Korean National

Protection Profile for

Database Encryption

V3.0

2023. 4. 27.

The certified Protection Profile is written in Korean. This document is a translation of the original from Korean into English.

Foreword

This Protection Profile has been developed with the support of National Security Research Institute (NSR) under the agreement between National Intelligence Service (NIS) and Ministry of Science and ICT (MSIT). The Protection Profile author converted Part 2, Common Security Requirements of 'Security Requirements for Government V3.0 for the Information Security Systems and Network Devices' and Security Requirements described in 'Database Encryption Product Testing Criteria(2022-03-15)' in conformity with the Common Criteria. The accurate interpretation of the requirements was made through the advice of the National Cyber Security Center of the National Intelligence Service. The Protection Profile includes application notes which give the additional interpretation and guidance for the evaluation and certification based on the Common Criteria, and the separated guidance supporting document (Korean only) for the Protection Profile is provided.

Revision History

Version	Date	Content
3.0	2023. 4. 27.	o Korean National Protection Profile for Database Encryption V3.0 First Issue

Table of Contents

1. PP introduction

1.1. PP reference

Title	Korean National Protection Profile for Database Encryption
Version	3.0
Evaluation Assurance Level	EAL1+(ATE_FUN.1)
Developer	National Security Research Institute
Evaluation Criteria	Common Criteria for Information Technology Security Evaluation
Common Criteria version	CC V3.1 r5
Certification Number	KECS-PP-1232-2023
Keywords	Database, Encryption

1.2. TOE overview

1.2.1. Database Encryption overview

Database encryption (hereinafter referred to as "TOE") performs the function of preventing the unauthorized disclosure of confidential information by encrypting the database (hereinafter referred to as "DB").

The encryption target of the TOE is the DB managed by the database management system (hereinafter referred to as "DBMS") in the operational environment of the organization, and the protection profile (hereinafter referred to as "PP") defines the user data as all data before/after encrypted and stored in the DB. Part or all of the user data can be the encryption target, depending on the organizational security policies that runs the TOE.

The DBMS that controls the DB in the operational environment of the organization is different from the DBMS that is directly used by the TOE to control the TSF data (security policy, audit data, etc.).

1.2.2. TOE type and scope

The TOE components are provided in the form of appliance or software and shall provide the encryption/decryption function for the user data by each column. The TOE can also provide a one-way encryption method in addition to the encryption/decryption method for each column. The TOE type defined in this PP can be grouped into the 'plug-in type' and 'API type', depending on the TOE operation type. The TOE can support both types. The TOE developed by the plug-in type can generally be composed of the agent and management server, whereas the TOE developed by the API type can be composed of the API module and management server.

The TOE developer can implement the management server with several TOE components by subdividing roles such as the encryption/decryption of the user data, security management function, and cryptographic key management function. For example, additional management tools developed for security management (like management console) can be included in the TOE component. In this case, the security target (hereinafter referred to as "ST") author shall identify all TOE components in the ST.

1.2.3. TOE usage and major security features

The TOE is used to encrypt the user data according to the policy set by the authorized administrator to prevent the unauthorized disclosure of the confidential information. In order that the authorized administrator can operate the TOE securely in the operational environment of the organization, the TOE provides various security features such as the security audit function that records and manages major auditable events; cryptographic support function such as cryptographic key management to encrypt the user and the TSF data, and cryptographic operation; user data protection function that encrypts the user data and protects the residual information; identification and authentication function such as verifying the identity of the authorized administrator, authentication failure handling, and mutual authentication among the TOE components; security management function for security functions, role definition, and configuration; TSF protection functions including protecting the TSF data transmitted among the TOE components, protecting the TSF data stored in the storage that is controlled by the TSF, and TSF self-test; and TOE access function to manage the access session of the authorized administrator.

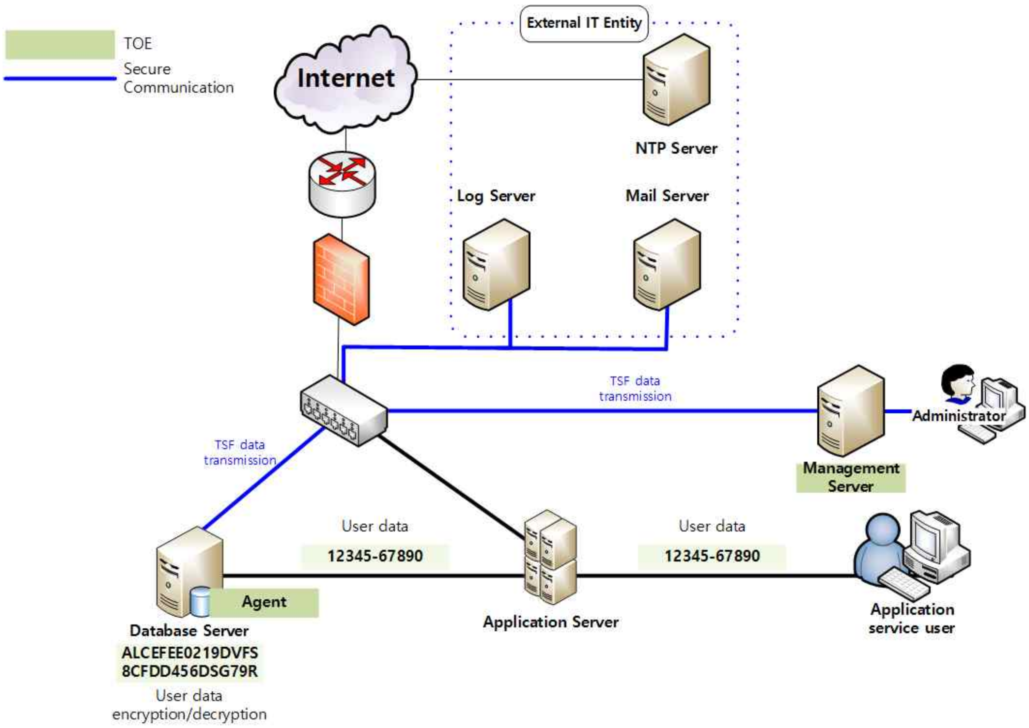
The DEK (Data Encryption Key) used to encrypt/decrypt the user data is protected by encryption with the KEK (Key Encryption Key). For the requirements regarding how to generate and use the DEK and KEK, refer to 5.1.2. Cryptographic Support (FCS).

1.2.4. Non-TOE and TOE operational environment

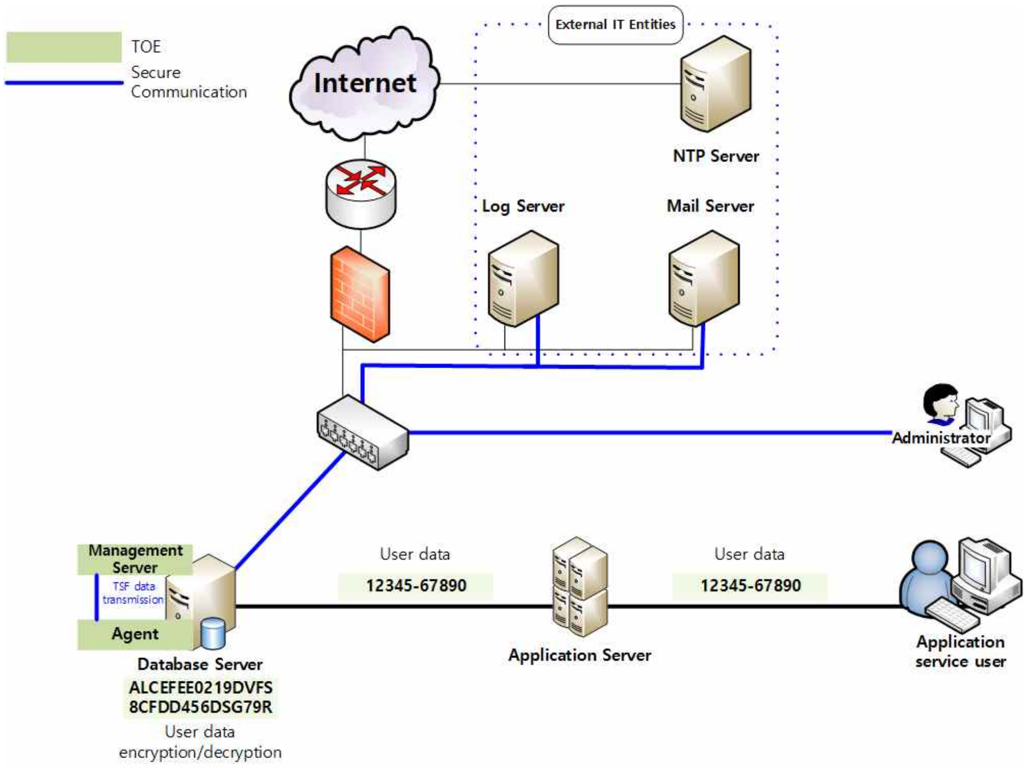
The TOE operational environment defined in this PP can be classified into two: plug-in type and API type.

Figure 1-1 and Figure 1-2 show the general operational environment of the plug-in type. The agent, which is installed in the protected database server of the DB, encrypts the user data of the application server before storing it in the DB according to the policy configured by the authorized administrator, and decrypts the encrypted user data sent from the database server to the application server.

The authorized administrator can encrypt/decrypt the user data through the management server according to the scope of the encryption that is required by the organizational security policy. In addition, the authorized administrator can perform security management through access to the management server. The management server can be installed in the database server along with the agent, or installed separately from the agent. The ST author shall clearly identify the operating location of the management server in the TOE operational environment, depending on the operation type of the TOE component.



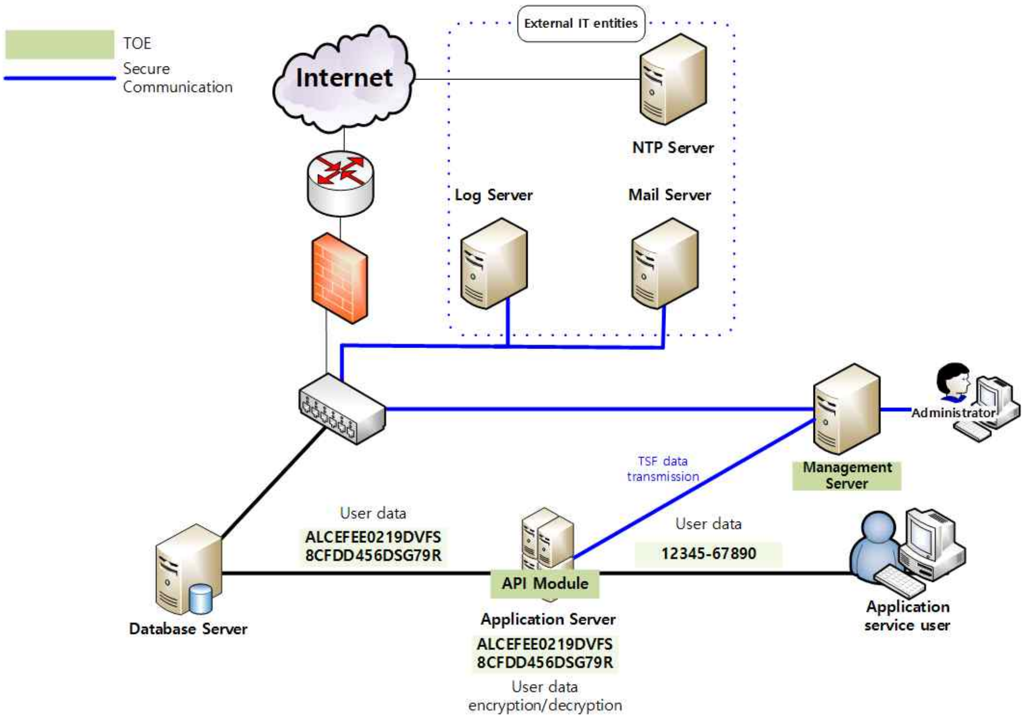
[Figure 1-1] Plug-in type operational environment (Agent, management server separate type)



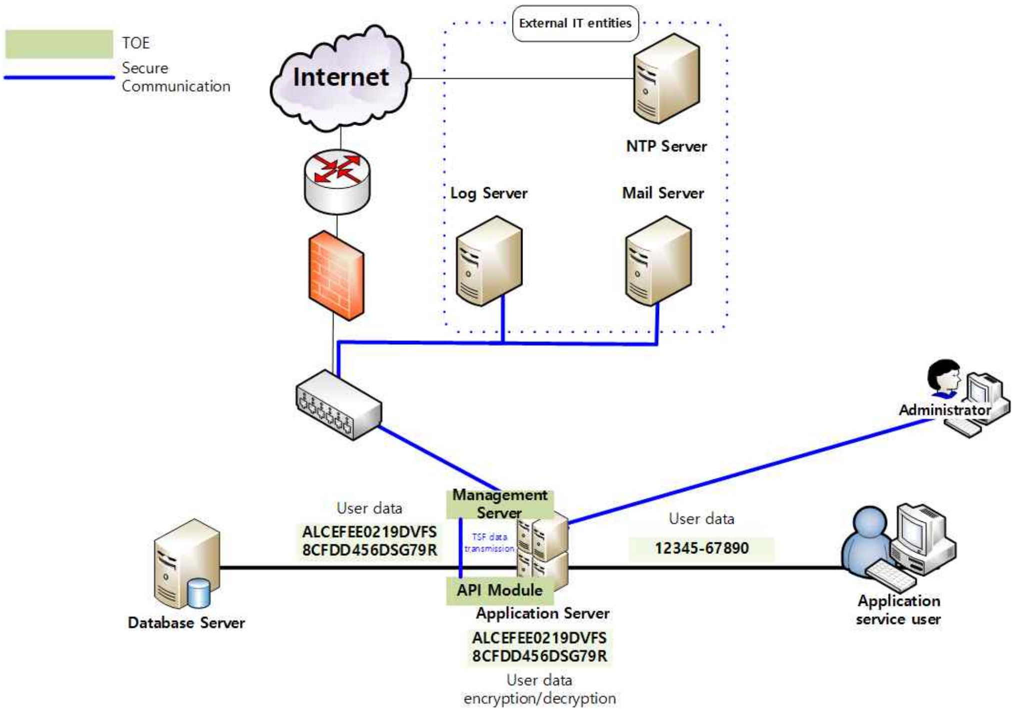
[Figure 1-2] Plug-in type operational environment (Agent, management server integrated type)

Figure 2-1 and Figure 2-2 show the general operational environment of the API type. The application, which is installed in the application server and provides application services, is developed using the API provided by API module in order to use the cryptographic function of the TOE. The API module is installed in the application server and performs encryption/decryption of the user data in accordance with the policies configured by authorized administrator. The user data entered by the application service user is encrypted by the API module, which is installed in the application server, and sent to the database server. The encrypted user data received from the database server is decrypted by the API module, which is installed in the application server, and sent to the application service user.

