	11

9	FCS_CKM.1(2)	[FCS_CKM.2 or FCS_COP.1]	13, 14, 15
		FCS_CKM.4	11
10	FCS_CKM.2	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	8
		FCS_CKM.4	11
11	FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	8
12	FCS_COP.1(1)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	8
		FCS_CKM.4	11
13	FCS_COP.1(2)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9
		FCS_CKM.4	11
14	FCS_COP.1(3)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9
		FCS_CKM.4	11
15	FCS_COP.1(4)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9
		FCS_CKM.4	11
16	FCS_COP.1(5)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	8
		FCS_CKM.4	11
17	FCS_COP.1(6)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	-
		FCS_CKM.4	-
18	FCS_RBG.1	-	-
19	FIA_IMA.1	-	-
20	FIA_AFL.1(1)	FIA_UAU.1	25
21	FIA_AFL.1(2)	FIA_UAU.1	26
22	FIA_SOS.1	-	-
23	FIA_SOS.2	-	-
24	FIA_SOS.3	FIA_SOS.2	23
25	FIA_UAU.2(1)	FIA_UID.1	31
26	FIA_UAU.2(2)	FIA_UID.1	32

27	FIA_UAU.4(1)	-	-
28	FIA_UAU.4(2)	-	-
29	FIA_UAU.7(1)	FIA_UAU.1	25
30	FIA_UAU.7(2)	FIA_UAU.1	26
31	FIA_UID.2(1)	-	-
32	FIA_UID.2(2)	-	-
33	FMT_MOF.1	FMT_SMF.1	36
		FMT_SMR.1	37
34	FMT_MTD.1	FMT_SMF.1	36
		FMT_SMR.1	37
35	FMT_PWD.1	FMT_SMF.1	36
		FMT_SMR.1	37
36	FMT_SMF.1	-	-
37	FMT_SMR.1	FIA_UID.1	32
38	FPT_ITT.1	-	-
39	FPT_PST.1	-	-
40	FPT_STM.1	-	-
41	FPT_TST.1	-	-
42	FTA_MCS.2	FIA_UID.1	32
43	FTA_SSL.5	FIA_UAU.2 or No dependencies	25, 26
44	FTA_TSE.1	-	-

[Table 5-6] Rationale for the dependency of the security functional requirements

- FAU_GEN.1 has a dependent relationship with FPT_STM.1 and this uses reliable time stamp provided by the TOE operating environment and records tests related to security. Therefore, it satisfies the security goal time stamp for operating environments.

- FAU_STG.3 and FAU_STG.4 have dependent relationships with FAU_STG.1 and this is satisfied by the DBMS operating environment.
- FCS_COP.1(6) have dependent relationships with FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 and FCS_CKM.4 and this is satisfied because the hash algorithm does not use cryptographic keys.
- FIA_AFL.1(1), FIA_UAU.7(1) have dependent relationships with FIA_UAU.1 and this is satisfied by FIA_UAU.2(1) that have hierarchal relations with FIA_UAU.1
- FIA_AFL.1(2), FIA_UAU.7(2) have dependent relationships with FIA_UAU.1 and this is satisfied by FIA_UAU.2(2) that have hierarchal relations with FIA_UAU.1
- FTA_SSL.5 have dependent relationships with FIA_UAU.1 and this is satisfied by FIA_UAU.2(1), FIA_UAU.2(2) that have hierarchal relations with FIA_UAU.1
- FIA_UAU.2(1) have dependent relationships with FIA_UID.1 and this is satisfied by FIA_UID.2(1) that have hierarchal relations with FIA_UID.1
- FIA_UAU.2(2), FMT_SMR.1, FTA_MCS.2 have dependent relationships with FIA_UID.1 and this is satisfied by FIA_UID.2(1) that have hierarchal relations with FIA_UID.1

5.3.2 Dependency rationale of security assurance requirements

The dependency of EAL1 assurance package provided in the CC is already satisfied, the rationale is omitted.