The authorized administrator can encrypt/decrypt the user data through the management server according to the scope of the encryption required by the organizational security policy. In addition, the authorized administrator can perform security management through access to the management server. The management server can be installed in the application server along with the agent, or installed separately from the API module. The ST author should clearly identify the operating location of the management server in the TOE operational environment, depending on the operation type of the TOE component.



[Figure 2-1] API-type operational environment (API module, management server separate type)



[Figure 2-2] API-type operational environment (API module, management server integrated type)

The communication among the TOE components shall be performed on the encrypted communication, and even though the TOE is operated as an integrated type, the TSF data shall be shared among the TOE components only through the encrypted communication. In addition, the encrypted communication shall be also applied even when communication between external IT entities and TOE components is required except for the NTP server.

The TOE user can be defined in various ways depending on the TOE operation and implementation. For the plug-in type, the authorized administrator who performs security management on the TOE using the management server is identified as the human user of the TOE. The DBMS that manages the DB in the database server and the application which is developed to provide application service in the application server can be the user of the TOE as the external IT entity, if the security function provided by the agent is used. For the API type, the authorized administrator who performs security management on the TOE using the management server is the human user of the TOE. The application developed to provide application service in the application server becomes the user of the TOE as the external IT entity when the security function provided by the API module is used.

The external IT entity needed to operate the TOE includes the NTP server to synchronize time, log server to store the audit data outside and manage the audit data, and email server to notify the authorized administrator in case of audit data loss. The ST author of the TOE complying with this PP shall identify all external IT entities that interact with the TOE in the ST.

The ST author shall include FAU_STG.1, a conditional mandatory security functional requirement, in the ST when the protected audit trail storage function is implemented in the TOE. If the function is not implemented in the TOE, the function must be provided in the operating environment (for example: using a DBMS, etc.), and accordingly, the security objectives for the operational environment must be added.

The ST author shall include FPT_STM.1, an optional security functional requirement, in the ST if the TOE implements a function that provides reliable time stamps. If the function is not implemented in the TOE, the function must be provided by the operating environment (for example: provided by the operating system, etc.), and accordingly, the security objectives for the operational environment must be added.

The ST author shall include the conditional mandatory security functional requirements defined in this PP if the following conditions are met.

- If the TOE provides additional identification and authentication mechanisms (e.g., certificate-based authentication method, OTP method, etc.) in addition to ID/PW-based identification and authentication, FIA_UAU.5 shall be included.

- When providing additional identification and authentication functions, the TOE can provide those functions by receiving the authentication results of external IT entities that interact with the TOE (e.g., 2FA support device that complies with the FIDO standards), and accordingly FPT_LEE.1(Extended) shall be included instead of FIA_UAU.5. In this case, the authentication information used by external IT entities to perform additional identification and authentication methods is safely managed by external IT entities, so the security objectives for the operating environment shall be added accordingly.

- In case of users(authorized administrators) directly access the management server through web browsers or terminal access programs, FTP_TRP.1 shall be included. Assuming that the web server is the TOE operating environment, and if a secure communication path is provided through communication between the user's web browser and web server, the ST author shall add the security objectives for the operational environment instead of including FTP_TRP.1. And if the user's web browser access the management server via the web server, such as when the web server and the management server are physically separated to perform communication, FTP_TRP.1 is included to provide a secure path between the management server and the user, and FTP_ITC.1 shall be included to provide a secure channel between the web server and the management server. FPT_ITT.1 shall be applied when transmitting TSF data between the TOE components which are physically separated.(eg, If communication between the TOE management console and the management server is directly implemented, FTT_ITT.1 shall be applied)

- When the TOE interacts with external IT entities(e.g., mail server, log server, etc.), FTP_ITC.1 shall be included.

Optional security functional requirements can be optionally implemented in the TOE. However, when the TOE additionally provides related capabilities, the ST author must include the corresponding SFRs. The ST author shall pay attention not to omit the security functional requirements for the security features provided by the TOE by referring to the application notes when applying each optional security functional requirement with regard to the applicability of the optional security functional requirements.

This PP has been developed considering various types of the TOE implementation. The ST author complying with this PP, shall describe any non-TOE hardware, software or firmware required by the TOE to operate.

1.3. Conventions

The notation, formatting and conventions used in this PP 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 marked 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 <u>*underlined and italicized.*</u>

Refinement

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

Security Target (ST) Author

This is used to represent the final decision of attributes being made by the ST author. The ST author's operation is denoted in braces, as in {decided by the ST author}. In addition, operations of SFR not completed in the Protection Profile must be completed by the ST author.

“Application notes” is provided to clarify the intent of requirements, provide the information for the optional items in implementation, and define "Pass/Fail" criteria for a requirement. The application notes is provided with corresponding requirements if necessary.

1.4. Terms and definitions

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

Agent Type1
Antivirus products, Software-Based Security USB products, Host Data Loss Prevention products, etc.

- The endpoint on which the agent is located is typically a PC with Windows® operating system accessible to employees within the organization, and if the agent is compromised, data present on the user's host can be compromised and leaked, requiring strict security requirements in terms of confidentiality, integrity, and availability.

Agent Type2
Network Access Control products, Patch Management Systems, etc.

- The endpoint on which the agent is located is typically a PC with Windows® operating system accessible to employees in the organization, and if the agent is compromised, it is unlikely that data present on the user's host will be corrupted or leaked, but it can cause problems in using the resources provided by the organization, requiring security requirements in terms of confidentiality, integrity.

Agent Type3
Database Access Control products, Access Control in Operating System(Server) products, Enterprise security management products, etc.

- Since the endpoint where the agent is located is generally a physically secure environment that can only be accessed by authorized employees of the organization, it corresponds to a product type with a relatively low threat occurrence.

Approved cryptographic algorithm
A cryptographic algorithm selected by Korea 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

Application Server

The application server defined in this PP refers to the server that installs and operates the application, which is developed to provide a certain application service by the organization that operates the TOE. The pertinent application reads the user data from the DB, which is located in the database server, by the request of the application service user, or sends the user data to be stored in the DB to the database server.

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

Authorized Administrator

Authorized user to securely operate and manage the TOE

Authentication Data

Information used to verify the claimed identity of a user

Authorized User

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

Automated recovery

Recovery without the user’s intervention

Can/could

The ‘can’ or ’could’ presented in Application notes indicates optional requirements applied to the TOE by ST author’s choice

Column

A set of data values of a particular simple type, one for each row of the table in a relational database

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).

Class

Set of CC families that share a common focus

Client Type
Vitual Private Network products, Wireless LAN Authentication Products, etcs.

- The client is an entity installed on the user's host and serves to request communication with the server on behalf of the user.

Database

A set of data that is compiled according to a certain structure in order to receive, save, and provide data in response to the demand of multiple users to support multiple application duties at the same time. The database related to encryption by column, which is required by this PP, refers to the relational database.

Database Server

The database server defined in this PP refer to the server in which the DBMS managing the protected DB is installed in the organization that operates the TOE

DBMS (Database Management System)

A software system composed to configure and apply the database. The DBMS related to encryption by column, which is required by this PP, refers to the database management system based on the relational database model.

Data Encryption Key (DEK)

Key that encrypts and decrypts the data

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

Encryption

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

Element

Indivisible statement of a security need

Endpoint
The point where the TOE components such as agents, clients, etc. are installed and operated without any further sub-interacted entities

External Entity

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

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

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

KCMVP, Korea Cryptographic Module Validation Program

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

Key Encryption Key (KEK)

Key that encrypts and decrypts another cryptographic key

Local access

Connection established through the console port between the administrator and the TOE

Management access

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

Manual recovery

Recovery through an update server, etc. by the user execution or user intervention

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

Organizational Security Policies
Set of security rules, procedures, or guidelines for an organization wherein the set is currently given by actual or virtual organizations, or is going to be given

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 keys

Random bit generator
A device or algorithm that outputs a binary string that is statistically independent and is not biased. The RBG used for cryptographic application generally generates 0 and 1 bit string, and the string 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

Security Function Policy (SFP)
A Set of rules that describes the specific security action performed by TSF (TOE security functionality) and describe them as SFR (security function requirement)

Secret Key

A cryptographic key which is used in an symmetric cryptographic algorithm and is uniquely associated with one or several entity, not to be disclosed

Security Target (ST)

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

Security attribute

The characteristics of the subject used to define the SFR, user (including the external IT product), object, information, session and/or resources. These values are used to perform the SFR

Security Token

Hardware device that implements key generation and electronic signature generation inside the device to save/store confidential information safely.

Selection

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

Shall/must

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

SSL (Secure Sockets Layer)

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

Symmetric cryptographic technique

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

Subject

Active entity in the TOE that performs operations on objects

Target of Evaluation (TOE)

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

Threat Agent

Entity that can adversely act on assets

TLS (Transport Layer Security)

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

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

TSF Data

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

User

Refer to "External entity"

User Data

Data for the user, that does not affect the operation of the TSF

1.5. PP organization

Chapter 1 introduces to the Protection Profile, providing Protection Profile references and the TOE overview.

Chapter 2 provides the conformance claims to the CC, PP and package; and describes the claim's conformance rationale and PP conformance statement.

Chapter 3 describes the security objectives for the operational environment.

Chapter 4 defines the extended components for the database encryption.

Chapter 5 describes the security functional and assurance requirements. If required, Application notes are provided to clarify the meaning of requirements and provide an explanation of detailed guidelines to the ST author for correct operations.

Reference describes the references for users who need more information about the background and related information than those described in this PP.

Abbreviated terms are listed to define frequently used terms in the PP.

2. Conformance claim

2.1. CC conformance claim

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)
Conformance claim	Part 2 Security functional components	Extended: FCS_RBG.1, FDP_UDE.1, FIA_IMA.1, FMT_PWD.1, FPT_LEE.1, FPT_PST.1, FPT_TUD.1
	Part 3 Security assurance components	*Conformant*
	Package	Augmented: EAL1 *augmented* (ATE_FUN.1)

2.2. PP conformance clam

This Protection Profile does not claim conformance to other PPs.

2.3. Package conformance claim

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

2.4. Conformance claim rationale

Since this Protection Profile does not claim conformance to other Protection Profiles, it is not necessary to describe the conformance claim rationale.

2.5. PP conformance statement

This Protection Profile requires “strict PP conformance” of any ST or PP, which claims conformance to this PP.

3. Security objectives

The followings are the security objectives handled by technical and procedural method supported from operational environment in order to provide the TOE security functionality accurately.

3.1. Security objectives for the operational environment

OE.PHYSICAL_CONTROL

The place where 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 guidances.

OE.SECURE_DEVELOPMENT

The developer who uses the TOE to interoperate with the 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.LOG_BACKUP

The authorized administrator of the TOE 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_RE-INFORCEMENT

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.

Application notes

o Depending on the implementation type of the TOE, the TOE components(agent, API module, management server) may not use the operating system independently, so care shall be taken that the operating system related settings of other external entities operating in the same operating system do not affect the secure operation of the TOE.

4. Extended components definition

4.1. Cryptographic support

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 Random bit generation

1

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 Random bit generation

Hierarchical to No other components.

Dependencies No dependencies.

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

4.2. Identification and authentication

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 TOE Internal mutual authentication

1

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 family is included in the PP/ST:

a) Minimal: 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*].

4.3. User data protection

4.3.1. User data encryption

Family Behaviour

This family provides requirements to ensure confidentiality of user data.

Component leveling

FDP_UDE User data encryption

1

FDP_UDE.1 User data encryption requires confidentiality of user data.

Management : FDP_UDE.1

The following actions could be considered for the management functions in FMT:
a) Management of user data encryption/decryption rules

Audit : FDP_UDE.1

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

a) Minimal : Success and failure of user data encryption/decryption

4.3.1.1. FDP_UDE.1 User data encryption

Hierarchical to	No other components.
Dependencies	FCS_COP.1 Cryptographic operation
FDP_UDE.1.1	TSF shall provide TOE users with the ability to encrypt/decrypt user data according to [assignment: *the list of encryption/decryption methods]* specified.

4.4. Security Management

4.4.1. ID and password

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 ID and password
1

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) Minimal: All changes of the password.

4.4.1.1. 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*].

Application notes

o If the TOE does not provide the capability for managing the ID and password combination rules by authorized roles, etc., 'None.' may be specified in assignment operations of FMT_PWD.1.1, FMT_PWD.1.2.

o The ID and password combination rules that can be set by authorized roles may include minimum and maximum length setting, mixing rule setting involving English upper case/lower case/number/special characters, etc.

4.5. Protection of the TSF

4.5.1. Linkable external entities

Family Behaviour

This family (FPT_LEE, Linkable External Entities) defines the requirement for the TSF to perform security functions with the support of external entities. In this family, external entities refer to software or hardware, but users are not counted as external entities.

Component leveling

FPT_LEE.1, linkable external entities, requires the TSF to provide the security functions by linking with external entities.

Management: FPT_LEE.1

There are no management activities foreseen.

Audit: FPT_LEE.1

It is recommended to record the following actions for audit if FAU_GEN Security audit data generation family is included in the PP/ST:

a) Minimal: Result of the execution of the security function provided by linking with external entities

4.5.1.1. FPT_LEE.1, Linkable External Entities

Hierarchical to	No other components.
Dependencies	No dependencies.

FPT_LEE.1.1 The TSF shall perform [assignment: *List of actions*] and provide [assignment: *List of functions*] by linking with external entities.

Application notes

- o In FPT_LEE.1.1, [assignment: List of actions] means the way the TSF is linked with external entities, such as API function call.
- o In FPT_LEE.1.1, [assignment: List of functions] shall specify the security functions (e.g. verification of secrets, protection of authentication feedback, etc.) provided by the TSF in linkage with external entities

4.5.2. 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.

4.5.2.1. 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*].

Application notes

o Containers controlled by the TSF mean storage in the TOE or external entities (DBMS, etc.)that interact with the TOE.

o Examples of TSF data to be protected as follows:

- User password, cryptographic key (pre-shared key, symmetric key, private key, etc), TOE configuration values (security policy, configuration parameters), audit data, etc.

o The TSF data can be encrypted and stored to be protected from the unauthorized disclosure or modification.

4.5.3. TSF Update

Family Behavior

This family defines TOE firmware/software update requirements.

Component leveling

FPT_TUD.1 TSF security patch update, requires trusted update of the TOE firmware/software including the capability to verify the validity on the update file before installing updates.

Management: FPT_TUD.1
The following actions could be considered for the management functions in FMT:
a) Management of update file verification mechanism

Audit: FPT_TUD.1
The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:
a) Minimal: Update file verification result (success, failure)

4.5.3.1. FPT_TUD.1 TSF Security Patch Update

Hierarchical to	No other components.
Dependencies	No dependencies.

FPT_TUD.1.1 The TSF shall provide the capability to view the TOE versions to [assignment: *the authorized identified roles*].

FPT_TUD.1.2 The TSF shall verify validity of the update files using [selection: *hash value comparison, digital signature verification*] before installing updates.

Application notes

o The TSF shall provide the capability to check the current version of TOE that most recently installed and executed by authorized roles.

o The latest updates and security patches are essential to remove security vulnerabilities. The validity verification on the update files is required since the installation of update files without any verification can result in system malfunction, or service failures, etc.

5. Security requirements

The security requirements specify security functional requirements and assurance requirements that must be satisfied by the TOE that claims conformance to this PP.

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

In addition, the security functional requirements are classified into mandatory SFRs and conditional mandatory SFRs, optional SFRs, as follows.

- Mandatory SFRs: are required to be mandatorily implemented in the Database Encryption
- Conditional mandatory SFRs: are required to be mandatorily implemented, if the stated conditions are met.
- Optional SFRs: are not required to be mandatorily implemented in database encryption. However, when the TOE additionally provides related capabilities, the ST author must include the corresponding SFRs.

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

Security functional class	**Security functional component**		**Remarks**
FAU	FAU_ARP.1	Security alarms	Mandatory SFR
	FAU_GEN.1	Audit data generation	Mandatory SFR
	FAU_SAA.1	Potential violation analysis	Mandatory SFR
	FAU_SAR.1	Audit review	Mandatory SFR
	FAU_SAR.3	Selectable audit review	Mandatory SFR
	FAU_STG.1	Protected audit trail storage	Conditional mandatory SFR
	FAU_STG.3	Action in case of possible audit data loss	Conditional mandatory SFR
	FAU_STG.4	Prevention of audit data loss	Conditional mandatory SFR
FCS	FCS_CKM.1(1)	Cryptographic key generation (User data encryption)	Mandatory SFR
	FCS_CKM.1(2)	Cryptographic key generation (TSF data encryption)	Mandatory SFR
	FCS_CKM.2	Cryptographic key distribution	Optional SFR
	FCS_CKM.4	Cryptographic key destruction	Mandatory SFR

Security functional class	Security functional component		Remarks
	FCS_COP.1(1)	Cryptographic operation (User data encryption)	Mandatory SFR
	FCS_COP.1(2)	Cryptographic operation (TSF data encryption)	Mandatory SFR
	FCS_RBG.1(Extended)	Random bit generation	Mandatory SFR
FDP	FDP_UDE.1(Extended)	User data encryption	Mandatory SFR
	FDP_RIP.1	Subset residual information protection	Mandatory SFR
FIA	FIA_AFL.1	Authentication failure handling	Mandatory SFR
	FIA_IMA.1(Extended)	TOE Internal mutual authentication	Mandatory SFR
	FIA_SOS.1	Verification of secrets	Mandatory SFR
	FIA_UAU.1	Timing of authentication	Mandatory SFR
	FIA_UAU.4	Single-use authentication mechanisms	Mandatory SFR
	FIA_UAU.5	Multiple authentication mechanisms	Conditional mandatory SFR
	FIA_UAU.7	Protected authentication feedback	Mandatory SFR
	FIA_UID.1	Timing of identification	Mandatory SFR
FMT	FMT_MOF.1	Management of security functions behaviour	Mandatory SFR
	FMT_MTD.1	Management of TSF data	Mandatory SFR
	FMT_PWD.1(Extended)	Management of ID and password	Mandatory SFR
	FMT_SMF.1	Specification of management functions	Mandatory SFR
	FMT_SMR.1	Security roles	Mandatory SFR
FPT	FPT_ITT.1	Basic internal TSF data transfer protection	Mandatory SFR
	FPT_LEE.1(Extended)	Linkable external entities – authentication	Conditional mandatory SFR
	FPT_PST.1(Extended)	Basic protection of stored TSF data	Mandatory SFR
	FPT_RCV.1	Manual recovery	Conditional mandatory SFR

Security functional class	Security functional component		Remarks
	FPT_RCV.2	Automated recovery	Conditional mandatory SFR
	FPT_STM.1	Reliable time stamps	Optional SFR
	FPT_TST.1	TSF testing	Mandatory SFR
	FPT_TUD.1(Extended)	TSF security patch update	Conditional mandatory SFR
FTA	FTA_MCS.2	Per user attribute limitation on multiple concurrent sessions	Mandatory SFR
	FTA_SSL.1	TSF-initiated session locking	Conditional mandatory SFR
	FTA_SSL.3	TSF-initiated termination	Conditional mandatory SFR
	FTA_TSE.1(1)	TOE session establishment	Mandatory SFR
	FTA_TSE.1(2)	TOE session establishment	Conditional mandatory SFR
FTP	FTP_ITC.1	Inter-TSF trusted channel	Conditional mandatory SFR
	FTP_TRP.1	Trusted path	Conditional mandatory SFR

[Table 1] Security functional requirements

5.1. Security functional requirements (Mandatory SFRs)

The database encryption that claims conformance to this PP must meet the following 'Mandatory SFRs'.

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
FCS	FCS_CKM.1(1)	Cryptographic key generation (User data encryption)
	FCS_CKM.1(2)	Cryptographic key generation (TSF data encryption)
	FCS_CKM.4	Cryptographic key destruction
	FCS_COP.1(1)	Cryptographic operation (User data encryption)
	FCS_COP.1(2)	Cryptographic operation (TSF data encryption)
	FCS_RBG.1(Extended)	Random bit generation
FDP	FDP_UDE.1(Extended)	User data encryption
	FDP_RIP.1	Subset residual information protection
FIA	FIA_AFL.1	Authentication failure handling
	FIA_IMA.1(Extended)	TOE Internal mutual authentication
	FIA_SOS.1	Verification of secrets
	FIA_UAU.1	Timing of authentication
	FIA_UAU.4	Single-use authentication mechanisms
	FIA_UAU.7	Protected authentication feedback
	FIA_UID.1	Timing of identification

Security functional class	Security functional component	
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
	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
FTA	FTA_MCS.2	Per user attribute limitation on multiple concurrent sessions
	FTA_TSE.1(1)	TOE session establishment

[Table 2] Mandatory 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 [assignment: *list of actions*] upon detection of a potential security violation.

Application notes

o If the TOE self-test result is a failure, response functions shall be performed.

- Examples of response functions to be performed when the self-test result is a failure are as follows:
  - *Termination of the program, warning message screen display, process restart, etc.*

o If the TOE integrity verification result is a failure, response functions shall be performed.

- Examples of response functions to be performed
- when the integrity verification result is a failure are as follows:
  - *Termination of the program, warning message screen display, etc.*

○ The TOE agents or clients shall verify the integrity periodically or upon the authorized administrator's request and provide the administrator with a result notification function.

- △ In case of abnormality in the integrity verification result, △ integrity verification result by the administrator shall be notified to the administrator.

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) [assignment: *other specifically defined auditable events*]

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 PP/ST, [assignment: *other* *audit relevant information*].

Application notes

1. Generate audit records related to Database Encryption

Security Functional Components	Audit events	Additional audit information
FDP_UDE.1	Success and failure of user data encryption/decryption	

2. Generation of other audit records

o The TOE shall generate audit records for major audit events.

- [Table 3] below shows the audit events for which audit records must be generated.

Sub-category	Audit events	Additional audit information
Identification and authentication	User login and logout	
	User registration, change and deletion	
	The reaching of the threshold for the unsuccessful user authentication attempts and the actions taken	
	All changes of the password	
Security management	Registration, deletion and change IP address of the management terminals.	
	Execution of security management function and all changes and deletions of security attribute values. ** However, among the security management functions, 'Audit record inquiry' and 'TOE version information inquiry' functions are excluded.	Changed security attribute data
	Default account(ID)/Password change	
	Management terminal access IP blocking	
Trusted session management	User's session locking or termination	
	Response actions when duplicate login attempts of the same account are detected	
	Denial of new sessions based on the limit on the number of concurrent sessions	
Cryptographic key generation	Cryptographic key generation failure	
Cryptographic operation	Cryptographic operation failure (including cryptographic operation type)	
Audit record	Start-up and shutdown of the TOE audit functions in the	

	form of H/W appliance	

[Table 3] Major mandatory audit events to be recorded

- [Table 4] below shows the audit events for which audit records must be generated when providing a function.

Sub-category	Audit events	Additional audit information
Self-protection	Execution of self-test	security function with failed self-test
	Execution of integrity verification of the TOE itself	Components with failed integrity verification
Update protection	Updated files validity verification by the administrator	
	Execution of update files validity verification	
Audit records	Start-up and shutdown of the TOE audit function in the form of software	
	Response actions when audit record fails to be stored	
Security management	Changes in agent registration status	

[Table 4] Audit events that must be recorded when providing a function

o If the TOE detects an attempt to reuse authentication information that is prohibited for reuse, authentication shall fail and an audit record of the authentication failure event shall be generated.

o Audit records shall be generated for self-test results.

o Integrity verification contents and results shall be confirmed through *screen display, audit records.*

o Audit records shall be generated for integrity verification results.

o Update file validation results(success•failure) shall be recorded in audit records.

o Audit records shall be generated for the update installation results and the reason for failure.

o Audit records shall be generated when the session locking or termination function is activated.

o Audit records shall be generated when blocking duplicate access.

o Audit records shall not contain more information than necessary.

- Items that shall be included at least in audit records are as follows.

- The date and time of the event, the type of event, the identity of the subject that caused the event (e.g., *account, process, IP, etc.*), and the outcome of the event (success• failure)

- Information such as authentication information (e.g., *password, etc.*) and encryption key shall not be stored in the audit records.

o Sensitive data (e.g., *password, resident registration number, etc.*) shall not be recorded, or shall be generated by processing with masking if record is inevitable.

o Each component of the TOE shall generate audit records using trusted time information.

- Trusted time information should use the time information provided by the NTP server or the operating system.

o If the WAS(*Tomcat, Jesus, etc.*) is included in the TOE package, the TOE shall be implemented so that important information is not included in the WAS log.

- It can be implemented so that the log may be left only in the TOE's audit record storage without leaving the WAS log.

- Important information such as passwords and encryption keys shall not be left in plain text in the WAS log.

o Clients and agents shall generate audit records listed in the following [Table 5].

Security function	Audit event	Additional audit information
Self-protection	Execution of integrity verification and its results	
Security management	When providing security management functions, execution of security management functions and any changes of security attribute values.	Changed security attribute data
Audit record	Agent start-up	
	When general users can request the audit record to be transmitted to the server through security management, execution of transmission of the audit record.	
Safe update and file distribution	(When providing online update function) Execution of digital signature verification of files received from the server and external update server and its results	Files that has failed digital signature verification

[Table 5] Major audit events to be generated

- The applicant shall describe the audit list for major events provided by agents or clients in the guidance documents.

- The integrity verification results shall be generated as audit records.

o The audit records of clients and agents shall include key information for each event.

- The date and time, event type, identity of the subject who caused the event, and the outcome of the event shall be included.

o If there is a server, the function to transmit the major audit records generated by agents or clients to the server shall be provided.

- [Table 5] The server transmission function of the audit records described in the major audit events to be generated shall be implemented.
- After disconnection from the server, the audit records loaded after the disconnection shall be all transmitted to the server when it is recovered.
- Protection of audit records transmitted to the server shall satisfy the requirements of FPT_ITT.1.

o The update file digital signature verification result (success or failure) shall be audited and recorded.