The augmented SAR ATE_FUN.1 has dependency on ATE_COV.1. but, ATE_FUN.1 is augmented to require developer testing in order to check if the developer correctly performed and documented the tests in the test documentation, ATE_COV.1 is not included in this ST since it is not necessarily required to show the correspondence between the tests and the TSFIs.

6. TOE Summary Specification

6.1 Security Audit (FAU)

6.1.1 Security Alert

The TOE applies a rule set indicating a potential security violation to the audit log to perform security alart in case of violation event.

- Start-up and shutdown of the audit functions
- Administrator authentication failure
- Administrator and end-user authorization fail exceeds allowed number
- Audit storage capacity exceeded
- Administrator access control policy violation
- Self-test failure of the validated cryptographic module
- Integrity test fail of the TOE
- License verification failure

When the TOE detects a potential security violation, it generates an audit record and sends an alarm (warning mail) to the authorized administrator.

6.1.2 Audit data generation

Generate audit data for security events of each TOE component. The generated audit data is stored in a DBMS provided by the operating environment. The TOE uses the trusted time stamp provided in the environment where the TOE operates in order to ensure that the audit data is generated sequentially.

Audit events are generated and stored by Session ID, Detail, subject identity (end-user or administrator ID, end-user or administrator IP, TOE components), date and time of event, type of event and success / failure.

The auditable events generated are as follows.

Security functional component	Auditable event	Additional audit record
FAU_ARP.1	Actions taken due to potential security violations	
FAU_SAA.1	Enabling and disabling of any of the analysis mechanisms, Automated responses performed by the tool	
FAU_STG.3	Actions taken due to exceeding of a threshold	
FAU_STG.4	Actions taken due to the audit storage failure	
FCS_CKM.1	Success and failure of the activity	
FCS_CKM.2	Success and failure of the activity (only applying to key distribution related to the TSF data encryption/decryption)	
FCS_CKM.4	Success and failure of the activity (only applying to key destruction related to the TSF data encryption/decryption)	
FCS_COP.1(1) FCS_COP.1(2) FCS_COP.1(3) FCS_COP.1(4) FCS_COP.1(5) FCS_COP.1(6)	Success and failure, and the type of cryptographic operation(only applying to items related to the issue, storing, verification, and destruction of a token)	
FIA_AFL.1(1) FIA_AFL.1(2)	The reaching of the threshold for the unsuccessful authentication attempts and the actions taken, and the subsequent, if appropriate, restoration to the normal state	
FIA_SOS.2	Rejection by the TSF of any tested secret	
FIA_SOS.3(Extended)	Success and failure of the activity(applicable to the destruction of SSO token only)	
FIA_UAU.2(1) FIA_UAU.2(2)	All use of the authentication mechanism	
FIA_UAU.4(1) FIA_UAU.4(2)	Attempts to reuse authentication data	

FIA_UID.2(1) FIA_UID.2(2)	All use of the administrator identification mechanism, including the administrator identity provided	
FMT_MOF.1	All modifications in the behaviour of the functions in the TSF	
FMT_MTD.1	All modifications to the values of TSF data	Modified values of TSF data
FMT_PWD.1(Extended)	All changes of the password	Modified values of TSF data
FMT_SMF.1	Use of the management functions	
FMT_SMR.1	Modifications to the user group of rules divided	
FPT_TST.1	Execution of the TSF self tests and the results of the tests	Modified TSF data in case of integrity violation
FTA_MCS.2	Denial of a new session based on the limitation of multiple concurrent sessions	
FTA_SSL.5(Extended)	Locking or termination of interactive session	
FTA_TSE.1	Denial of a session establishment due to the session establishment mechanism All attempts at establishment of a user session	

[Table 6-1] Audit event

6.1.3 Audit data review

The TOE stores the audit data in the audit trail storage (DBMS) and provides the authorized administrator with the function to inquire all the audit data so as to be suitable for interpreting the information from the audit record. It is possible to query by Session ID, Detail, identity of subject (general user or administrator ID, general user or manager IP, TOE component), date and time of event, type of event and AND condition of success / failure of event.