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 [assignment: *subset of defined auditable events*] known to indicate a potential security violation

b) [assignment: *any other rules*]

Application notes

o If the result of the TOE’s self-test is failure, the response function shall be performed.

o The TOE shall perform the response function if the integrity verification fails.

o The TOE agents or clients shall verify the integrity periodically or upon the authorized administrator's request and provide the administrator with a result notification function.

- △In case of abnormality in the integrity verification results △Integrity verification results by the administrator shall be notified to the administrator.

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.

Application notes

- o The TOE shall provide a function for the authorized administrator to inquire the audit record.
  - The audit record shall be inquired only through the security function provided by the TOE.
  - The TOE shall provide audit records for the authorized administrator to properly 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 capability to apply [assignment: *methods of selection and/or ordering*] of audit data based on [assignment: *criteria with logical relations*].

Application notes

- o The TOE shall provide a function for the administrator to select a logical condition when inquiring audit records, and to search or sort the records according to various conditions.

5.1.2. Cryptographic support (FCS)

5.1.2.1. FCS_CKM.1(1) Cryptographic key generation (User data encryption)

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 **data encryption keys(DEK)** in accordance with a specified cryptographic key generation algorithm [assignment: *cryptographic key generation algorithm*] and specified cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*].

Application notes

- o The TOE shall use the random bit generator of the validated cryptographic module when generating cryptographic key required for database encryption.
  - It shall be applied when generating data encryption key(DEK).

- The entropy of the random bit generator SEED value shall be 2^{112} or higher.
- When generating a key encryption key(KEK), it is also allowed to derive a cryptographic key from the password.
  - A key encryption key(KEK) can be derived from the password entered by the user.
  - When deriving a key encryption key(KEK) from the password, a secure method shall be applied suggested in TTAK.KO-12.0334-Part1~Part4.
- Cryptographic keys generated by using the password is limited to the generation of a key encryption key(KEK).

5.1.2.2. FCS_CKM.1(2) Cryptographic key generation (TSF data encryption)

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 [assignment: *cryptographic key generation algorithm*] and specified cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: list of standards].

Application notes

o The TOE shall generate cryptographic keys in a secure method.

- Examples of secure cryptographic key generation methods are as follows:
  - *Password-based key derivation(PKCS#5 v2.1(RFC 8018), NIST SP 800-132, etc.)*
  - *Key derivation with pre-shared keys(TTAK.KO-12.0272)*
  - *Key generation using random bit generator(CTR_DRBG, HASH_DRBG, HMAC_DRBG, etc.)*
- The random bit generator shall be implemented in compliance with domestic and foreign standards.
- It is possible to generate asymmetric key pairs (public keys/private keys) or symmetric keys using random bits generated by the random bit generator.
- The password-based key derivation function shall only be used to generate a Key Encryption Key(KEK).
  - The initial key encryption key shall be generated differently for each TOE.
  - Initial data required to generate a key encryption key can be directly entered or injected from stored values in storage media such as smart cards, security USBs, security tokens(HSM: Hardware Security Module).
  - It is recommended to use products that have obtained security function test report or

domestic/foreign CC certificates for the storage media.

- For details, refer to the Encryption Key Generation of the 'Encryption Key Management Guide' (Ministry of Science and ICT, 2014).
- If a password is used as the initial data for generating a key encryption key(KEK), the value entered at the time of the initial installation of the product can be stored and used, and the stored data shall be protected from unauthorized exposure attempts.

5.1.2.3. 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 destruct cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: *cryptographic key destruction method*] that meets the following: [assignment: *list of standards*].

Application notes

1. Destruction of DB encryption-related cryptographic keys and critical security parameters

o The TOE shall delete the used key encryption key(KEK).

o When terminating execution, all cryptographic keys and critical security parameters loaded in the memory shall be deleted.

- When destroying cryptographic keys and critical security parameters, a method of overwriting at least 3 times with 0 or 1 can be used.

2. Destruction of other cryptographic keys

o The TOE shall securely destroy the cryptographic keys generated or used in the TOE.

- △When terminating execution of the TOE, △When calling cryptographic key deletion function, △When terminating cryptographic communication, etc., all cryptographic keys and information related to cryptographic key that have expired shall be destroyed.
- When destroying cryptographic keys, a method of overwriting at least 3 times with values of 0 or 1 can be used.
- For details, refer to the cryptographic key destruction method of the 'Encryption Key Management Guide' (Ministry of Science and ICT, 2014).

5.1.2.4. FCS_COP.1(1) Cryptographic operation (User data encryption)

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 [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*].

Application notes

o The TOE shall perform encryption when authorized users store important information in the database.

- The database encryption/decryption function shall use the approved cryptographic algorithm of the validated cryptographic module of which security and implementation suitability have been confirmed through the Korea Cryptographic Module Validation Process(KCMVP), and the validated cryptographic module must run in the approved mode of operation.

o The TOE shall use the cryptographic algorithm of the validated cryptographic module to perform cryptographic operations on the database.

- The TOE cannot use the ECB mode if the plain text size is larger than the encryption block when using block cipher algorithm, nor use fixed IV when using CFB or OFB mode.
- Refer to the list of product types subject to installation of validated cryptographic modules and validated cryptographic modules posted on the website of the National Intelligence Service.

5.1.2.5. FCS_COP.1(2) Cryptographic operation (TSF data encryption)

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 [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*].

Application notes

o The TOE shall use the recommended cryptographic algorithm when transmitting and storing important information.

o The recommended cryptographic algorithm is a standard algorithm with a security strength of 112 bits or more. Refer to the [Attachment] to the auxiliary document. Examples are as follows:

- *Hash Algorithm: SHA-224 or higher*
- *Symmetric key Algorithm: Key length 128 bits or higher*
- *Public key Algorithm: RSA 2048 or higher, DSA(2018, 224) or higher*
- *Digital signature Algorithm: RSA-PSS 2048 or higher, KCDSA(2048, 224) or higher, ECDSA/EC-KCDSA (B-233, B-283, K-223, K-283, P-224, P-256)*

o However, the use of TDES(including 2 keys and 3 keys) is not permitted.

o When using block cipher, ECB mode shall not be used if the plain text size is larger than the encryption block size.

o When using block cipher, fixed IV shall not be used in CFB or OFB mode.

o Domestic/foreign standard cryptographic algorithms shall be used, and the use of the national cryptographic algorithm is recommended.

o For details of cryptographic algorithm with a security strength of 112 bits or higher, refer to 'Guide to Cryptographic Algorithm and Key Length' (Ministry of Science and ICT, 2018), 'Software Cryptographic Module Validation Standard' and 'NIST SP 800-131Ar2'.

5.1.2.6. FCS_RBG.1 Random bit generation (Extended)

Hierarchical to	No other components.
Dependencies	No dependencies.

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

Application notes

1. Random bit generation related to DB encryption

o The TOE shall use the random bit generator of the validated cryptographic module when generating cryptographic key required for database encryption.

- The entropy of the random bit generator SEED value shall be 2^{112} or higher.

2. Other random bit generation related to TSF data

o Examples of secure cryptographic key generation methods are as follows:

- *Password-based key derivation(PKCS#5 v2.1(RFC 8018), NIST SP 800-132, etc.)*
- *Key derivation with pre-shared keys(TTAK.KO-12.0272)*

- *Key generation using random bit generator(CTR_DRBG, HASH DRBG, HMAC_DRBG, etc.)*

o The random bit generator shall be implemented in compliance with domestic and foreign standards.

o It is possible to generate asymmetric key pairs (public keys/private keys) or symmetric keys using random bits generated by the random bit generator.

o User password used by the TOE for user identification and authentication shall be stored using a one-way encryption(Hash) to prevent decryption.

- When performing a one-way encryption, it is necessary to add and apply a randomly generated value called salt to the password.
- The salt value does not need to be confidential. It shall be generated using random bit generator and the size must be at least 48 bits.
- The iteration count shall be applied as large as possible. (at least 1000 times)

5.1.3. User data protection (FDP)

5.1.3.1. FDP_UDE.1 User data encryption (Extended)

Hierarchical to	No other components.
Dependencies	FCS_COP.1 Cryptographic operation

FDP_UDE.1.1 The TSF shall provide a function that can encrypt/decrypt the user data to the TOE user according to the specified [encryption/decryption method by column, [assignment: *List of other encryption/decryption methods*]].

Application notes

o The TOE shall perform encryption when authorized users store important information in the database.

- When authorized users store important information in the database, it is mandatory for the TOE to provide the encryption function for each column.
  - When requested by authorized users, the TOE shall provide the function to decrypt important information encrypted and stored in the database.
- The TOE may provide a one-way encryption method in addition to the encryption method for each database column.
- The database encryption/decryption function shall use the approved cryptographic algorithm of the validated cryptographic module of which security and implementation suitability have been confirmed through the Korea Cryptographic Module Validation Program(KCMVP), and the validated cryptographic module must run in the approved mode of operation.
- The encryption algorithm used, cryptographic key security, and cryptographic key storage method shall meet the requirements related to DB encryption of FCS Class and FPT_PST.1.

5.1.3.2. FDP_RIP.1 Subset residual information protection

Hierarchical to	No other components.
Dependencies	No dependencies.

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

Application notes

o In FDP_RIP.1.1, 'not available' means unrecoverable deletion.

o When user data encryption/decryption are performed at the TOE operational environment (Application Server, or Database Server) by further development (or modification) of the TOE purchaser, the TOE operational environment shall be developed in accordance with the requirements provided by the TOE and this note shall be described in the TOE guidance documents.

5.1.4. Identification and authentication (FIA)

5.1.4.1. FIA_AFL.1 Authentication failure handling

Hierarchical to	No other components.
Dependencies	FIA_UAU.1 Timing of authentication

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

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been [selection: *met, surpassed*], the TSF shall [assignment: *list of actions*].

Application notes

o If user authentication fails consecutively as many times as the set number in the TOE, the identification and authentication functions shall be deactivated.

- Examples of how to activate after deactivating the identification and authentication functions are as follows:
  - *Activation in a specified period of time after account lock-out*
  - *Provision of other identification and authentication means for activation after account lock-out*

- Additional identification and authentication means specified in FIA_UAU.1 may be provided. In case of authentication failure with additional identification and authentication means, it shall be included in the number of user authentication failures.
- The number of consecutive authentication failures in which identification and authentication are deactivated shall be fixed or settable at a value of 5 or less.
- When implementing to deactivate the authentication function for a certain period of time, the time required for re-activation shall be fixed or settable at a value of 5 minutes or more.

o If administrator authentication fails consecutively as many times as the set number, the TOE shall notify the administrator through means that can be immediately checked.

- Notification shall be made through at least one of *alarm, text messaging, e-mail, etc.*

5.1.4.2. FIA_IMA.1 TOE Internal mutual authentication (Extended)

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*].

Application notes

o This SFR must be applied among the TOE components that are physically separated.

o If the TOE components include the server and the agent that receives the security policy from it, the agent shall perform identification and authentication for the server.

- Agents shall perform identification and authentication to confirm the legitimacy of the server.
- One of the server IP address and domain name must be included in the server identification information, and additional identification information can be used.
- The authentication method for the server includes a *certificate-based authentication method, etc*.
- When using a certificate, verification of the validity of the certificate(within 1 year of validity) shall be performed.

5.1.4.3. 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 [assignment: *a defined quality metric*].

Application notes

o If ID/password is the only means of user identification and authentication, the TOE shall meet the security criteria of [Table 6] Password Security Criteria Type(1) when registering and changing passwords.

Description	Contents	Remarks
Compliance	Secure the length of more than 9 digits	Mandatory
	Contains at least one number, uppercase letter(english), lowercase letter(english), and special character	Mandatory
Prohibition	Do not set the same password as the user account (ID)	Mandatory
	Prohibition of consecutive repeated input of the same letter/number	Mandatory
	Prohibit sequential input of consecutive letters or numbers on the keyboard	Mandatory
	Prohibition of reuse of the password used immediately before	Implement either one of the two
	Prohibition of reuse of the password used within the past 3 months	

[Table 6] Password Security Criteria Type(1)

o If ID/password input and additional identification and authentication functions are performed concurrently, the TOE shall meet the security criteria of [Table 7] Password Security Criteria Type(2) when registering and changing passwords.

Description	Contents	Remarks
Compliance	Secure the length of more than 6 digits.	Mandatory
	Contains at least one number, uppercase letter(english), lowercase letter(english), and special character.	Optional
Prohibition	Do not set the same password as the user account (ID)	Mandatory
	Prohibition of consecutive repeated input of the same letter/number	Optional
	Prohibit sequential input of consecutive letters or numbers on the keyboard	Optional
	Prohibition of reuse of the password used immediately before	Optional
	Prohibition of reuse of the password used within the past 3 months	Optional

[Table 7] Password Security Criteria Type(2)

5.1.4.4. FIA_UAU.1 Timing of authentication

Hierarchical to No other components.

Dependencies FIA_UID.1 Timing of identification

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

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user, except for the actions specified in FIA_UAU.1.1.

Application notes

o The TOE shall provide user account/password-based identification and authentication functions to verify the identity of the user.

- Identification and authentication must be performed to confirm that the user is a legitimate user of the TOE.
- If it is required to identify and authenticate users who exist in the agents or clients constituting the TOE, the identification value shall be a unique value that is not registered in duplicate.
  - When authenticating the user, the additional attributes of the registered agents or clients shall also be authenticated.
  - Additional attributes: IP address is mandatory, and at least one of *the MAC address, Serial Number, and information that can uniquely identify the agent itself* shall be additionally used.

o In case of the TOE supports additional identification and authentication methods, for user identification and authentication, the TOE must provide additional identification and authentication functions on its own or by interacting with external IT entities in parallel with user account and password-based identification and authentication.

- In order to provide additional identification and authentication functions, △2FA support device complying with FIDO standards, △certificates, △one-time password generator(OTP), etc. can be used.
  - If it is supported in the TOE operating environment, '2FA support device complying with FIDO standards' is recommended.
- If additional identification and authentication functions are provided by the TOE, the functions can be provided by receiving the authentication results from the inside of the TOE or from interaction with the external IT entities.
  - If the TOE provides a certificate utilization method, certificate validation shall be performed.
  - The authentication information used by external IT entities to perform additional identification and authentication methods shall be securely managed by the external IT entities. If the TOE stores authentication information use to perform additional identification and authentication methods, the requirements of FPT_PST.1 shall be applied.

- o If the TOE authenticates external IT entities, the TOE shall authenticate the interacted external IT entities.

5.1.4.5. FIA_UAU.4 Single-use authentication mechanisms

Hierarchical to	No other components.
Dependencies	No dependencies.

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

Application notes

o The TOE shall prevent reuse of user's authentication information(*using timestamp, encrypting session ID, etc.*)

- It is mandatory to apply to authentication information to be used for user account/password-based identification and authentication specified in FIA_UAU.1.
- If the TOE receives authentication information from the user to provide additional identification and authentication methods specified in FIA_UAU.1, it is mandatory to apply to the corresponding authentication information.
- It can be prevented by encrypting the session ID or guaranteeing the uniqueness of the session ID*(including timestamp and random bit values, setting session expiration time, etc.)*
- If the TOE detects an attempt to reuse authentication information that is prohibited from being reused, authentication shall fail and an audit record shall be generated for the authentication failure event.

5.1.4.6. FIA_UAU.7 Protected authentication feedback

Hierarchical to	No other components.
Dependencies	FIA_UAU.1 Timing of authentication

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

Application notes

o The TOE shall not display the contents when displaying the information used for authentication on the output device.

- It shall be applied when the authentication information specified in FIA_UAU.1 is displayed on the output device.
- The information used for authentication shall be output in the form of *no-display of input contents, display of "*" instead of input characters, etcs*.
- When users log in, the authentication information shall not be exposed with plain text in the memory area.

o In case of identification and authentication failures, the TOE shall not provide the feedback for the cause of failure (*e.g. non-existent account(ID), password error, etcs.)*.

5.1.4.7. FIA_UID.1 Timing of identification

Hierarchical to	No other components.
Dependencies	No dependencies.

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

FIA_UID.1.2 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user, except for the actions specified in FIA_UAU.1.1.

Application notes

o The TOE shall provide user account/password-based identification and authentication functions to verify the identity of the user.

- Identification and authentication must be performed to confirm that the user is a legitimate user of the TOE.

o When supporting additional identification and authentication methods, the TOE shall provide additional identification and authentication functions on its own or in conjunction with external IT entities, in parallel with user account/password-based identification and authentication.

o If the TOE authenticates external IT entities, the TOE shall authenticate the interacted external IT entities.

5.1.5. Security management (FMT)

5.1.5.1. FMT_MOF.1 Management of security functions behaviour

Hierarchical to	No other components.
Dependencies	FMT_SMF.1 Specification of Management Functions FMT_SMR.1 Security roles

FMT_MOF.1.1 The TSF shall restrict the ability to ***conduct management actions of*** the functions [assignment: *list of functions*] to [the authorized administrator].

Application notes

o The TOE shall provide the authorized administrator with the security management functions to set and manage security functions, security policies, important data, etc.

- The security management functions include the followings:
  - A function to add, delete or change conditions or rules that can determine the operation of the security function.
  - A function to add, remove or change the actions to be performed by the TOE in accordance with the conditions or rules.
  - A function to select or change TOE settings
- The security management functions to be implemented by the TOE are shown in [Table 8] below.

Sub-category	Security management	Remarks
Identification and authentication	User registration, deletion and change, grant privileges	Not applicable, if the user registered in the TOE is the only one.
	Setting user's password combination/length policy	Mandatory when providing the function
	Setting the allowed number of user's authentication failures	Mandatory when providing the function
	Setting the response methods to user's authentication failures	Mandatory when providing the function
	Setting the time from deactivation of user's authentication function to re-activation	Mandatory when providing the function
	Setting the authentication information of external IT entities that is authenticated by the TOE.	Mandatory when providing the function
Security management	IP registration, deletion and change of the management terminals	
	Backup of important data, configuration information, audit records, etc.	Mandatory when providing the function
	Recovery of of important data, configuration information, audit records, etc.	Mandatory when providing the function
Security management	Enabling and disabling management access service	Mandatory when providing the function

	Agent inquiry - status, version, and applied security policy	Mandatory when including agents
	Agent security policy management – policy settings, policy transmission	Mandatory when including agents
	Setting the authentication information for access to external IT entities	Mandatory when providing the function
Self-protection	Performing self-test for TOE's security function by administrator's request	Mandatory when providing the function
	Setting response actions when self-test fails	Mandatory when providing the function
	Performing an integrity verification of the TOE setting values and the TOE itself by the administrator's request	
	Setting response actions when integrity verification fails	Mandatory when providing the function
Update protection	Manual validation of update files by administrator	Mandatory when providing the function
	Manual recovery of failed installation of update files by administrator	Mandatory when providing the function
	Inquiry of TOE version information	
Safe session management	User session locking time, user session timeout time setting	Mandatory when providing the function
	(In case session locking) Administrator or individual user authentication when unlocking sessions	
	Setting the number of concurrent user access sessions	Mandatory when providing the function
Audit records	Inquiry of audit records	
	Response-related settings for loss of audit records	Mandatory when providing the function

[Table 8] Security management functions to be implemented by TOE

o The TOE shall provide enable/disable functions for all management access.

o If the agent itself has a security management function, the server shall be able to enable/disable the agent setting function.

o The communication service that does not support encrypted communication channels shall be able to be disabled.

o During TOE operation, it shall support the self-test execution periodically or by administrator's request.

o To ensure correct operation, the TOE shall perform the response function implemented on its own or the response function set by the administrator when the self-test fails.

o The TOE shall provide the administrator with the function to perform integrity verification.

o The TOE shall perform the response function implemented on its own or the response function set by the administrator when the integrity verification fails.

o If the TOE provides online update or manual update function, only the update files that have succeeded in validation shall be installed or applied.

o If the TOE does not provide the function of automatically maintaining the existing version when the update installation fails, manual recovery by the administrator shall be supported.

o Locked sessions shall be unlocked by the administrator or through the user authentication function for each session, after the locking time has elapsed.

o Additionally, the TOE may provide a function to send audit records to external log servers by administrator.

- *If syslog is supported, it shall support encrypted transmission through syslog over TLS(RFC 5424), or syslog over DTLS(RFC 6012).*
- The cryptographic algorithm used, cryptographic key security, and cryptographic key storage method shall meet the requirements of 'Protection when storing cryptographic key' of FCS class and FPT_PST.1.

o When agents or clients provide a management function, agents or clients shall provide a security management function that allows users to set and manage their own security functions, security policies and important data.

- If the TOE component includes a server and an agent, the agent must be able to enforce the security policy sent by the server as the agent's setting.
- Guidance documents that identify and describe all the security management functions provided by agents or clients shall be submitted.

o TOE agents or clients shall verify the integrity periodically or upon the authorized administrator's request, and provide the administrator with the result notification function.

5.1.5.2. 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*** [assignment: *list of TSF data*] to [assignment: *the authorized roles*].

Application notes

o The TOE shall provide the authorized administrator with the security management functions to set and manage security functions, security policies, important data, etc.

- The security management functions include the followings:
  - A function to add, delete or change conditions or rules that can determine the operation of the security function.
  - A function to add, remove or change the actions to be performed by the TOE in accordance with the conditions or rules.

- A function to select or change TOE settings

- The security management functions to be implemented by the TOE are shown in [Table 8].

o The administrator shall be able to grant privileges each user or each group.

o The user account(ID) is a unique value and shall not be registered in duplicate.

o The number of consecutive authentication failures in which identification and authentication are deactivated shall be fixed or settable at a value of 5 or less.

o When implementing to deactivate the authentication function for a certain period of time, the time required for re-activation shall be fixed or settable at a value of 5 minutes or more.

o If ID/password is the only means of user identification and authentication, the TOE shall meet the security criteria, <Password Security Criteria Type(1)> of FIA_SOS.1 when registering and changing passwords.

o If ID/password input and additional identification and authentication functions are performed concurrently, the TOE shall meet the security criteria, <Password Security Criteria Type(2)> of FIA_SOS.1 when registering and changing passwords.

o If authentication information necessary for external IT entity authentication is required to be set, the TOE shall provide the function to set the information necessary for external IT entity authentication.

- The application target may be a pre-shared key for the authentication server connection, an SNMP authentication/encryption password, etc.

- When passwords are used for external IT entity authentication, the security criteria, <Password Security Criteria Type(1)> or <Password Security Criteria Type(2)>of FIA_SOS.1 shall be complied with.

o The TOE shall provide a function to limit the IP of the accessible management terminals.

- The IP address of the management terminals shall be able to be registered, deleted or changed.

- Management terminals that can be accessed by administrators who have only read permission instead of for management purpose (e.g., *monitoring administrators, etc.*) can be additionally registered and operated.

- Only one single host IP can be added per time for accessible management terminals.

- A method of specifying an IP address range, such as 192.168.10.2~253, or registration using 0.0.0.0, 192.168.10.*, any, etc. which means the the entire network range is not allowed.

o When providing a function that requires a password to access internal components of the TOE or external IT entities, the TOE shall provide the default password change function used to access internal components or external IT entities.

- Examples of default passwords include DBMS passwords and web server/WAS server passwords.

- If the TOE stores the default password to access the DBMS, the TOE shall provide a function to change the default password.
- Examples of authentication information include the password used to authenticate the TOE in the SMTP server.
- Depending on whether additional identification and authentication functions are concurrently used when generating a password, the security criteria, <Password Security Criteria Type(1)> or <Password Security Criteria Type(2)> of FIA_SOS.1 shall be complied with.
- If a default account(ID) exists in the TOE to access DBMS/Web Server/WAS Server, a function to change it may be provided.

o If an external IT entity interacted with the TOE requests authentication information for TOE authentication, the TOE shall provide a function to set the authentication information required to be authenticated by the external IT entity.

- Examples of authentication information include the password used to authenticate the TOE in the SMTP server.
- It is recommended that passwords should comply with the security criteria, <Password Security Criteria Type (2)> of FIA_SOS.1.
  - However, even the characters included in the password security criteria may not include characters that are not permitted to be entered by the interacted external IT entity.

ㅇ If the TOE includes agents, the TOE shall provide a function to inquire information about the agent.

- The essential inquiry information for the agent is as follows.
  - Agent version, security policy applied to the agent, agent operation status (enabled/disabled), agent integrity verification result (success/failure)
- Additional information about the agent is as follows.
  - *Additional agent attributes, others (operating system information of the managed system where the agent is installed, IP information, other information, etc.), etc.*

o If the TOE includes agents, the TOE shall centrally manage the security policy and provide a function to enforce the server's security policy to the agent.

- If the TOE includes agents, the server must centrally manage the policy and shall be able to enforce the server's security policy regardless of the agent's own security management function.

o The TOE shall provide an interface that allows only authorized administrators to access the TOE settings, and other persons than authorized administrators shall not be able to access the TOE settings.

- Access means operations such as read, change, and delete, etc.

o When providing the function to backup the TOE settings in the form of external file, an encryption function shall be provided.

ㅇ For encryption, the encryption algorithm used, encryption key security, and encryption key storage method shall satisfy the 'protection when storing encryption key' requirements of FCS class and FPT_PST.1.

o The TOE shall provide a function for the administrator to check the contents and results of integrity verification.

- The contents and results of integrity verification shall be confirmed through *screen display, audit records*.

o The TOE shall provide a function for users to check 'the unique identification information of the TOE'.

- The TOE identification information must be unique, can be checked by the user through the interface, and cannot be modified or changed. It shall include the following:

• TOE name, TOE version, TOE release or build number

- If the TOE includes multiple components that are physically separated, the identification information of each component shall be unique, can be checked, and cannot be modified or changed by users. It shall include the following:

• The name and version of the TOE including the component, the component name, the component version, and the component release or build number

- A version management system shall be applied to check the patch of the TOE/components and whether functions are improved.

(e.g., In case of patch and function improvement, a system for changing the major version, minor version, release number, and build number for each case is established to track the reason for the change of TOE/components with version information)

- In case of hardware appliance, users shall be able to view the unique identification information of the firmware in addition to TOE identification information through TOE interface.

ㅇ A certain amount of time, which is the cumulative amount of time after connection that triggers user session locking or session time-out, the administrator can fix the accumulated amount of time from a value of 10 minutes or less, or set it in proportion to the number of authentication failures.

o Audit records shall be inquired only through the security function provided by the TOE.

o The relevant user interface(UI) and CLI commands shall not be provided so that even an authorized administrator cannot delete or change audit records.

o Examples of conditions to notify administrators related to audit record loss response are as follows.

- *90% or more of the setup disk capacity, 100 MB or more, etc.*

o When an agent or a client provides a management function, the agent or client shall provide a security management function that allows users to set and manage their own security functions, security policies, and important data, etc.

- If the TOE component includes a server and an agent, the agent must be able to enforce the security policy sent by the server as the agent's setting.

- A document that identifies and describes all security management functions provided by the agent or client shall be submitted.

5.1.5.3. FMT_PWD.1 Management of ID and password (Extended)

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 [the authorized administrator].