The authorized administrator (top administrator, audit administrator) can inquire and retrieve the audit data through the security management interface of KSignAccess Server.

6.1.4 Audit repository inspection and security violation response

The audit records generated by the TOE are stored in the repository (DBMS) provided by the TOE operational environment. Only the authorized administrator can access the audit record DB through the repository and perform the audit record clearance work.

The TOE periodically checks the space of the audit record storage and generates an audit record of the excess events when it exceeds the remaining space of the storage set by the authorized administrator and sends an alarm (warning mail) to the authorized administrator. When the audit record storage is possible audit data loss, the TOE ignores the audit contents and sends an alarm (warning mail) to the administrator to protect the audit records.

- The threshold excess rate for the total capacity of audit records storage space is 50% by default, and the range of values that can be set by the administrator is 50 to 80%. Alarm (send warning mail) to an authorized administrator when the threshold is exceeded.
- The threshold saturation ratio for the total capacity of audit records storage space is 81% by default and the range of values that can be set by the administrator is 81% to 100%. Ignoring the audited event upon saturation of the threshold, alerts (send warning mail) to the authorized administrator.

6.1.5 SFR Mapping

SFR to be satisfied: FAU_ARP.1, FAU_GEN.1, FAU_SAA.1, FAU_SAR.1, FAU_SAR.3, FAU_STG.3, FAU_STG.4

6.2 Cryptographic Support (FCS)

The TOE is cryptographic support using the validated cryptographic module KSignCASE64 v2.5 in the policy transmission sector to support the cryptographic functions between the TOE components. Details of the validated cryptographic module included in the TOE are as follows.

Item	Specification
Cryptographic module name	KSignCASE64 v2.5
Developer	KSign Co., Ltd.

Validation date	Oct 05, 2015
Validation level	VSL1
Validation number	CM-103-2020.10

[Table 6-2] Specific information of validated cryptographic module

6.2.1 Cryptographic support

The encryption key used to Encryption/decryption is generated using the server information, the Salt value, and the Process ID. When the KSignAccess Agent is operated, the KSignAccess Agent is certified using the KSignAccess Agent information registered in the KSignAccess server. If the authentication result is successful, the KSignAccess Server certificate is included in the response information. Afterwards, encryption keys are created by KSignAccess Agent and are applied with RSAES standard and encrypted with a public key on the KSignAccess Server certificate before being distributed to the KSignAccess server. The generated and distributed encryption keys are encrypted and stored securely by a certificate from the KSignAccess Server.

When communicating between TOE components, the cryptographic key is loaded into memory by each module, decrypted through the private key, and used for encryption / decryption operations.

When generating cryptographic key for Symmetric Key Cryptographic operation, 128bit cryptographic key length and 128bit initial vector are generated by HASH_DRBG algorithm which meets ISO / IEC 18031 (2011) and NIST SP 800-90 standard.

When generating cryptographic key for Asymmetric Key (Public Key) Cryptographic operation, 2048bit cryptographic key length are generated by RSAES algorithm which meets ISO/IEC 18033-2(2006) standard.

When distributing the cryptographic key, the key distribution is performed in a self-implemented with reference to RFC5652 (Cryptographic Message Syntax).

The cryptographic key stored in the memory during key generation, distribution, and operation is overwritten with data 0x00 after expiration of the validity period, and the cryptographic key is destroyed.

- When encryption key related information is deleted :
  System power down, password key clear function call, and trusted channel shutdown

The supported cryptographic algorithms use the validated cryptographic module, and the algorithm information for each application is as follows.