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 [the authorized administrator].

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: *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*].

Application notes

o The user account(ID) is a unique value and shall not be registered in duplicate.

o The TOE shall provide a function to forcibly change/generate the administrator default password during the initial access (management access, local access) to the TOE.

- If there is a default password, the function to change the default password shall be provided during the initial access to the TOE, and then management and local access to the TOE shall be possible.
- If there is no default password, a new password shall be created, and then management and local access to the TOE shall be possible.
  - Passwords shall comply with the security criteria, <Password Security Criteria Type (1)> or <Password Security Criteria Type (2)> of FIA_SOS.1.
- If there is no default account(ID), a new account(ID) shall be created, and then management and local access to the TOE shall be possible.

5.1.5.4. 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: [assignment: *list of management functions to be provided by the TSF*].

5.1.5.5. FMT_SMR.1 Security roles

Hierarchical to	No other components.
Dependencies	FIA_UID.1 Timing of identification

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

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

5.1.6. Protection of the TSF (FPT)

5.1.6.1. 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 the TSF data from disclosure, modification when it is transmitted between separate parts of the TOE.

Application notes

○ The TOE shall transmit using an encrypted channel to protect data transmitted among TOE components (e.g., security policies, control commands, audit records, etc.)

- For secure encrypted communication, confidentiality and integrity shall be provided using standard protocols.
  - Secure cryptographic communication protocols include *HTTPS (implemented using TLS), TLS (TLS 1.2-RFC5246 or higher), SSH (SSH V2-RFC 4251, 4254), etc*.
- Use of its own protocol is not allowed.
- The cryptographic algorithm used, cryptographic key security, and cryptographic key storage method shall satisfy the requirements of 'protection when storing cryptographic key' of FCS class and FPT_PST.1.

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

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 *disclosure, modification*.

Application notes

1. Protection when storing TSF data (important information)

o The TOE shall store important information in a secure way when storing it inside the TOE.

- At least when the TOE stores the following important information, it shall be encrypted and stored.
  - • Password used by the TOE for user identification and authentication
  - • Authentication information used by the TOE for additional identification and authentication
  - • Data Encryption Key(DEK)
- The data encryption key(DEK) shall be encrypted and stored using the key encryption key(KEK).
- Requirements related to generation and storage of key encryption key(KEK) shall satisfy the 'protection when storing encryption key' requirements of FCS_CKM.1(1), FCS_CKM.1(2) and FPT_PST.1.
- When the TOE stores the following information, it must be stored using *encryption, access control, etc.*
  - • Information used for mutual authentication when the TOE and external IT entities are interacted
  - • DBMS/web server/WAS server's administrator password required for the TOE to access DBMS/web server/WAS server that exist inside or outside the TOE.
  - • Encryption key (pre-shared key, symmetric key, private key)
- The user password used by the TOE for user identification and authentication shall be stored using one-way encryption(hash) to prevent decryption.
  - • When performing one-way encryption, it is necessary to add and apply a randomly generated value called salt to the password.
  - • The salt value does not need to be confidential. It shall be generated using a random bit generator and the size must be at least 48 bits.
  - • The iteration count shall be applied as large as possible (at least 1000 times).
- DBMS/Web server/WAS server's administrator password, etc. required for TOE operation can be stored after being encrypted by applying the public key/symmetric key encryption algorithm.
- Encryption key means pre-shared key, symmetric key, private key, etc., and covers all keys

used for TOE management access/local access, and interaction settings among TOE components.

- Passwords and encryption keys included in the minimum important information that shall be encrypted shall not be stored in the TOE by hard-coding.
- The encryption algorithm used, encryption key security, and encryption key storage method shall satisfy the 'protection when storing encryption key' requirements of FCS class and FPT_PST.1.

2. Protection when storing TSF data (setting values, audit records)

ㅇ The TOE shall provide a function to protect the stored TOE setting values (security policies, environment setting parameters, etc.) so that only authorized administrators can access.

- For hardware appliance-type TOE, the TOE settings stored inside shall be protected, and for software-type TOE, the TOE settings stored in the store controlled by the TOE after installation.
- The TOE shall provide an interface that allows only authorized administrators to access TOE settings, and other persons than authorized administrators shall not be able to access TOE settings
  - Access means operations such as read, change, delete, etc.
- When providing the function to backup the TOE settings in the form of external files, an encryption function shall be provided.
- During encryption, the encryption algorithm used, encryption key security, and encryption key storage method shall satisfy the 'protection when storing encryption key' requirements of FCS class and FPT_PST.1.

o If WAS(*Tomcat, Jesus, etc.*) is included in the TOE package, the TOE shall implement not to include important information in the WAS log.

- Important information such as passwords and encryption keys shall not be left in plain text in the WAS log.

o The TOE may safely encrypt and store audit records when they are stored inside the TOE.

- The encryption algorithm used, encryption key security, and encryption key storage method shall satisfy the 'protection when storing encryption key' requirements of FCS class and FPT_PST.1.

3. Protection when storing cryptographic key

o The TOE shall store the cryptographic key in a secure way.

- Data encryption key(DEK) can be stored by using key encryption key(KEK).
- Key Encryption Key(KEK) can be generated through multiple stages of key chain, among which the final key encryption key(KEK) can be encrypted and stored using the key

encryption key(KEK) of the previous stage.

- The key encryption key(KEK) except the final key encryption key(KEK) in the key chain cannot be stored.
- When the cryptographic key is stored outside the TOE, it is recommended to use storage media that have been verified for safety such as smart cards, security USBs, and security tokens(HSM).
  - It is recommended to use a product that has obtained a security function test report or a domestic/foreign CC certificate for the storage media.
- Hard-coding and storing the encryption key in the TOE are not permitted.
- As shown in the [Table 9] below, the applicant shall identify all cryptographic keys used for storage and transmission in the TOE, and prove security by submitting a list and explanatory materials for key storage and destruction methods.

Cryptographic key type	How to store and destroy keys
TLS private key	- Type: RSA Private Key - Generator: Generated by TOE - Storage/Protection: Store in the TOE/Block unauthorized access to TOE storage area - Destruction: Overwrite 3 times with 0 and 1 when executing key destruction command
TLS session encryption key	- Type: ARIA Key - Generator: Generated by TOE - Storage/Protection: Store only in memory(RAM) - Destruction: Overwrite 3 times with 0 and 1 when at the end of the session
TLS session integrity verification key	- Type: HMAC Key - Generator: Generated by TOE - Storage/Protection: Store only in memory(RAM) - Destruction: Overwrite 3 times with 0 and 1 when at the end of the session

[Table 9] How to store and destroy cryptographic keys

- When the TOE stores cryptographic keys (pre-shared key, symmetric key, private key, etc.) used for local/administrative access for TOE management and for interacted setting with separate equipment, it shall be protected and stored in a way such as *encryption, access control, etc.*

4. Protection when storing agent or client TSF data (important information)

○ When the TOE agent or client stores important information in the file system or registry, the agent or client stores important information in the file system or registry, it shall be encrypted and stored.

- At least when the TOE stores the following important information, it shall be encrypted and stored.

  • User password

  • Encryption key (pre-shared key, symmetric key, private key)

- User password includes agent deletion key, and password shall be stored using one-way encryption(hash) not to be generally decrypted.

  • When performing one-way encryption, it is necessary to add a randomly generated salt to the password.

  • The salt value does not need to be confidential. It shall be generated using a random bit generator and it is the size of at least 48 bits.

  • The iteration count shall be applied as large as possible. (at least 1000 times)

- Encryption key means pre-shared key, symmetric key, private key, etc., and covers all keys used for TOE management access/local access, and interacting settings among TOE components.

- Passwords and encryption keys included in the minimum important information that shall be encrypted shall not be stored in the TOE by hard-coding.

- The encryption algorithm used, encryption key security, and encryption key storage method shall satisfy the 'protection when storing encryption key' requirements of FCS class and FPT_PST.1.

- Even if encryption is provided, it is recommended to protect in a way to additional file hiding, access control, etc.

5. Protection when storing agent or client TSF data (setting values, audit data)

o When storing TOE settings and audit data in the file system or registry, a function to protect against unauthorized access may be provided.

- The relevant user interface(UI) and CLI commands shall not be provided to prevent deletion or modification of audit data even by agent users.

- Even agent users shall not be able to access the stored TOE settings.

  • Access means operations such as read, change, and delete.

- If the TOE security function cannot be fully implemented, it can be supported to protect the TOE settings storage in the TOE operating environment.

- When providing the function to backup the TOE settings in the form of external file, an encryption function shall be provided.

6. Protection when storing TSF data related to DB encryption (cryptographic key, critical security parameters)

o When the TOE stores the encryption key or critical security parameters, it shall be encrypted with the key encryption key(KEK) through the encryption algorithm of the validated cryptographic module to store safely.

- The stored encryption key or critical security parameters shall be stored by using the key encryption key(KEK) generated in accordance with the FCS_CKM.1(1) requirements.

5.1.6.3. 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 [selection: *at the initial start-up, periodically during normal operation, upon the request of authorized user, at the conditions [assignment: conditions under which self-test should occur]* to demonstrate the correct operation of [selection: [assignment: *parts of TSF], the TSF*].

FPT_TST.1.2 The TSF shall provide authorized users with the capability to verify the integrity of [selection: [assignment: *parts of TSF data], TSF data*].

FPT_TST.1.3 The TSF shall provide authorized users with the capability to verify the integrity of [selection: [assignment: *parts of TSF], TSF*].

Application notes

1. TOE server self-test, response function, and audit record generation

o The TOE shall perform self-test during initial start-up(or execution)/operation periodically or at the request of the administrator.

- When initial start-up(or execution) the TOE, it is mandatory to perform self-test, and during operation, it shall support the perform self-test periodically or at the request of the administrator.
- The self-test target means the main process of the TOE, and shall check whether the process is running normally.
- The subject of self-test can be selected by the applicant, but if the entity's abnormal state(*e.g., error, stop, etc.*) affects the security function of the TOE, the corresponding entity shall be included as the subject of self-test.
- The history of self-testing shall be confirmed through *screen output, audit records.*
- The hardware appliance-type TOE shall satisfy the following requirements.
  - A self-test shall be performed to detect errors in hardware(*e.g., memory, flash, NIC, etc.*) and

software(*e.g., process, etc.*) included in the scope of the TOE at the start-up and during operation of the TOE.

- If physically separated TOE components exist, self-test shall be performed by selecting the subjects to include all components.
- The sponsors shall describe the self-test function in detail in the submission document.

o If the TOE self-test result is a failure, it shall perform the response function.

- The TOE shall perform the implemented response function or the response function set by the administrator to ensure correct operation.
- Audit records shall be generated for self-test results.
- Examples of response functions performed when the self-test result is a failure are as follows.
  - *Termination of program, warning message screen display, restart process, etc.*
- A security management function may be provided for the administrator to set the response function.

2. TOE server integrity verification, response function, and audit record generation

o The TOE shall provide a function to verify the integrity of itself and its setting values.

- Integrity verification covers the TOE setting values(*configuration files, etc.)* and the TOE itself(*processes, libraries, executable files, etc.*).
- Integrity verification shall be performed when the TOE is initial executed(or start-up), and periodic integrity verification can be performed additionally.
- The subject of integrity verification can be selected by the sponsor, but if the entity's abnormal state(e.g., *error, stop, etc.*) affects the security function of the TOE, the corresponding entity shall be included as the subject of integrity verification.
- If physically separated TOE components exist, integrity verification shall be performed by selecting the subjects to include all components.
- A function for the administrator to perform integrity verification shall be provided.
- The cryptographic algorithm used, cryptographic key security, and cryptographic key storage method shall satisfy 'protection when storing cryptographic key' requirements of FCS class and FPT_PST.1.

o If the operating system kernel or kernel level module is included in the scope of the TOE, the TOE shall provide a function to verify the integrity of the operating system kernel or kernel level module.

- When verifying integrity by hash value comparison method, the cryptographic algorithm used, cryptographic key security, and cryptographic key storage method shall satisfy 'protection when storing cryptographic key' requirements of FCS class and FPT_PST.1.

o The TOE shall provide a function for the administrator to check the contents and results of the integrity verification.

- The contents and results of the integrity verification shall be checked through *screen display and audit records.*

o The TOE shall perform response function if integrity verification fails.

- The TOE shall perform its own implemented response function or the response function set by the administrator.
- Audit records shall be generated for integrity verification results.
- Examples of response functions performed when the integrity verification result is a failure are as follows.
  - *Interrupt program execution, warning message screen display, etc.*
- A security management function may be provided for the administrator to set the response function.

3. TOE agents, clients, management consoles integrity verification, response function, and audit record generation

o The agent or client shall provide the function to verify the integrity of the TOE setting values and its own at the initialization phase and periodically or at the request of authorized administrators.

- Integrity verification covers agent or client setting values(*policies, environment settings, etc.*) and the TOE itself (*executable files, filter drivers, etc.*).
- In the case of a TOE running on a Windows® operating system, the modification shall be detected during normal booting of the operating system, if integrity is compromised in the safe mode of the operating system.
- In the case that integrity verification is performed periodically or at the request of authorized administrators, △when an abnormality occurs in the integrity verification result, △the integrity verification result by the administrator shall be notified to the administrator.
- Audit records shall be generated for integrity verification results.
- Cryptographic-related parts shall satisfy the ‘protection when storing encryption keys’ requirements of FCS class and FPT_PST.1.

ㅇ The agent or client shall provide a function to can recover modified information(*setting values, executable files, filter drivers, etc.*).

- 'Modified information' shall identify and include all files that affect the normal operation and of security functions of the TOE.
- 'Agent Type1' shall provide an automatic recovery function, and △Agent Type2 △Agent Type 3 and △Client Type may provide a manual recovery function.

o In the case of an agent or client installed on the endpoint in Windows® environment, the agent or client shall provide an integrity verification function for the server/update server address.

o If there are two or more servers or update servers on the file transfer path, the receiving server shall perform integrity verification for the address of the sending server.

5.1.7. TOE access (FTA)

5.1.7.1. 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 belonging to the same user according to the rules [limiting the maximum number of concurrent sessions to 1 for users who have the same privilege and the same user, rules on the maximum number of concurrent sessions {determined by the ST author}].

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

Application notes

o The TOE shall not allow duplicate access to the TOE with the same user account or the same privilege.

- If a user logs in with the same account on another terminal after logging in, it is required to block a new access or terminate the previous access.
- Duplicate logins with the same privilege shall not be allowed.
- An audit record should be generated when duplicate access is blocked.

5.1.7.2. FTA_TSE.1(1) TOE session establishment

Hierarchical to	No other components.
Dependencies	No dependencies.

FTA_TSE.1.1 The TSF shall be able to deny **the administrator’s management access session** establishment based on [access IP, [selection: [assignment: *important management function attributes], none*]].

Application notes

o The TOE shall provide a function to restrict the IP of the accessible management terminals.

- It shall be possible to register, delete, and change the IP address of the management terminals.

- Management terminals accessible to administrators who only have read access instead of for management purposes(e.g., *monitoring administrators*) can be additionally registered for operation.
- The IP of accessible management terminals can be added one by one at a time as a host IP.
- It is not allowed to register by designating an IP address range such as 192.168.10.2~253, or by using 0.0.0.0, 192.168.10.*, any, which means the entire network range.

5.2. Security functional requirement (Conditional mandatory SFR)

'Conditional mandatory SFRs' in this PP are as follows. 'Conditional mandatory SFRs' mandatorily require to be included in the ST, if they meet 'the additional conditions for the ST' in the table below.

Security functional class	Security functional component		SFR additional conditions	Remark
FAU	FAU_STG.1	Protected audit trail storage	In case of the TOE server stores audit records in local storage	
	FAU_STG.3	Action in case of possible audit data loss	In case of the TOE server stores audit records in local storage	
	FAU_STG.4	Prevention of audit data loss	In case of the TOE server stores audit records in local storage	
FIA	FIA_UAU.5	Multiple authentication mechanisms	In case of the TOE server supports additional identification and authentication functions by itself in addition to the ID/password-based authentication method	
FPT	FPT_LEE.1	Linkable external entities (Extended) - authentication	In case of the TOE server supports additional identification and authentication functions by interacting with external IT entities in addition to the ID/password-based authentication method	
	FPT_RCV.1	Manual recovery	In case of TOE components include agents or clients	
	FPT_RCV.2	Automated recovery	In case of TOE server update function is provided	
	FPT_TUD.1	TSF security patch update (Extended)	In case of TOE update function is provided	
FTA	FTA_SSL.1	TSF-initiated session locking	In case of TOE provides session locking function	One of the two must be implemented
	FTA_SSL.3	TSF-initiated termination	In case of TOE provides session termination function	
	FTA_TSE.1(2)	TOE session establishment	In case of it is necessary to identify and authenticate users existing in the agent, management console, or client constituting the TOE	
FTP	FTP_ITC.1	Inter-TSF trusted channel	In case of interacting with external IT entities is supported In case of audit records are transmitted and stored to external IT entities in real time In case of providing the online update function through the developer update server.	
	FTP_TRP.1	Trusted path	In case of authorized administrators	

Security functional class	Security functional component		SFR additional conditions	Remark
			and general users directly access the management server through web browsers or terminal access programs, etc.	

[Table 10] Conditional mandatory SFRs

5.2.1. Security audit (FAU)

5.2.1.1. FAU_STG.1 Protected of audit trail storage

Hierarchical to No other components

Dependencies FAU_GEN.1 Audit data generation

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

FAU_STG.1.2 The TSF shall be able to *prevent* unauthorised modifications to the stored audit records in the audit trail.

Application notes

o The TOE shall protect the audit records from being deleted or changed.

- A function shall be implemented to store audit records in a local storage or to transmit and store audit records to an external IT entity in real time.
- Relevant user interface(UI) and CLI commands shall not be provided so that even authorized administrators cannot delete or change audit records.
- Unauthorized person's access shall be controlled to protect the stored audit records.
- If the TOE security function cannot be fully implemented, the TOE operational environment can support the protected audit trail storage.
  - Example: When audit records are stored in the DBMS installed on the same operating system as the TOE, the DBMS' identification and authentication functions can be used to protect deletion or modification by unauthorized users.
- If audit records are stored in the log server outside the TOE, encrypted communication shall be performed.
  - If syslog is supported, encrypted transmission shall be supported through *syslog over DTLS(RFC 5424), syslog over DTLS(RFC 6012), etc.*

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

Hierarchical to No other components

Dependencies FAU_STG.1 Protected of audit trail storage

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

Application notes

o In case of the size of the audit record reaches the predefined capacity, the TOE shall take response actions such as *notifying the administrator*.

- A function shall be implemented to store audit records in the local storage or to transmit and store audit records to an external IT entity in real time.
- A function to notify the administrator shall be provided. Examples of the function are as follows.
  - *Screen alarm, sending email to the administrator, etc.*
- Examples of conditions for notifying the administrator in response to audit record loss are as follows.
  - *90% or more of the setup disk capacity, 100MB or more, etc.*
- In addition, a function for the administrator to send audit records to an external log server may be provided.
  - If syslog is supported, encrypted transmission shall be supported through *syslog over DTLS(RFC 5424), syslog over DTLS(RFC 6012), etc.*
  - The cryptographic algorithm used, cryptographic key security, and cryptographic key storage method shall satisfy the 'protection when storing cryptographic key' requirements of FCS class and FPT_PST.1.

5.2.1.3. 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 [selection, choose one of: *"ignore audited events", "prevent audited events, except those taken by the authorised user with special rights", "overwrite the oldest stored audit records"*] and [assignment: *other actions to be taken in case of audit storage failure*] if the audit trail is full.

Application notes

ㅇ In case of the audit record storage capacity is full, the TOE shall respond to failure to save in an appropriate way.

- A function shall be implemented to store audit records in a local storage or to transmit and store audit records to an external IT entity in real time.
- Examples of response functions in case of failure to save are as follows.
  - *Overwriting the oldest audit records, save audit records compression, etc.*

5.2.2. Identification and authentication (FIA)

5.2.2.1. FIA_UAU.5 Multiple authentication mechanisms

Hierarchical to	No other components.
Dependencies	No dependencies.

FIA_UAU.5.1	The TSF shall provide [password authentication mechanism, [assignment: l*ist of additional authentication mechanism]*] to support user authentication.
FIA_UAU.5.2	The TSF shall authenticate any user's claimed identity according to the [assignment: *rules describing how the multiple authentication mechanisms provide authentication*].

Application notes

o In case of the TOE supports additional identification and authentication methods, the TOE shall provide additional identification and authentication functions on its own or by interacting with external IT entities, in parallel with user account/password-based identification and authentication.

- In order to provide additional identification and authentication functions, *△2FA support device complying with FIDO standards, △certificates, △one-time password generator(OTP), etc.* can be used.
  - If it is supported in the TOE operational environment, '2FA support device complying with FIDO standards' is recommended.
- If additional identification and authentication functions are provided in the TOE, the functions can be provided by receiving the authentication results from the inside of the TOE or from the interacted external IT entities.
  - If the TOE provides a certification utilization method, certificate validation shall be performed.
  - The authentication information used by external IT entities to perform additional identification and authentication methods shall be securely managed by the external IT entities. If the TOE stores authentication information use to perform additional identification and authentication methods, the requirements of FPT_PST.1 shall be applied.

5.2.3. Protection of the TSF (FPT)

5.2.3.1. FPT_LEE.1 Linkable external entities (Extended) - authentication

Hierarchical to	No other components.
Dependencies	No dependencies.

FPT_LEE.1.1	The TSF shall perform [assignment: *list of actions*] and provide [assignment:

list of functions] by linking with external entities.

Application notes

o In case of the TOE supports additional identification and authentication methods, the TOE shall provide additional identification and authentication functions on its own or by interacting with external IT entities, in parallel with user account/password-based identification and authentication.

- In order to provide additional identification and authentication functions, *△2FA support device complying with FIDO standards, △certificates, △one-time password generator(OTP), etc.* can be used.
  - If it is supported in the TOE operational environment, '2FA support device complying with FIDO standards' is recommended.
- If additional identification and authentication functions are provided in the TOE, the functions can be provided by receiving the authentication results from the inside of the TOE or from the interacted external IT entities.
  - If the TOE provides a certification utilization method, certificate validation shall be performed.
  - The authentication information used by external IT entities to perform additional identification and authentication methods shall be securely managed by the external IT entities. If the TOE stores authentication information use to perform additional identification and authentication methods, the requirements of FPT_PST.1 shall be applied.

5.2.3.2. FPT_RCV.1 Manual recovery

Hierarchical to	No other components.
Dependencies	AGD_OPE.1 Operational user guidance

FPT_RCV.1.1 After [assignment: *list of failures/service discontinuities*] the TSF shall enter a maintenance mode where the ability to return to a secure state is provided..

Application notes

ㅇ The agent or client shall provide a function to can recover modified information(*setting values, executable files, filter drivers, etc.*).

- 'Modified information' shall identify and include all files that affect the normal operation and of security functions of the TOE.
- 'Agent Type1' shall provide an automatic recovery function, and △Agent Type2 △Agent Type3 and △Client Type may provide a manual recovery function.

5.2.3.3. FPT_RCV.2 Automated recovery

Hierarchical to	FRP_RCV.1 Manual recovery

Dependencies	AGD_OPE.1 Operational user guidance
FPT_RCV.2.1	When automated recovery from [assignment: *list of failures/service discontinuities*] is not possible, the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.
FPT_RCV.2.2	For [assignment: *list of failures/service discontinuities*], the TSF shall ensure the return of the TOE to a secure state using automated procedures.

Application notes

o If the update function is provided, the TOE shall provide a function to automatically maintain the existing version when the update installation fails.

- If it is not supported by the TOE, manual recovery by the administrator shall be supported.
- The sponsor shall describe the manual recovery procedure by the administrator in detail in the deliverables.

5.2.3.4. FPT_TUD.1 TSF security patch update (Extended)

Hierarchical to	No other components.
Dependencies	No dependencies.
FPT_TUD.1.1	The TSF shall provide the capability to view the **unique identification** to [assignment: *the authorized identified roles*].
FPT_TUD.1.2	The TSF shall verify validity of the update files using [selection: *hash value comparison, digital signature verification,* ***[assignment: other secure validation mechanism]***] before installing updates.

Application notes

○ The TOE shall provide a function for users to check the 'unique identification information of the TOE'.

- The TOE identification information shall be unique, can be checked by users through the interface, and cannot be modified or changed. It shall include the following.
  - TOE name, TOE version, TOE release or build number
- If the TOE includes multiple components that are physically separated, the identification information of each component shall be unique, can be checked, and cannot be modified or changed by users. It shall include the following:
  - The name and version of the TOE including the component, The component name, The component version, The component release or build number.
- A version management system that can check whether the TOE and TOE components are patched and functionally improved should be applied.

(e.g., In case of patch and function improvement, a system for changing the major version, minor version, release number, and build number for each case is established to track the reason for the change of TOE/TOE components with version information)

- In case of hardware appliances, users shall be able to view the unique identification information of the firmware in addition to TOE identification information through TOE interface.

o In case of providing the update function, the TOE shall verify the validity of the TOE update files before installing or applying the update files.

- If the TOE provides online update or manual update function, only the update files that have succeeded in verification of the validity shall be installed or applied.
- Integrity verification is mandatory when verify the validity of the update files, and shall be implemented using *digital signature verification, public hash value verification, etc.*
- When verifying the digital signature, verification of the validity of the certificate (within 1 year of validity) shall be performed.
- Cryptographic algorithm and cryptographic key security shall satisfy FCS class requirements.
- Update file validation results (success・failure) shall be audited and recorded.

o If the update function is provided, the TOE shall provide a function to automatically maintain the existing version when the update installation fails.

- An audit record shall be generated for the update installation result and the reason for failure.
- If it is not supported by the TOE, manual recovery by the administrator shall be supported.
- The developer shall describe the manual recovery procedure by the administrator in detail in the deliverables.

o In the case of an agent or client installed on the endpoint in Windows® environment, the agent or client shall perform the digital signature verification on the subject of file generation of the update target files received from the server or update server.

- It shall be applied to the agent or client existing on the endpoint where Windows® operating system is installed.
- All files that are irrelevant to TOE configuration without being included in installation files and policy files(e.g., patch files, general executable files, etc.) are not allowed to be distributed to agents and clients.
- In case of verifying the digital signature, verification of the validity of the certificate(within 1 year of validity) shall be performed.
- The update file digital signature verification result(success, failure) shall be recorded in the audit record.
- The cryptographic-related part shall satisfy the FCS class requirements.
- Developers or administrators (who perform digital signatures on update files) shall perform digital signatures on the separate offline server that is disconnected from the Internet.

- o In the case of an agent or client installed on the endpoint in Windows® environment, the agent or client shall provide an integrity verification function for the server/update server address.
- o If there are two or more servers or update servers on the file transfer path, the receiving server shall perform integrity verification for the address of the sending server.