Item		Algorithm	Key length	Standards
mutual authentication (KSignAccess Server ↔ KSignAccess Agent)	Data encryption key (DEK)	SEED-CBC	128bits	TTAS.KO-12.0004/R1(2005), TTAS.KO-12.0025(2003)
	Key exchange	RSAES	2048bits	ISO/IEC 18033-2(2006)
	Integrity (Agent)	HMAC-SHA256	256bits	ISO/IEC 9797-2(2011)
	Integrity (Server)	RSA-PSS	2048bits	ISO/IEC 14888-2(2008)
Token request	Encryption/ Decryption	SEED-CBC	128bits	TTAS.KO-12.0004/R1(2005), TTAS.KO-12.0025(2003)
	Integrity	HMAC-SHA256	256bits	ISO/IEC 9797-2(2011)
Authenticate token	Encryption/ Decryption	SEED-CBC	128bits	TTAS.KO-12.0004/R1(2005), TTAS.KO-12.0025(2003)
	Integrity	HMAC-SHA256	256bits	ISO/IEC 9797-2(2011)
Key encryption key (KEK)		RSAES	2048bits	ISO/IEC 18033-2(2006)
Administrator / End-user password store		SHA256	256bits	ISO/IEC 10118-3(2004), ISO/IEC 10118-3 Amd 1 (2006)
TOE module integrity		SHA256	256bits	

[Table 6-3] Algorithm information

6.2.2 Random generate

The TOE generates a random bit necessary for cryptographic key generation using the HASH_DRBG algorithm through a random bit generator of KSignCASE64 v2.5, which is a validated cryptographic module whose safety and implementation conformity is validated through a cryptographic module verification system.

6.2.3 SFR Mapping

SFR to be satisfied: FCS_CKM.1, FCS_CKM.2, FCS_CKM.4, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FCS_COP.1(5), FCS_COP.1(6), FCS_RBG.1(Extended)

6.3 Identification and Authentication (FIA)

The TOE identifies all users (administrators and end-users) to access. Until the identification is achieved, the administrator and the IT entity attempting access can not use any functions of the TOE.

All users (administrators and end-users) are identification and authentication through the ID and password-based certification process. Even if the user successfully completes the self-identification and authentication procedure, access is restricted according to the IP address-based access control function.

An authorized administrator accesses the TOE security management interface through a web browser to perform security management of the TOE. At this time, identification and authentication procedures are performed through the login screen, and only the authorized administrator who has been successfully identified and authenticated provides the function to use the security management function provided by the TOE.

6.3.1 Authentication failure response

The TOE protects the TOE from malicious user authentication attempts by providing the user (administrator and end-user) account lock function in user identification and authentication.

In case of administrator, KSignAccess Server prints out "No administrator information does not exist or the password does not match" to the web browser screen so that the authentication mechanism failure reason information can not be known when authentication fails. If the number of authentication failures of the account reaches 5 times, the account lock is performed and the lock is enabled automatically after 5 minutes.

The TOE displays the "administrator account status is locked" to the user attempting to authenticate through the locked account. After outputting the authentication failure cause information window, the authentication is failed and the audit data is generated. A user whose account is unlocked can normally use the functions provided by the TOE upon successful completion of user identification and authentication.

In case of end-user, If the number of authentication failures of the account reaches 5 times, the account lock is performed and the lock is enabled automatically after 5 minutes.

6.3.2 Protection of authentication data

When the end-user accesses the TOE, the identification and authentication process operates on the basis of ID and password. The password entered by the user during the authentication mechanism is displayed as "●" instead of the original character, thereby preventing leakage of the authentication information. In case of authentication failure of the end-user, only the authentication failure message to provide.

Only administrators with IP addresses (up to two) allowed by the authorized administrator (top administrator) can access the TOE by enforcing the security management access control. When the administrator accesses the TOE, the identification and authentication process operates on the basis of ID and password. The password entered by the administrator during the authentication mechanism is displayed as "●" instead of the original character, thereby preventing leakage of the authentication information. In case of authentication failure of the administrator, only the authentication failure message to provide.

6.3.3 Password policy validation

When generating and changing the passwords of the end-user or the authorized administrator, it verifies the allowable characters, combination rule, minimum / maximum length validity of the password value according to the password policy, and enforces each verification criterion.

In the case of an authorized administrator, the security management interface is forced to change the password when performing initial identification and authentication, and the password verification mechanism is the same as the mechanism for generating and changing.

The TOE provides the following specified secret information verification mechanism when generating and changing the password.

- Min/Max length : 9 ~ 16 digits
- Allowable characters : English letter (26 letters: a~z), Number (10 letters: 0~9), Special character that can be input by using a keyboard (27 letters: .-/+=_:!@#$%^*()~{}[]|<>;&)
- At least 1 Letter to English letter, Number, Special character

6.3.4 Authentication data prevents

To prevent the reuse of authentication information, the TOE provides the following.

- Prevent reuse of administrator authentication information : In order to prevent Cross Side Request Fogery (CSRF) attacks, nonce is allocated to each page before authentication of the administrator, and access is restricted when the assigned nonce is not transmitted together.
- Prevent reuse of end-user authentication information : When a end-user accesses KSignAccess Agent, it identification and authentication using the authentication token issued by the server. The Nonce value as the one-time authentication data including the time stamp used when generating the authentication token, and the generated authentication token are immediately discarded without being stored after use.

6.3.5 Mutual authentication

The TOE performs mutual authentication through the self-implemented protocol between KSignAccess Server and KSignAccess Agent, and the detailed mechanism for mutual authentication is as follows.

1. When registering KSignAccess Agent in security management interface, Agent Identifier and Agent Secret values for mutual authentication with KSignAccess Server are generated. KSignAccess Agent generates authentication message (HMAC SHA256) and transmits it to KSignAccess Server through HTTP communication to request mutual authentication.
   - request header -> AccessAgent_Authroization:base64(Agent Gid + Agent Identifier)
   - request Body -> { alg: HS256, signature: hmac(state, Agent Secret), state: state }
2. The KSignAccess Server verifies the mutual authentication request of the KSignAccess Agent through mutual authentication request (HMAC) of the following authentication message received for the mutual authentication request of the KSignAccess Agent.
   - request header validation (Agent Gid, Agent Identifier validation)
   - request body validation (signature validation in body)
3. KSignAccess Server verifies KSignAccess Agent and generates digital signature value (RSAPSS) using KSignAccess Server's private key and sends a response message to KSignAccess Agent.

- response body -> { alg: PS256, signature: RSAPSSwithSHA256(KSignAccess server_state + KSignAccess agent_state, server private key), state: KSignAccess server_state }

4. KSignAccess Agent receives mutual authentication response of KSignAccess Server and verifies digital signature value (RSAPSS) and completes mutual authentication with KSignAccess Server.

- verify_signature((KSignAccess server_state + KSignAccess agent_state), KSignAccess Server Certificate)

6.3.6 Authentication token generate and distruction

The TOE generates the authentication token using the validated cryptographic module KSignCASE64 v2.5 when generating the authentication token, and enforces the use of the authentication token for the login of the end-user.

Detailed information of the validated cryptographic module included in the TOE is as follows.

Item	Specification
Cryptographic module name	KSignCASE64 v2.5
Developer	KSign Co., Ltd.
Validation date	Oct 05, 2015
Validation level	VSL1
Validation number	CM-103-2020.10

[Table 6-4] Specific information of validated cryptographic module

When KSignAccess Agent requests creation of authentication tokens, it generates a state with a single-use authentication value through a validated cryptographic module random number generator and performs encryption operation with SEED-CBC by including the state in the request, and examines integrity of authentication token by generating HMAC data.

The protocol of the authentication token is as follows.

Timestamp	Client IP	AuthenMethod	Subject	Extend_information	HMAC

After the authentication token is used, the authentication token is discarded by overwriting the data with '0x30'.

- When encryption key related information is deleted :
  System power down, password key clear function call, and trusted channel shutdown

6.3.7 SFR Mapping

SFR to be satisfied: FIA_AFL.1(1), FIA_AFL.1(2), FIA_IMA.1(Extended), FIA_SOS.1, FIA_SOS.2, FIA_SOS.3(Extended), FIA_UAU.2(1), FIA_UAU.2(2), FIA_UAU.4(1), FIA_UAU.4(2), FIA_UAU.7(1), FIA_UAU.7(2), FIA_UID.2(1), FIA_UID.2(2)

6.4 Security Management (FMT)

6.4.1 Management of Security functions behaviour

The security management access control function is invoked only when the TOE performs the enforced identification and authentication function successfully. Only the administrator who has allowed the authorized administrator (top administrator) to access the security management interface is allowed.

The TOE provides the authorized administrator with management functions for the following security functions.

SFR Component	Security Function	Administrator Type
FAU_ARP.1	Management of actions (addition, removal, modification) to be taken	Top Administrator
FAU_SAA.1	Maintenance of the rules (addition, removal and modification of the rules in the rule group)	Top Administrator
FAU_SAR.1	Maintenance (deletion, modification, addition) of the group of users with read access right to the audit records	Top Administrator, Audit Administrator
FAU_STG.3	Maintenance (deletion, modification, addition) of actions to be taken in case of imminent audit storage failurer	Top Administrator

FAU_STG.4	Maintenance (deletion, modification, addition) of actions to be taken in case of audit storage failure.	Top Administrator
FIA_AFL.1(1)	Management of the threshold for unsuccessful authentication attempts	Top Administrator
FIA_AFL.1(2)	Management of actions to be taken in the event of an authentication failure	Top Administrator
FIA_SOS.1	Management of the metric used to verify the secrets	Top Administrator
FIA_UAU.2	Management of the authentication data by an administrator	Top Administrator
FIA_UID.2	Management of the administrator and end-user identities	Top Administrator
FMT_MOF.1	Management of the group of roles that can interact with the functions in the TSF	Top Administrator
FMT_MTD.1	Management of the group of roles that can interact with the TSF data	Top Administrator
FMT_SMR.1	Management of the group of users that are part of a role.	Top Administrator
FPT_ITT.1	Management of the mechanism used to provide the protection of the data in transit between different parts of the TSF	Top Administrator
FPT_TST.1	Management of the conditions under which TSF self testing occurs, such as during intial start-up. Regular interval, or under specified conditions Management of the time interval if appropriate	Top Administrator

[Table 6-5] List of security functions management

6.4.2 ID and Password management

The authorized administrator forces the password to be changed at the first access to the security management interface. In case of the audit administrator, the password must be changed after the password is reset by the authorized administrator.The authorized administrator can change the password of the administrator or end-user through the security management interface.

When generating and changing the passwords of the end-user and the authorized administrator, the validity of the password value is verified according to the password policy. The TOE provides the following verification mechanism when generating the password.

- Min/Max length : 9 ~ 16 digits
- Allowable characters : English letter (26 letters: a~z), Number (10 letters: 0~9), Special character that can be input by using a keyboard (27 letters: .-/+=_:!@#$%^*()~{}[]|<>;&)
- At least 1 Letter to English letter, Number, Special character

6.4.3 SFR Mapping

SFR to be satisfied: FMT_MOF.1, FMT_MTD.1, FMT_PWD.1(Extended), FMT_SMF.1, FMT_SMR.1

6.5 Protection of the TSF

6.5.1 Internal TSF data transfer protection

In order to protect the TSF data between TOE components, the TOE protects the communication sector using the validated cryptographic module KSignCASE v2.5 in the policy transmission interval as follows.

1. KSignAccess Agent generates key and distributes it to KSignAccess Server
2. KSignAccess Sever and KSignAccess Agent authentication through digital signature generation and verification with RSA-PSS (2048bit) / SHA256 digital signature algorithm
3. RSAES (2048bit) Asymmetric Key (Public Key) Cryptographic operation performed on generated key
4. The distributed key is managed by the RSAES (2048bit) certificate of each TOE component.
5. Provides integrity with SEED-CBC data encryption algorithm and HMAC-SHA256 algorithm with distributed encryption key

6.5.2 Protection of stored TSF data

The TOE encrypts and stores the protected TSF data to protect against unauthorized disclosure and modification of the stored TSF data.

The protected TSF data list and applied cryptographic algorithm are as follows.

TSF Data	Algorithm and Data	Encoding
Administrator Password	SHA256(password)	Base64
End-user Password	SHA256(password)	Base64
Token	SEED-CBC(token)	Base64
Key Encryption Key	SEED-CBC(token)	Base64
Critical Security Parameter (Secret Key)	SEED-CBC(token)	Base64
Critical Security Parameter (Private Key)	SEED-CBC(token)	Base64
Intigrity Value	SHA256(data)	Base64

[Table 6-6] Protection of stored TSF data

6.5.3 TSF Self Tests

The TSF self-test demonstrates the correct operation of the TSF and provides the ability for the authorized administrator to verify that the integrity of the TSF data is not compromised through self-testing. The TOE performs self-tests periodically during normal operation (every 3 hours) at the start-up of the TOE to verify correct operation of all TSFs.

The TOE generates a hash value for the integrity check and process check objects for the self test at the specified inspection cycle and compares it with the hash value (reference value) stored at the initial operation. At this time, if the integrity violation is found, the TOE notifies the authorized administrator and generates the audit data through the security management interface. After the authorized administrator has successfully completed the identification and authentication, the TOE can update the hash value of the integrity object through the TOE security management interface. The TOE performs integrity check on all configuration files and modules such as the security policy file necessary for the TOE operation. The TOE audits and records the results of the self-test, the integrity check, and the actions of the authorized administrator.

6.5.4 Integrity Tests

The integrity check function determines whether the TSF data and TSF executables are tampered. The authorized administrator performs the integrity check on the KSignAccess Server. Compares the stored HMAC value with the HMAC value when the integrity check is performed and verifies whether the data is correct. Integrity verification uses the HMAC-SHA256 algorithm.

The conditions for the integrity check are as follows.

TOE Component	Integrity check occurrence condition
KSignAccess Server	During initial start-up periodically during normal operation (every 12 hours)

[Table 6-7] Integrity check occurrence condition

6.5.5 SFR Mapping

SFR to be satisfied: FPT_ITT.1, FPT_PST.1(Extended), FPT_TST.1

6.6 TOE Access (FTA)

The TOE performs user session restriction and session lock function of the security management interface for TOE access control.

When the TOE administrator accesses the TOE security management interface by entering the TOE_Server web address in the browser, the browser makes an HTTPS secure channel with the following protocol.

- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA25 (TLS V1.2)

6.6.1 Administrator Session Restrictions

The TOE limits the maximum number of concurrent sessions belonging to the same administrator to one in accordance with the administrator re-connection request rule (access permission IP) authorized by the same account or privilege after accessing the administrator for the top administrator, After connecting to the management server, if the other administator with the same account, the administrator's previous connection session terminate. In addition, the administrator access session is allowed according to the administrator access permission IP (two IP addresses by default) registered through the security management interface, and the unauthorized IP restricts the connection session.

In the case of the audit administrator, the access session is restricted according to the access permission IP rule and the audit data on the result of the session restriction of the security management interface is generated.

6.6.2 Locking the Session in the Security Management Interface

When the TOE is normally distributed and installed, the authorized administrator accesses the TOE security management interface through the web browser of the administrator's PC. The TOE permits access to the security management interface (HTTPS) only when the administrator attempting to access the connection that has been explicitly granted access has successfully completed the identification and authentication process.

The TOE terminates the session that interacts with the authorized administrator when the authorized administrator logs in to the security management interface (Web UI) of the TOE and then exceeds the inactivity elapse period. The inactivity elapsed period is set to a default value of 10 minutes and can not be changed. At the end of the session, the TOE disables all actions through the existing session and terminates the session. The TOE generates a new session only when the administrator re-authentication (administrator identification and authentication) has been successfully completed and the access to the security management interface is permitted when the authorized administrator who has disabled all actions re-attempts to use the security management. The TOE generates audit data on the result of these events, the result of performing the session lock of the security management interface.

6.6.3 SFR Mapping

SFR to be satisfied: FTA_MCS.2, FTA_SSL.5(Extended), FTA_TSE.1