5.2.4. TOE access (FTA)

5.2.4.1. FTA_SSL.1 TSF-initiated session locking

Hierarchical to	No other components.
Dependencies	FIA_UAU.1 Timing of authentication

FTA_SSL.1.1 The TSF shall lock the interactive session after [assignment: *time interval of user inactivity*] by:

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

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

FTA_SSL.1.2 The TSF shall require the following events to occur prior to unlocking the session: [*[selection: unlocking session by the administrator, user re-authentication before unlocking session]*].

Application notes

- o The TOE shall provide a function to lock or terminate the session if it is not used for a certain period of time after the user session is connected.
  - The time information used shall be applied based on the server time.
  - A certain period of time refers to the amount of time accumulated after a connection that triggers session locking or termination.
    - • A certain period of time can be fixed by the administrator among 10 minutes or less or set in proportion to the number of authentication failures.
  - After the lock time has elapsed, a locked session shall be unlocked by the administrator or through the user authentication function for each session.
  - An audit record shall be generated when the session lock or termination function is activated.
  - It shall be applied to all management and local access included in the TOE.

5.2.4.2. FTA_SSL.3 TSF-initiated termination

Hierarchical to	No other components.
Dependencies	No dependencies.

FTA_SSL.3.1 The TSF shall terminate an interactive session after a [assignment: *time interval of user inactivity*].

Application notes

o The TOE shall provide a function to lock or terminate the session if it is not used for a certain period of time after the user session is connected.

- The time information used shall be applied based on the server time.
- A certain period of time refers to the amount of time accumulated after a connection that triggers session locking or termination.
  - A certain period of time can be fixed by the administrator among 10 minutes or less or set in proportion to the number of authentication failures.
- After the lock time has elapsed, a locked session shall be unlocked by the administrator or through the user authentication function for each session.
- An audit record shall be generated when the session lock or termination function is activated.
- It shall be applied to all management and local access included in the TOE.

5.2.4.3. FTA_TSE.1(2) TOE session establishment

Hierarchical to	No other components.
Dependencies	No dependencies.

FTA_TSE.1.1 The TSF shall be able to deny session establishment based on [assignment: ***list of additional attributes of agent or client***.]

Application notes

o In case of it is necessary to identify and authenticate a user existing in the agent or client constituting the TOE, the identification value shall be a unique value that is not registered in duplicate.

- During user authentication, additional attributes of the registered agent or client shall also be authenticated.
- Additional attributes: IP address is mandatory, and at least one of *MAC address, serial number, and information that can uniquely identify the agent itself* shall be additionally used.

5.2.5. Trusted path/channels (FTP)

5.2.5.1. FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to	No other components.
Dependencies	No dependencies.

FTP_ITC.1.1	The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2	The TSF shall permit [selection: *the TSF, another trusted IT product*] to initiate communication via the trusted channel.
FTP_ITC.1.3	The TSF shall initiate communication via the trusted channel for [assignment: *list of functions for which a trusted channel is required*].

Application notes

o In case of interacting with external IT entities is supported, the TOE shall transmit data using an encrypted communication channel to protect the transmitted data when interacting with external IT entities.

- For secure cryptographic communication, confidentiality and integrity shall be provided using standard protocols.
  - Secure cryptographic communication protocols include *HTTPS (implemented using TLS), TLS (TLS 1.2-RFC5246 or higher), SSH (SSH V2-RFC 4251, 4254), etc.*
- Use of its own protocol is not allowed.
- The cryptographic communication channel can be implemented directly in the TOE or to be provided by the TOE using the operating environment.
- This requirement shall be applied when the TOE provides a function that interacting with external IT entities to provide a security function.
- If transmission data is not protected using an cryptographic communication channel when interacting with external IT entities, the needlessness to protect the confidentiality and integrity of transmitted data shall be proven.
- Communication services that do not support cryptographic communication channels shall be able to be disabled.
- The cryptographic algorithm used, cryptographic key security, and cryptographic key storage method shall satisfy the requirements of 'protection when storing cryptographic key' of FCS class and FPT_PST.1.

o In case of audit records are stored in a log server outside the TOE, cryptographic communication shall be performed.

- If syslog is supported, encrypted transmission shall be supported through *syslog over DTLS(RFC 5424), syslog over DTLS(RFC 6012), etc.*

5.2.5.2. FTP_TRP.1 Trusted path

Hierarchical to	No other components.
Dependencies	No dependencies.

FTP_TRP.1.1 The TSF shall provide a communication path between itself and [selection: *remote, local*] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from *modification, disclosure, [assignment: other types of integrity or confidentiality violation]*.

FTP_TRP.1.2 The TSF shall permit [selection: *the TSF, local users, remote users*] to initiate communication via the trusted path.

FTP_TRP.1.3 The TSF shall require the use of the trusted path for [selection: ***the authentication of management access administrator***, *[assignment: other services for which trusted path is required]*].

Application notes

○ During management access, the TOE shall transmit data using an cryptographic communication channel to protect the transmitted data.

- For secure cryptographic communication, confidentiality and integrity shall be provided using standard protocols.
  - Secure cryptographic communication protocols include *HTTPS (implemented using TLS), TLS (TLS 1.2-RFC5246 or higher), SSH (SSH V2-RFC 4251, 4254), etc*.
- Use of its own protocol is not allowed.
- The cryptographic communication channel can be implemented directly in the TOE or to be provided by the TOE using the operational environment.
- The cryptographic algorithm used, cryptographic key security, and cryptographic key storage method shall satisfy the requirements of 'protection when storing cryptographic key' of FCS class and FPT_PST.1.

5.3. Security function requirements (optional SFRs)

The 'optional SFRs' in this PP are as follows. The 'optional SFRs' are not required to be implemented mandatorily, but if the TOE provides relevant functions additionally, the ST author shall include the corresponding SFRs in the ST.

Security function class	Security functional component	
Cryptographic support (FCS)	FCS_CKM.2	Cryptographic key distribution
Protection of the TSF (FPT)	FPT_STM.1	Reliable timestamp

[Table 11] Optional SFRs

5.3.1. Cryptographic support (FCS)

5.3.1.1. 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 [assignment: *cryptographic key distribution method*] that meets the following: [assignment: *list of standards*].

Application notes

o FCS_CKM.2 cryptographic key distribution is a selectively implementable functional requirement('optional SFRs'), and if the TOE additionally provides the above function, the ST author shall include this requirement in the SFR.

o If the ST author includes this SFR, security problem definition and security objectives shall be derived additionally, if necessary.

o The key used in the cryptographic key establishment method defined in FCS_CKM.2.1 shall be related to the key generated in FCS_CKM.1.1 of FCS_CKM.1(1) and FCS_CKM.1(2).

5.3.2. Protection of the TSF (FPT)

5.3.2.1. FTP_STM.1 Reliable time stamps

Hierarchical to	No other components.
Dependencies	No dependencies.

FTP_STM.1.1 The TSF shall be able to provide the reliable timestamp.

Application notes

o Each component of the TOE shall generate audit records using trusted time information.

- Trusted time information shall use the time information provided by the NTP server or operating system.

5.4. 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 12] Security assurance requirements

5.4.1. Security Target evaluation

5.4.1.1. 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.

5.4.1.2. 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.

5.4.1.3. 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.

5.4.1.4. 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.

5.4.1.5. 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.

5.4.1.6. 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

Evaluator action 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.

5.4.2. Development

5.4.2.1. 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.

5.4.3. Guidance documents

5.4.3.1. 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.

5.4.3.2. 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.

5.4.4. Life-cycle support

5.4.4.1. ALC_CMC.1 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.

5.4.4.2. 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.

5.4.5. Tests

5.4.5.1. 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.

5.4.5.2. 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.

5.4.6. Vulnerability assessment

5.4.6.1. 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.

5.5. Security requirements rationale

5.5.1. Dependency rationale of security functional requirements

The following table shows dependency of security functional requirements.

No.	Security functional requirements	Dependency	Reference No.	SFR type
1	FAU_ARP.1	FAU_SAA.1	3	
2	FAU_GEN.1	FPT.STM.1	Rationale(1)	Mandatory
3	FAU_SAA.1	FAU_GEN.1	2	Mandatory
4	FAU_SAR.1	FAU_GEN.1	2	Mandatory
5	FAU_SAR.3	FAU_SAR.1	4	Mandatory
6	FAU_STG.1	FAU_GEN.1	2	Conditional mandatory
7	FAU_STG.3	FAU_STG.1	Rationale(2)	Conditional mandatory
8	FAU_STG.4	FAU_STG.1	Rationale(2)	Conditional mandatory
9	FCS_CKM.1(1)	[FCS_CKM.2 or FCS_COP.1]	11, 13	Mandatory
		FCS_CKM.4	12	
10	FCS_CKM.1(2)	[FCS_CKM.2 or FCS_COP.1]	11, 14	Mandatory
		FCS_CKM.4	12	
11	FCS_CKM.2	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9, 10	Optional
		FCS_CKM.4	12	
12	FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9, 10	Mandatory
13	FCS_COP.1(1)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9	Mandatory
		FCS_CKM.4	12	
14	FCS_COP.1(2)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	10	Mandatory
		FCS_CKM.4	12	
15	FCS_RBG.1	-	-	Mandatory
16	FDP_UDE.1	FCS_COP.1	13	Mandatory
17	FDP_RIP.1	-	-	Mandatory
18	FIA_AFL.1	FIA_UAU.1	21	Mandatory
19	FIA_IMA.1	-	-	Mandatory
20	FIA_SOS.1	-	-	Mandatory

No.	Security functional requirements	Dependency	Reference No.	SFR type
21	FIA_UAU.1	FIA_UID.1	25	Mandatory
22	FIA_UAU.4	-	-	Mandatory
23	FIA_UAU.5	-	-	Conditional mandatory
24	FIA_UAU.7	FIA_UAU.1	21	Mandatory
25	FIA_UID.1	-	-	Mandatory
26	FMT_MOF.1	FMT_SMF.1	29	Mandatory
		FMT_SMR.1	30	
27	FMT_MTD.1	FMT_SMF.1	29	Mandatory
		FMT_SMR.1	30	
28	FMT_PWD.1	FMT_SMF.1	29	Mandatory
		FMT_SMR.1	30	Mandatory
29	FMT_SMF.1	-	-	Mandatory
30	FMT_SMR.1	FIA_UID.1	25	Mandatory
31	FPT_ITT.1	-	-	Mandatory
32	FPT_LEE.1	-	-	Conditional mandatory
33	FPT_PST.1	-	-	Mandatory
34	FPT_RCV.1	AGD_OPE.1	-	Mandatory
35	FPT_RCV.2	AGD_OPE.1	-	Conditional mandatory
36	FPT_STM.1	-	-	Conditional mandatory
37	FPT_TST.1	-	-	Mandatory
38	FPT_TUD.1	-	-	Conditional mandatory
39	FTA_MCS.2	FIA_UID.1	25	Mandatory
40	FTA_SSL.1	FIA_UAU.1	21	Conditional mandatory
41	FTA_SSL.3	-	-	Conditional mandatory
42	FTA_TSE.1(1)	-	-	Mandatory
43	FTA_TSE.1(2)	-	-	Conditional mandatory
44	FTP_ITC.1	-	-	Conditional

No.	Security functional requirements	Dependency	Reference No.	SFR type
				mandatory
45	FTA_TRP.1	-	-	Conditional mandatory

[Table 13] Rationale for the dependency of the security functional requirements

The ST author refers to the table above and prepares a dependency relationship rationale table for the SFRs included in the ST.

Rationale(1) : FAU_GEN.1 has the dependency on FAU_STG.1. However, if the pertinent function is implemented by the TOE, the ST author needs to identify the optional SFR (FAU_STM.1) as the SFR of the ST and describe the pertinent reference number. In addition, if FAU_STM.1 is supported by the operational environment, the author shall add the security objectives for the operational environment and provide justification that a subordinate relationship is satisfied.

Rationale(2) : FAU_STG.3 and FAU_STG.4 have the dependency on FAU_STG.1. However, if the pertinent function is implemented by the TOE, the ST author needs to identify the optional SFR (FAU_STG.1) as the SFR of the ST and describe the pertinent reference number. In addition, if FAU_STG.1 is supported by the operational environment (e.g., DBMS), the author shall add the security objectives for the operational environment and provide justification that a subordinate relationship is satisfied.

5.5.2. 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 PP since it is not necessarily required to show the correspondence between the tests and the TSFIs.

References

Title	Author	Remark
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) • Common Criteria for Information Technology Security Evaluation. Part 2: Security Functional Components, Version 3.1, Revision 5 (CCMB-2017-04-002) • Common Criteria for Information Technology Security Evaluation. Part 3: Security Assurance Components, Version 3.1, Revision 5 (CCMB-2017-04-003)	CCMB	2017. 4
Security Requirements for Government V3.0 for the Information Security Systems and Network Devices - Part 2, Common Security Requirements	National Cybersecurity Center, IT Security Certification Center	2021. 4, 2021. 9
Database Encryption Product Testing Criteria	National Cybersecurity Center, IT Security Certification Center	2022. 3

Abbreviated terms

CBC	Cipher Block Chaining
CC	Common Criteria
CCMB	Common Criteria Maintenance Board
CFB	Cipher Feedback
CTR	Counter Mode
ECB	Electronic Codebook
DEK	Data Encryption Key
EAL	Evaluation Assurance Level
HMAC	Hash-based Message Authentication Code
HTTPS	Hypertext Transfer Protocol over Secure Socket Layer
IP	Internet Protocol
IPSec	Internet Protocol Security
IT	Information Technology
IV	Initial Vector
KEK	Key Encryption Key
NTP	Network Time Protocol
OFB	Output Feedback
SFP	Security Function Policy
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SMS	Short Message Service
SSH	Secure Shell
SSL	Secure Socket Layer
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality