MagicDBPlus v2.1
Security Target

v1.2

< **Change History >**

Version	Date	Contents / Prepared by
1.0	May 26, 2026	Formulated / Dreams Security, Future Technology Lab, Cryptographic Key Application Technology Team
1.1	June 9, 2026	Component version update (v2.1.0.0 -> v2.1.0.1)
1.2	July 6, 2026	Extended SFR correction

< Table of Contents >

1 Security Target Introduction

1.1 Security Target Reference

Cls.	Contents
Title	MagicDBPlus v2. 1 Security Target
ST Version	v1.2
Prepared by	Dreamsecurity Co.,Ltd, Dreams Security, Future Technology Lab, Cryptographic Key Application Technology Team
Date Prepared	June 9, 2026
Evaluation Criteria	Information Security System Common Criteria CC:2022 R1 - Part 1: Introduction and General Model, CC:2022 R1 (CCMB-2022-11-001) - Part 2: Security Functional Components, CC:2022 R1 (CCMB-2022-11-002) - Part 3: Security Assurance Components, CC:2022 R1 (CCMB-2022-11-003) - Part 4: Framework for Evaluation Methodology, CC:2022 R1 (CCMB-2022-11-004) - Part 5: Pre-defined Security Requirement Packages, CC:2022 R1 (CCMB-2022-11-005) - Errata and Interpretation for CC:2022 (R1) and CEM:2022 (R1), v1.2, CCMB-2025-001, 2025.10
Protection Profile	National Database Encryption Protection Profile V3.1
EAL	EAL1+(ATE_FUN.1)
Keywords	Database (DB), DBMS, Encryption, Decryption, Oracle, DB2

1.2 TOE Reference

Cls.		Contents
TOE		MagicDBPlus v2.1
Version		v2.1.0.1
TOE Component	Management Server	MagicDBPlus v2.1 Server v2.1.0.1 : MagicDBPlus_v2.1_Server_v2.1.0.1.sh
	Administrative Tool	MagicDBPlus v2.1 Admin v2.1.0.1 : MagicDBPlus_v2.1_Admin_v2.1.0.1.exe
	Agent	MagicDBPlus v2.1 Agent v2.1.0.1 : MagicDBPlus_v2.1 Agent_v2.1.0.1.sh
Guidance Document	Installation Guide	MagicDBPlus v2.1 Installation Guide v1.1 : MagicDBPlus_v2.1_PRE_v1.1.pdf
	Operational Guidance	MagicDBPlus v2.1 Operational Guidance v1.1 : MagicDBPlus_v2.1_OPE_v1.1.pdf
Developer		Dreamsecurity Co.,Ltd, Future Technology Lab, Cryptographic Key Application Technology Team

1.3 TOE Overview

1.3.1 Overview of Database Encryption

MagicDBPlus v2.1 (hereinafter referred to as “TOE”) performs a role in encrypting database (“DB”) and preventing unauthorized disclosure of information to be protected.

Subjected to encryption of the TOE, the DB is controlled by the Database Management System (“DBMS”) in an organization’s operational environment. In this ST, all the data before and after being encrypted and stored in the DB are defined as user data. According to security policy of the organization that operates the TOE, the whole or part of the user data could be subject to encryption.

1.3.2 TOE Type and Scope

The TOE is provided in a form of software and offers a function of encrypting/decrypting user data as per column. Types of the TOE defined herein include a “Plug-in” under which the TOE is comprised of an Agent, a Management Server and an Administrative Tool.

TOE components are shown in the following [Table 1-1].

[Table 1-1] TOE Components

Component	Contents
MagicDBPlus v2.1 Server v2.1.0.1	Serves as storage of cryptographic key management and audit data for encryption/decryption of the TOE’s database
MagicDBPlus v2.1 Admin v2.1.0.1	Provides establishment and management of encryption/decryption policies through a console, in order to control the TOE
MagicDBPlus v2.1 Agent v2.1.0.1	Conducts encryption/decryption of user DB data with a plug-in installed to the database server that has DB under the protection of the TOE

DEKs (Data Encryption Key) used to encrypt/decrypt user data of the TOE are encrypted into KEKs (Key Encryption Key) for protection. In addition, DEKs are also used for protection of communications between TSF data stored and TOE components and utilize a validated cryptographic module whose security and implementation conformity have been verified through the Korea Cryptographic Module Validation Program (KCMVP)

1.3.3 TOE Usage and Major Security Features

Along with an Agent installed in the database server that has the DB to be protected, the TOE encrypts user data received from the application server before storing them into the DB and decrypts the user data encrypted which have been transmitted from the database server to the application server in accordance with policies defined by the authorized administrator who performs encryption/decryption of user data as prescribed by the scope of encryption objects required for an organization’s security policy via the Management Server, using the Administrative Tool. The TOE executable codes offer integrity.

In order for the authorized administrator to operate the TOE under an organization’s operational environment in a security policy, it provides security audit function that records and manages audit data regarding major auditable events; management of cryptographic keys for user and TSF data encryption; cryptographic support such as cryptographic operation; user data protection that encrypts user data and protect incidental information; verification of the authorized administrator’s identity and authentication failure handling; identification and authentication such as TOE internal mutual authentication; definition of security functions and roles; security management functions for configuration; protection of TSF data transmitted among TOE components; protection of TSF data stored in repositories controlled by the TSF; TSF protection function such as TSF self-testing; and provision of TOE access to manage access session(s) of the authorized administrator; and the Agent performs encryption/decryption of user data in accordance with the security policy defined.

1.3.4 Non-TOE and TOE Operational Environment

The TOE is software that provides a function of preventing unauthorized disclosure of information that intends to be protected with encryption of the DB. All hardware and operating system (OS) where the TOE is installed and DBMS are regarded as non-TOE. As an external IT entity, the mail server (SMTP , SMTP Server) is used to notify an alarm to an administrator with respect to threats that occur in operation.

Hardware/software required for the TOE to be installed is listed in [Table 1.2].

[Table 1-2] Installation Hardware/Software installed for TOE

TOE	Category	Item	Minimum Specifications
Agent	H/W	CPU	Intel Core i5 CPU 2.20 GHz or higher
		Memory	4 GB or higher
		HDD	500 MB or more of space required for TOE installation
		NIC	Ethernet 100/1000 Mbps * 1 Port or more
	S/W	OS	Rocky 9.6 (Linux Kernel 5.14.0) 64 bit
		DBMS to be	DB2 11.5.8.0 (64 bit)

		protected	
	H/W	CPU	Intel Core i5 CPU 2.20 GHz or higher
		Memory	4 GB or higher
		HDD	500 MB or more of space required for TOE installation
		NIC	Ethernet 100/1000 Mbps * 1 Port or more
	S/W	OS	Rocky 9.6 (Linux Kernel 5.14.0) 64 bit
		DBMS to be protected	Oracle 19.3.0.0.0 (64 bit)
Management Server	H/W	CPU	Intel Core i5 CPU 2.20 GHz or higher
		Memory	4 GB or higher
		HDD	100 GB or more of space required for TOE installation
		NIC	Ethernet 100/1000 Mbps * 1 Port or more
	S/W	OS	Rocky 9.6 (Linux Kernel 5.14.0) 64 bit
Administrative Tool	H/W	CPU	Intel Core i5 CPU 2.50 GHz or higher
		Memory	4 GB or higher
		HDD	500 MB or more of space required for TOE installation
		NIC	Ethernet 100/1000 Mbps * 1 Port or more
	S/W	OS	Windows 11 Pro 64 bit

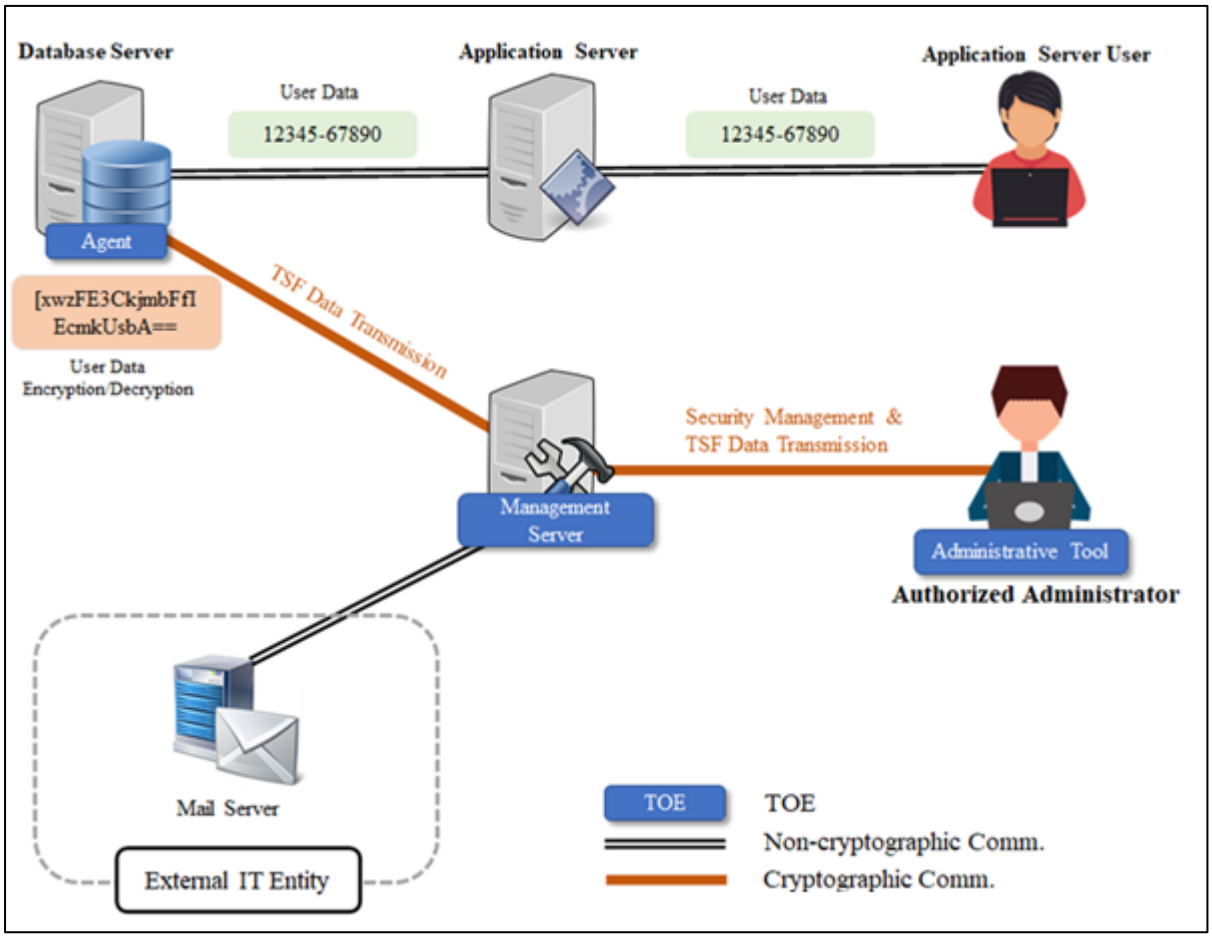
Furthermore, the software required for the TOE is as shown below.

- The following is the external IT entity required for the TOE.

External IT Entity	Description
Mail Server	To be used to notify or send an alarm (a warning mail) to an administrator regarding threats that arise during operation of the Management Server

1.3.5 TOE Operational Environment

As shown in Figure 1-1, the operational environment is comprised of MagicDBPlus v2.1 Agent (“Agent”), MagicDBPlus v2.1 Server (“Management Server”) and MagicDBPlus v2.1 Admin (“Administrative Tool”).
Installed as a plug-in to the DBMS which has to be protected, the Agent receives TSF data from the Management Server and performs encryption/decryption of user data upon the request from the application server. In addition, the authorized administrator manages the scope and policies of encryption in accordance with security policy required in the organization via the Management Server, using the Administrative Tool. Upon the request of application service users, the application server makes a request to the database server while the Agent encrypting/decrypting user data, if necessary, and deliver them to application service users.
Moreover, if a critical event (e.g., reaching to audit data threshold, etc.) arises in the Management Server, a mail is sent to a user designated by the authorized administrator via a mail server.



[Figure 1-1] TOE Operational Environment

Communication between TOE components is performed using TLS v1.3 encrypted communication.

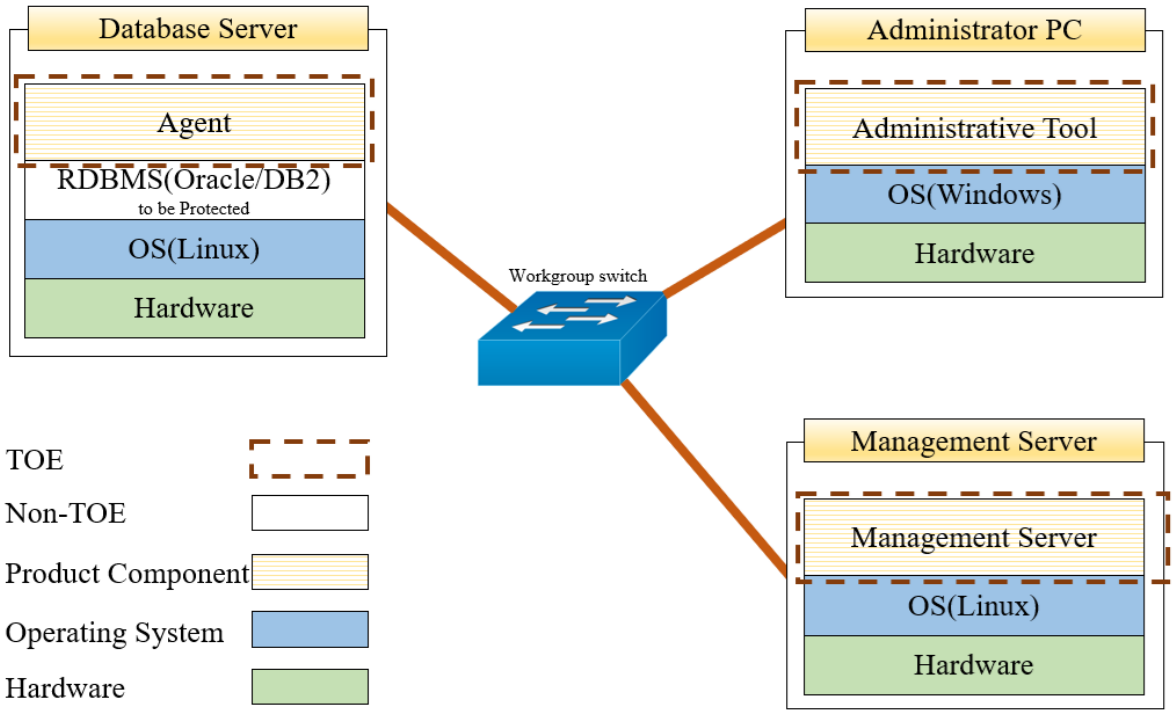
1.4 TOE Description

This Chapter describes the physical and logical scopes of the TOE.

1.4.1 Physical Scope of the TOE

The TOE consists of the Management Server, the Administrative Tool, the Agent, an operational guidance (MagicDBPlus v2.1 Operational Guidance v1.1, MagicDBPlus_ v2.1_OPE_v1.1.pdf) and an installation guide (MagicDBPlus v2.1 Installation Guide v1.1, MagicDBPlus_v2.1_PRE_v1.1.pdf).

The Management Server is MagicDBPlus v2.1 Server v2.1.0.1 (MagicDBPlus_v2.1_Server_v2.1.0.1) that performs a role in login processing, policy establishment, audit data storage and generation and management of encryption keys.



[Figure 1-2] Physical Scope of the TOE

The Administrative Tool refers to MagicDBPlus v2.1 Admin v2.1.0.1 (MagicDBPlus_v2.0_Admin_v2.1.0.1) which is software offering the security management function to the administrator whereas the Agent refers to MagicDBPlus v2.1 Agent v2.1.0.1 (MagicDBPlus_v2.0_Agent _v2.1.0.1) which is software that provides a function of encrypting/decrypting user data. Hardware and OS where the TOE is installed are not included in the scope of the TOE.

The following are types and delivery methods as per TOE component.

Category		Type	Delivery
TOE	MagicDBPlus v2.1		
Version	v2.1.0.1		
Developer	Dreamsecurity Co.,Ltd, Dreams Security, Future Technology Lab, Cryptographic Key Application Technology Team		
TOE Component	MagicDBPlus v2.1 Server v2.1.0.1 : MagicDBPlus_v2.1_Server_v2.1.0.1.sh	S/W	Included in an installation CD of the product package provided to users
	MagicDBPlus v2.1 Admin v2.1.0.1 : MagicDBPlus_v2.1_Admin_v2.1.0.1.exe	S/W	
	MagicDBPlus v2.1 Agent v2.1.0.1 : MagicDBPlus_v2.1_Agent_v2.1.0.1.sh	S/W	
Guidance Document	MagicDBPlus v2.1 Installation Guide v1.1 : MagicDBPlus_v2.1 PRE_v1.1.pdf		
	MagicDBPlus v2.1 Operational Guidance v1.1 : MagicDBPlus_v2.1OPE_v1.1pdf		

Third-party software included in TOE components:

TOE Component	3rd Party SW	Description
Management Server	SQLite v3.53.2	Used as file DB storage for TSF data (DEK for user data, CSPs, TOE configuration, audit data, etc.) on management server
	OpenSSL 3.6.3	Used for TLS v1.3 encrypted communication
Management Tool	OpenSSL 3.6.3	Used for TLS v1.3 encrypted communication
Agent	OpenSSL 3.6.3	Used for TLS v1.3 encrypted communication

The following validated cryptographic module (KCMVP), whose security and implementation conformance have been verified, is used.

-Module Name: MagicCrypto V2.3.0
-Validation Number: CM-263-2030.1
-Validation Date: 2025-01-24
-Expiry Date: 2030-01-24
-Developer: DreamSecurity Co., Ltd.

1.4.2 Logical Scope of the TOE

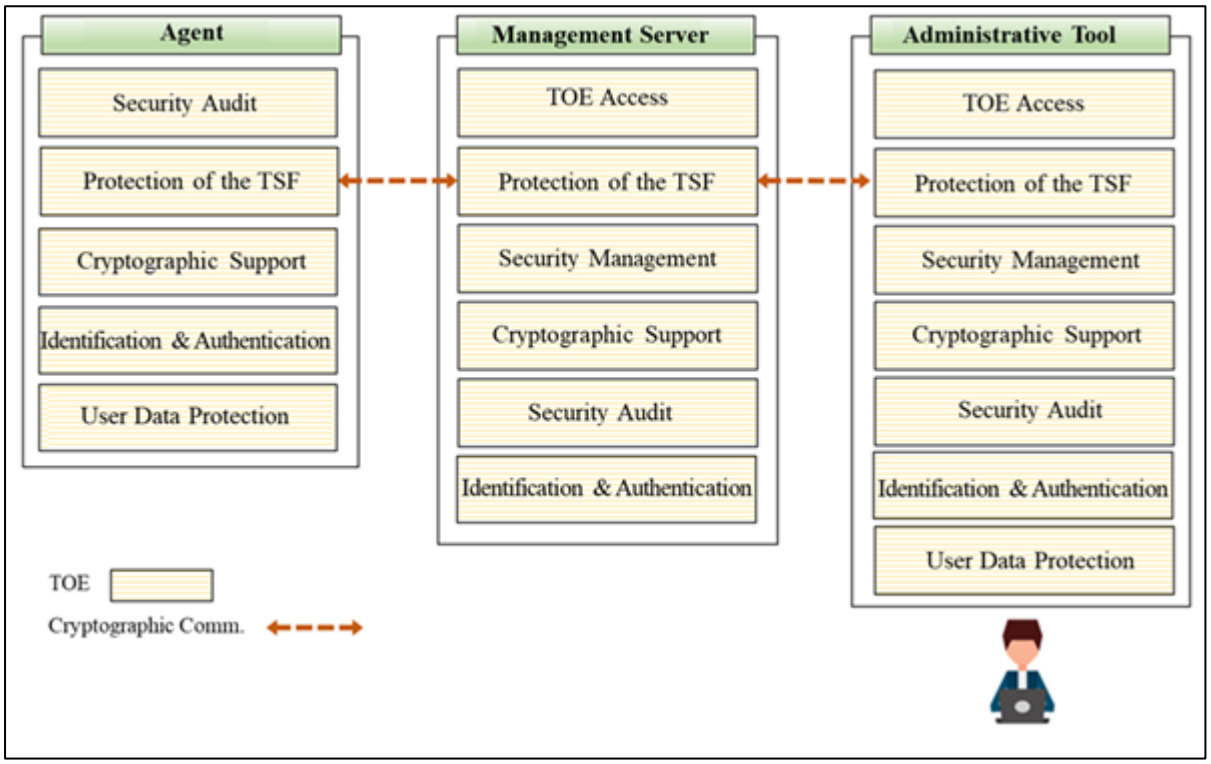


Figure 1-3 Logical Scope of the TOE

Security Audit

The TOE records and manages audit data including event date/time, event type, subject's ID (identity) and privileges, event result, and event detail in the management server. The generated audit data can be queried by the administrator through the Administrative tool MagicDBPlus v2.1 Admin, and there is only one authorized administrator who can perform security management and view audit records.

Security audit records can be selectively queried in descending order by event occurrence date/time based on event date, access IP, access ID, event type, and event result.

The results of self-tests performed by each TOE component are stored and managed in the management server. If a self-test fails, an alert is sent to the email address set by the authorized administrator.

When the space for storing audit data exceeds the capacity designated by the administrator, a warning message is sent to the administrator's email. When the audit data storage is full, old audited event data is overwritten and a warning message is sent to the administrator's email.

All audit data is encrypted and stored in a local file DB on the management server, and access is managed to allow only authorized management server processes.

Cryptographic Support

The validated cryptographic module (KCMVP) MagicCrypto V2.3.0 used by the TOE supports random number generator algorithms and constructs entropy input with noise sources satisfying entropy of 128 bits or more that have passed the entropy test (TTAK.KO-12.0341/R1 randomness test). It performs health tests (RCT, APT) on

collected entropy noise sources when the RNG entropy is first collected or reseeded.
After passing the noise source health test, input/output data uses hexadecimal binary data by default to input/output messages and hash values. When input is received through the console, it is converted to hexadecimal, and additional seed values (QRNG output, etc.) from users are input using algorithm parameters.
The TOE uses the validated cryptographic module (KCMVP) MagicCrypto V2.3.0 to perform cryptographic key generation, distribution, destruction, operations, and random number generation. The cryptographic module is also used for cryptographic key generation and exchange during encrypted communication between physically separated TOE components.
The TOE generates TSF data encryption keys and user data encryption keys using the random number generator (HASH_DRBG 256) of the validated cryptographic module. It encrypts user data of the target DBMS using symmetric key encryption algorithms (ARIA-CBC 128/192/256-bit, SEED-CBC 128-bit, LEA-CBC 128/192/256-bit) and hash algorithms (SHA-256/384/512).
TSF data is protected using symmetric key encryption algorithm (ARIA-CBC 256-bit), digital signature algorithm (RSA-PSS 2048-bit), hash algorithm (SHA-256), and MAC algorithm (HMAC-SHA 256-bit).
Distribution of encryption keys between TOE components is performed securely using public key encryption (RSAES 2048-bit), and encryption keys are overwritten three times with '0x00'.
The KEK is derived using the PBKDF2(SHA-256) algorithm of the validated cryptographic module (KCMVP) with the password entered at startup, Salt (16 bytes) stored in the configuration file, and Iteration Count of 1024. The entered password is destroyed by overwriting it with zeros three times.

[Table 1-3] Key Derivation Algorithm Standard List

Component	Cryptographic Operation	Key Derivation Algorithm	Key Length (bits)	Reference Standard
Management Server / Management Tool / Agent	Password-based derivation for generating KEK to encrypt TSF data encryption key (DEK)	PBKDF2(SHA-256)	256	TTAS.KO-12.0334

User Data Protection
The TOE uses the validated cryptographic module (KCMVP) MagicCrypto V2.3.0 through the user data encryption/decryption policy set by the authorized administrator to perform encryption/decryption when storing and modifying user data in the target DB.
The TOE operates as a plug-in and supports column-level user data encryption/decryption in the agent.
The TOE generates different encryption result values each time encryption is performed on the same user data.
When encryption/decryption is complete, the TOE initializes so that the previous original user data value cannot be recovered.

Identification and Authentication
The TOE provides identification and authentication functions for administrators who perform security management functions, and the ID and password must be changed upon first login after product installation. To protect authentication feedback during administrator identification and authentication data input, '*' is displayed as a protection indicator when entering a password.
Additionally, feedback on the reason for failure is not provided in case of authentication failure, and the account is locked (for 5 minutes) after consecutive authentication failures (5 times).
The TOE blocks authentication information reuse attempts for administrators logging into the TOE.
The TOE provides the following criteria for password verification:

- Password length: minimum 9 to maximum 16 characters
- Available password characters: digits (0-9), uppercase (English), lowercase (English), special characters (~, `, !, @, #, $, %, ^, &, *, (,), -, _, +, =)
- Password must include at least one character from each valid character type
- Passwords identical to the user account (ID) shall be prohibited.
- Consecutive repetition of identical characters or digits shall be prohibited.
- Sequential input of consecutive characters or digits on the keyboard shall be prohibited.

- Reuse of any password that has been used within the previous three (3) months shall be prohibited.

The TOE shall perform mutual authentication between its components through a self-implemented protocol.The TOE shall perform mutual authentication between its components through a self-implemented protocol. Communication between TOE components shall be encrypted using TLS v1.3.

Security Management
The TOE has only one authorized administrator account, and the ID and password must be changed upon first login.
The TOE provides security management functions including cryptographic key generation and deletion, policy registration and deletion, mail notification configuration, audit threshold configuration, and user data encryption/decryption. The authorized administrator performs security management through the management tool's security management interface.

Protection of the TSF
TOE components perform encrypted communication using TLS v1.3. TSF data is protected using a TSF Data Encryption Key (DEK) in storage controlled by the TSF. The Key Encryption Key (KEK), which encrypts the TSF Data Encryption Key (DEK), is derived through a Password-Based Key Derivation Function (PBKDF2), and the TSF Data Encryption Key (DEK) is encrypted and stored using this derived key. In addition, the TOE protects TSF data transmitted between TOE components and performs checks on major security function processes through TSF self-tests. The TOE performs self-tests on major processes, critical files, and cryptographic modules at TOE startup and periodically during operation (every 3 hours after startup). For critical files, an authorized administrator may also manually initiate integrity testing through the security management interface. If the result of a self-test is abnormal, an alert email is sent to the administrator.

When RCT/APT test errors occur for the health test on collected entropy noise sources of the validated cryptographic module, or when integrity is compromised, the management tool, agent, and management server can be manually reinstalled to return to a safe state.

TOE Access
The management access session available for performing security management functions is limited to 1. If an administrator session already logged into the management server exists, no further authorized administrator access is permitted.
If no activity is detected for a certain period (default: 10 minutes) after the authorized administrator logs into the management server through the management tool, the session connected to the management server is terminated. Additionally, the authorized administrator IP addresses are limited to 2. During initial management server installation, one accessible administrator IP address is pre-specified during the installation process.

1.5 Terms and Definitions

Private Key
A cryptographic key used with an asymmetric cryptographic algorithm, uniquely bound to a single entity (the subject using the private key); must not be disclosed.

Object
A passive entity within the TOE that is the target of a subject's operations and contains or receives information.

Health Test
Implemented within a random bit generator to perform real-time monitoring of the noise source.

A health test is not a process for verifying statistical properties of the noise source, but rather a method to detect cases where the collected noise source does not operate normally due to equipment aging or similar causes.

※ Refer to the health test defined in Section 5.2 of TTAK.KO-12.0306/R1 for details.

Deterministic Random Bit Generator (DRBG)
An algorithm that generates a bit string from an initial value called a seed; produces the same bit string when

given the same seed input.

Approved Mode of Operation
The operational mode of a cryptographic module using an approved cryptographic algorithm.

Approved Cryptographic Algorithm
A cryptographic algorithm selected by the cryptographic module validation authority with consideration for security, reliability, and interoperability, covering block ciphers, hash functions, message authentication codes, random bit generators, key distribution, public-key cryptography, and digital signature algorithms.

Attack Potential
The extent of effort required to exploit a vulnerability in the TOE.
NOTE 1: Effort is expressed as a function of attacker-related attributes (e.g., expertise, resources, and motivation) and vulnerability-related attributes (e.g., opportunity, time of exposure).

Public Key
A cryptographic key used with an asymmetric cryptographic algorithm, uniquely bound to a single entity (the subject using the public key); may be made publicly available.

Public Key (Asymmetric) Cryptographic Algorithm
A cryptographic algorithm that uses a public/private key pair.

Management Access
The act of an administrator connecting to the TOE using HTTPS, SSH, TLS, IPSec, or similar protocols for the purpose of TOE administration.

Recommend / be recommended
"Recommend" or "be recommended" as used in application notes indicates requirements that are not mandatory for the TOE but are advised for its safe operation.

Random Bit Generator (RBG)
A device or algorithm that outputs a statistically independent and unbiased sequence of binary digits. RBGs used for cryptographic applications generally produce a bit string of 0s and 1s that can be combined into random blocks. RBGs are classified as deterministic or non-deterministic. A deterministic RBG generates a bit string from an initial value called a seed key; a non-deterministic RBG produces output that depends on an unpredictable physical source.

※ A cryptographic RBG consists of an entropy source (used for seed construction) and a Deterministic Random Bit Generator (DRBG).

Symmetric Cryptographic Technique
A cryptographic technique that uses the same secret key for both encryption and decryption; also referred to as secret key cryptography.

Local Access
A connection established between an administrator and the TOE through a console port.

Database (DB)
A collection of data organized according to a defined structure to receive, store, and supply data in response to the needs of multiple users simultaneously supporting multiple application workloads. In this Protection Profile, the database relevant to column-level encryption refers to a relational database.

Data Encryption Key (DEK)
A key used to encrypt and decrypt data.

Iteration
The use of the same component to express two or more different requirements.

Security Function Policy (SFP)
A set of rules describing specific security behaviors performed by the TSF (TOE Security Functionality) that can be expressed as SFRs (Security Functional Requirements).

Security Target (ST)
An implementation-dependent specification of security requirements for a specific TOE, grounded in the security problem definition.

Security Attribute
A property of subjects, users (including external IT products), objects, information, sessions, and/or resources used to define SFRs; these values are used to enforce SFRs.

Security Token
A hardware device in which key generation, digital signature generation, and related operations are processed internally, designed for the secure storage and protection of sensitive information.

Protection Profile (PP)
An implementation-independent statement of security requirements for a TOE type.

Decryption
The process of using a decryption key to restore ciphertext to its original plaintext.

Secret Key
A cryptographic key used with a secret key cryptographic algorithm, uniquely bound to one or more entities; must not be disclosed.

User
A technical system of persons, or one of their components, that interacts with the TOE from outside the TOE boundary. Within the TOE, users include authorized administrators and authorized general users.

※ User types related to SFRs are classified as human users and external IT entities. Human users are further divided into local human users (interacting directly with the TOE via TOE equipment) and remote human users (interacting with the TOE indirectly through other IT products).

User Data
Data for users that does not affect the TSF (TOE Security Functionality).

Seed
A secret value used for random bit generator initialization.

Selection
Specifying one or more items from a list described in a component.

Manual Recovery
Recovery performed by a user or through an update server with user intervention.

Identity
A unique representation identifying an authorized user; may be the user's real name, abbreviated name, or pseudonym.

Encryption
The conversion of plaintext to ciphertext using an encryption key.

Korea Cryptographic Module Validation Program (KCMVP)
A program that validates the security and implementation conformance of cryptographic modules used to protect important, non-classified information transmitted over national and public institution information networks.

Agent Type 1

Includes products such as anti-virus software, software-based secure USB, and host data loss prevention (DLP) products.

Endpoints hosting the agent are generally Windows® OS PCs accessible by organizational employees. If the agent is compromised, data on the user host may be damaged or leaked; this product type requires strict application of security requirements covering confidentiality, integrity, and availability.

Agent Type 2
Includes products such as network access control (NAC) and patch management systems.

Endpoints hosting the agent are generally Windows® OS PCs accessible by organizational employees. While agent compromise is unlikely to result in direct data damage or leakage, it may prevent users from normally accessing organizational resources; this product type requires application of security requirements covering confidentiality and integrity.

Agent Type 3
Includes products such as database access control, OS (server) access control, and integrated security management products.

Endpoints hosting the agent are generally in physically secure environments accessible only to authorized personnel, resulting in a relatively lower threat probability for this product type.

Endpoint
A point at which TOE components such as agents or clients are installed and operated, with no further subordinate interconnected entities.

Entropy
A measure used to assess the unpredictability of data; a numerical representation of the information content of data. Represents disorder or randomness — the higher the randomness, the higher the entropy.

Entropy Rate
The entropy of a dataset divided by its size; expressed as a value between 0 and 1.

Entropy Source
A function or device combining a noise source, health test, and conditioning algorithm.

Element
A self-contained statement of a security requirement assigned to a SAR or SFR.

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

Operation (on a CC component)
Modifying or iterating a component. Permitted operations on a component are: assignment, iteration, refinement, and selection.

Operation (on an object)
A specific type of action performed by a subject on an object.

External Entity
A technical system of persons, or one of their components, that interacts with the TOE from outside the TOE boundary.

Threat Agent
An entity with the potential to execute malicious actions against assets protected by the TOE.
Authorized Administrator
An authorized user who safely operates and manages the TOE.

Authorized User
An entity authorized to perform operations on the TOE in accordance with the SFRs.

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

Automated Recovery
A recovery action performed without user intervention.

Assets
Entities to which the owner of the TOE assigns value.

Noise Source
A function or device that generates non-deterministic data.

Refinement
The addition of detail to a security component in its specification.

Organizational Security Policies
A set of security rules, procedures, practices, and guidelines imposed or expected to be imposed on an operational environment by a real or hypothetical organization.

Dependency
A relationship between components where, if a PP, ST, functional package, or assurance package includes a given component, it must also include the other component on which it depends, or provide a rationale for its absence.

Subject
An active entity within the TOE that performs operations on objects.

Augmentation
The addition of one or more requirements to a package.
NOTE 1: For functional packages, augmentation is considered only within a single package, not across other packages, PPs, or STs. NOTE 2: For assurance packages, augmentation applies to one or more SARs.

Conditioning
The process of removing bias from a collected noise source to increase the per-bit entropy rate.

Column
A set of data values of a specific data type corresponding to a single value per row in a relational database table.

Component
A set of elements; the smallest selectable unit that can be used to form the basis of requirements.

Client Type
Includes products such as virtual private network (VPN) and wireless LAN authentication products.

A client is an entity installed on the user's host that requests communication with a server on behalf of the user.

Class
A grouping of Common Criteria families sharing the same security objective.

Key Encryption Key (KEK)
A key used to encrypt and decrypt other cryptographic keys.

Target of Evaluation
A set of software, firmware, and/or hardware, possibly accompanied by guidance documentation, that is the subject of an evaluation.

Evaluation Assurance Level
A well-defined package of assurance requirements representing a predefined assurance level.
NOTE 1: EAL is defined in Part 5 of the CC.

Family
A grouping of components sharing a similar purpose but differing in emphasis or rigor.

Assignment
Specifying a parameter identified within a functional or assurance component.

Can / Could
"Can" or "could" as used in application notes indicates requirements that may be applied to the TOE at the discretion of the Security Target author.

Shall / Must
"Shall" or "must" as used in application notes indicates requirements that must be mandatorily applied to the TOE.

Critical Security Parameters
Security-related information whose disclosure or modification could compromise the security of a cryptographic module (e.g., secret/private keys, authentication data such as passwords or PINs).

Application Server
In this Protection Profile, an Application Server refers to a server on which an application developed by the TOE-operating organization is installed and operated to provide specific application services. The application reads user data from the DB on the Database Server at the request of application service users, or transmits user data to be stored in the DB to the Database Server.

Database Server
In this Protection Profile, a Database Server refers to a server on which a DBMS is built and operated by the TOE-operating organization to manage the protected database.

DBMS (Database Management System)
A software system configured to build and utilize a database. In this Protection Profile, the DBMS relevant to column-level encryption refers to a database management system based on the relational database model.

SSL (Secure Sockets Layer)
A security protocol proposed by Netscape to provide security properties such as confidentiality and integrity over computer networks.

TOE Security Functionality
The combined set of all hardware, software, and firmware of the TOE upon which the correct enforcement of the SFRs depends.

TSF Data
Data created by and for the TOE that can affect the operation of the TOE.

TLS
An encrypted, authenticated communication protocol between servers and clients, based on SSL and described in RFC 2246.

1.6 Notational Conventions

This Security Target uses some abbreviations and English terms for clarity. The notation, format, and writing conventions follow the Common Criteria.

The Common Criteria allows the following operations to be performed on security functional requirements: iteration, assignment, selection, and refinement.

Iteration
Used when one component is used multiple times with different applications. Results are indicated with (iteration number) in parentheses after the component identifier.

Assignment
Used to assign a specific value to an unspecified parameter. Results are indicated in brackets [assignment value].

Selection
Used to select one or more items from a list. Results are indicated in underlined italics.

Refinement
Used to add detail to a requirement. Results are indicated in bold text.

2 Conformance Claim

2.1 CC, PP and Security Requirements Package Conformance

The Common Criteria (CC) and the Protection Profile (PP) and the Security Requirements Package to which this ST and the TOE conform are as follows:

Category		Conformance
Common Criteria		Information Security System Common Criteria CC:2022 R1 - Part 1: Introduction and General Model, CC:2022 R1 (CCMB-2022-11-001) - Part 2: Security Functional Components, CC:2022 R1 (CCMB-2022-11-002) - Part 3: Security Assurance Components, CC:2022 R1 (CCMB-2022-11-003) - Part 4: Framework for Evaluation Methodology, CC:2022 R1 (CCMB-2022-11-004) - Part 5: Pre-defined Security Requirement Packages, CC:2022 R1 (CCMB-2022-11-005) - Errata and Interpretation for CC:2022 (R1) and CEM:2022 (R1), v1.2, CCMB-2025-001, 2025.10
Protection Profile		National Database Encryption Protection Profile V3.1
Type of Conformance	Part 2 Security Functional Components	Extended: FDP_UDE.1, FIA_IMA.1, FMT_PWD.1, FPT_PST.1
	Part 3 Security Assurance Components	Conformance
	Package	Addition: EAL1 added (ATE_FUN.1)

2.2 Declaration of Protection Profile Conformance

The Protection Profile conformed to by this Security Target is 'National Database Encryption Protection Profile V3.1'. There is no other Protection Profile being composed.
This Security Target contains only a PP-conformance claim, and since it is based on CC:2022, only conformance claims for direct-rationale PP exist.

2.3 Package Conformance Declaration

The assurance requirement package conformed to by this Security Target is EAL1, with some additional assurance requirements defined.
- Assurance Package: EAL1 augmented (ATE_FUN.1)

2.4 Rationale for Conformance Declaration

Since this Security Target accepts the same TOE type, security problem definition, security objectives, and security requirements as the Protection Profile, the conformance claim to the 'National Database Encryption Protection Profile V3.1' is 'Strict PP-conformance'.

2.5 Method of Protection Profile Conformance

The package conformed to by this Security Target requires the use of the evaluation methods/evaluation activities defined in Section 6.2 Assurance Requirements.

3 Definition of Security Problems

The definition of security problems specifies threats, security policies, and assumptions that the TOE and its operational environment are intended to address.

3.1 Assets

The primary assets protected by database encryption are as follows:
Databases managed by the DBMS in the organization's operational environment
- The TOE itself and critical data related to TOE operation (e.g., TSF data)

3.2 Threats

Threat agents are IT entities and human users who may cause harm to protected assets by unauthorized access or abnormal means. A variety of threats may arise, and threat agents pertaining to the TOE possess a basic level of expertise, resources, and motivation.

3.2.1 Unauthorized Access

T.SESSION_HIJACK
A threat agent may gain access to the user's privileges by accessing a screen left unattended by a logged-in user or by exploiting a user session that remains active after logout.

T.RETRY_AUTH_ATTEMPT
A threat agent may successfully access the TOE as an authorized user by repeatedly attempting authentication and using acquired information to impersonate a legitimate user.

T.IMPERSONATION
A threat agent may attempt to access the TOE by impersonating an authorized user or TOE component.

T.REPLAY
A threat agent may gain access to the TOE by capturing and reusing authentication information.

T.WEAK_PASSWORD
A threat agent may acquire weakly managed passwords (such as default values) to access the TOE as an authorized user; if only basic password policies are applied, the agent may impersonate an authorized user and gain access to the TOE.

3.2.2 Information Disclosure

T.UNAUTHORIZED_INFO_LEAK
A threat agent may leak sensitive information stored in the database through unauthorized means.

T.STORED_DATA_LEAKAGE
A threat agent may cause unauthorized disclosure of sensitive data stored within the TOE or by external entities interacting with the TOE (e.g., cryptographic keys, TOE configuration data).

T.TRANSMISSION_DATA_DAMAGE
A threat agent may expose or modify data transmitted between TOE components and external IT entities in an unauthorized manner.

T.WEAK_CRYPTO_PROTOCOLS
A threat agent may analyze traffic using weak cryptographic protocols or low-strength cryptography to deduce cryptographic keys or obtain the contents of encrypted communications.

3.2.3 Compromise of TOE Functions

T.TSF_COMPROMISE
A threat agent may compromise the TSF through unauthorized access, leading to malfunction or disabling of TOE

functions.

3.3 Organizational Security Policies

T.TSF_COMPROMISE
A threat agent may compromise the TSF through unauthorized access, leading to malfunction or disabling of TOE functions.

P.AUDIT
Security-related events must be logged and retained to ensure accountability. The recorded data should be reviewed. The available disk space for storing audit data must be regularly checked to prevent audit data loss, and stored audit data must be protected against unauthorized modification and deletion.

P.SECURE_OPERATION
The administrator must be provided with management capabilities to securely configure the TOE in accordance with organizational security policy and to operate it correctly following the TOE operation manual.

P.CRYPTO_STRENGTH
Cryptographic measures must be applied to important data storage and transmission (such as passwords for user authentication), and secure cryptographic algorithms must be used.

3.4 Assumptions

The following conditions are assumed to exist in the operational environment adopting this Protection Profile.

A.PHYSICAL_CONTROL
Physical access to the location where the TOE is installed and operated must be controlled and restricted to authorized administrators, and protection facilities must be in place.

A.TRUSTED_ADMIN
Authorized administrators of the TOE are assumed to be non-malicious, properly trained in TOE management functions, and to perform their duties correctly in accordance with administrator guidelines.

A.SECURE_DEVELOPMENT
Developers integrating cryptographic functions using the TOE in applications or DBMS must comply with the requirements specified in the documentation provided with the TOE to ensure proper application of security functions.

A.OPERATION_SYSTEM_REINFORCEMENT
The operating system on which the TOE is installed and operated must be reinforced for the latest vulnerabilities to ensure reliability and safety of the operating system.

4 Security Objectives

The following security objectives for the operational environment are to be addressed by technical and procedural measures in the operational environment, to ensure that the TOE can provide its security functionality correctly.

4.1 Security Objectives for Operational Environment

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.

OE.LOG_BACKUP
Authorized administrators of the TOE shall periodically check the available space in the audit data repository to prevent loss of audit records and perform audit log backups (such as to an external log server or separate storage device) to ensure audit data is not exhausted.

OE.PHYSICAL_CONTROL
Physical access to the location where the TOE is installed and operated shall be controlled and restricted to authorized administrators, and protection facilities must be in place.

OE.TRUSTED_ADMIN
Authorized administrators of the TOE shall be non-malicious, properly trained in TOE management functions, and perform their duties correctly in accordance with administrator guidelines.

OE.SECURE_DEVELOPMENT
Developers integrating cryptographic functions using the TOE in applications or DBMS shall comply with the requirements specified in the documentation provided with the TOE to ensure secure application of the TOE's security features.

OE.OPERATION_SYSTEM_REINFORCEMENT
Authorized administrators of the TOE shall reinforce the operating system on which the TOE is installed and operated against the latest vulnerabilities in order to ensure the reliability and safety of the operating system.

OE.TRUSTED_TIMESTAMP
The TOE must use reliable timestamps provided by the operational environment (NTP or OS time) to accurately record security-related events.

4.2 Rationale for Security Objectives

4.2.1 Rationale for Security Objectives for the Operational Environment

	OE. LOG_BACKUP OE	OE. PHYSICAL_ CONTROL	OE TRUSTED_ADMIN	OE. SECURE_DEVELOPMENT	OE. OPERATION_ SYSTEM_RE-INFORCEMENT	OE.TRUSTED_ TIMESTAMP
P. AUDIT	O					O
P.SECURE_OPERATION			O			
A.PHYSICAL_ CONTROL		O				
A.TRUSTED_ ADMIN	O		O			
A. SECURE_DEVELOPMENT				O		
A. OPERATION_ SYSTEM_RE-INFORCEMENT					O	
A.TRUSTED_TIMESTAMP						O

P.AUDIT – OE.LOG_BACKUP, OE.TRUSTEDTIMESTAMP

P.AUDIT is fulfilled by OE.LogBackup and OE.TrustedTimestamp.

OE.TRUSTEDTIMESTAMP
The components of the TOE shall generate audit records using trusted time information. The trusted time information shall be obtained from NTP or the time information provided by the operating system.

OE.LOGBACKUP
In addition to the functions of the TOE, the administrator shall perform periodic checks on the audit data storage capacity. To prevent loss of log records, regular log backups or transmission of logs to an external log server shall be performed.

P.SECURE_OPERATION – OE.TRUSTED_ADMIN
P.SECURE_OPERATION is enforced by OE.TRUSTED_ADMIN. OE.TRUSTED_ADMIN ensures that administrators operate the TOE correctly in accordance with the organization's security policy and operational manual.

A.PHYSICAL_CONTROL – OE.PHYSICAL_CONTROL
A.PHYSICAL_CONTROL is supported by OE.PHYSICAL_CONTROL. OE.PHYSICAL_CONTROL ensures that the management server is placed in a facility with proper protection measures, and access is controlled so that only authorized administrators can enter.

A.TRUSTED_ADMIN – OE.TRUSTED_ADMIN, OE.LOG_BACKUP
A.TRUSTED_ADMIN is supported by OE.TRUSTED_ADMIN and OE.LOG_BACKUP.
OE.TRUSTED_ADMIN ensures that administrators are non-malicious, properly trained in TOE management functions, and perform their duties correctly in accordance with the administrator's guidelines.
OE.LOG_BACKUP ensures that authorized administrators periodically check available space in the audit data repository and back up audit logs (to an external log server or separate storage device) to prevent loss of audit records.

A.SECURE_DEVELOPMENT – OE.SECURE_DEVELOPMENT
A.SECURE_DEVELOPMENT is supported by OE.SECURE_DEVELOPMENT.
OE.SECURE_DEVELOPMENT ensures that developers integrating cryptographic functions using the TOE in applications or DBMS comply with the documentation provided with the TOE to ensure secure application of the TOE's security features.

A.OPERATION_SYSTEM_REINFORCEMENT – OE.OPERATION_SYSTEM_REINFORCEMENT
A.OPERATION_SYSTEM_REINFORCEMENT is supported by OE. OPERATION SYSTEM REINFORCEMENT.

OE.OPERATION_SYSTEM_REINFORCEMENT ensures that authorized administrators reinforce the operating system on which the TOE is installed and operated against the latest vulnerabilities, thereby guaranteeing reliability and safety of the operating system.

A. TRUSTED_TIMESTAMP - OE.TRUSTED_TIMESTAMP
A.Trusted_Timestamp is addressed by OE.Trusted_Timestamp. OE.Trusted_Timestamp ensures that timestamps and time information generated by the TOE are provided from a trusted source such as an NTP server or operating system, thereby maintaining the accuracy of time information in audit records and other time-sensitive data.

5 Extended Components Definition

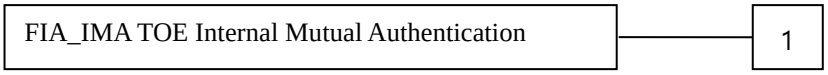
5.1 Identification and Authentication (FIA)

5.1.1 TOE Internal Mutual Authentication

Family Overview

This family (FIA_IMA, TOE Internal Mutual Authentication) requires to provide the function of TOE internal mutual authentication during the user identification and authentication process.

Component Leveling



FIA_IMA.1 TOE Internal Mutual Authentication requires to provide the function of TOE internal mutual authentication in the process of user identification and authentication.

Management: FIA_IMA.1
There is no expected management requirement.

Audit: FIA_IMA.1

The following actions are recommended to document as audit records if FAU_GEN Security Audit Data Generation family is included in the PP/ST.

a) Minimum : Success/failure of mutual authentication
b) Minimum : Modification of authentication protocol

5.1.1.1 FIA_IMA.1 TOE Internal Mutual Authentication

Hierarchical to	No other components
Dependencies	No dependencies

FIA_IMA.1 The TSF shall perform mutual authentication between [assignment: *parts of TOE*] using the [assignment: *authentication protocol*] that meets the following [assignment: *list of standards*].

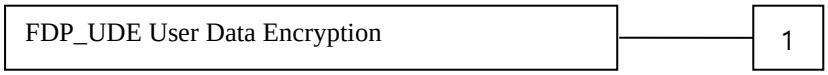
5.2 User Data Protection (FDP)

5.2.1 User Data Encryption

Family Overview

This family (FDP_UDE User Data Encryption) provides the requirements to ensure confidentiality of user data.

Component Leveling



FDP_UDE1 User Data Encryption requires to ensure confidentiality of user data.

Management: FDP_UDE.1

The following management function can be considered in the FMT.
a) Management of rules for encrypting/decrypting user data

Audit: FDP_UDE.1

The following action is recommended to document as audit records if FAU_GEN Security Audit Data Generation family is included in the PP/ST.

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

5.2.1.1 FDP_UDE.1 User Data Encryption

Hierarchical to	No other components
Dependencies	FCS_COP.1 Cryptographic Operation

FDP_UDE.1.1 The TSF shall provide users with the function of encrypting/decrypting user data in accordance with [assignment: *list of encryption/decryption methods*] specified.

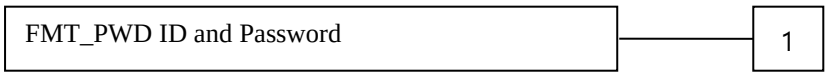
5.3 Security Management (FMT)

5.3.1 ID and Password

Family Overview

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

Component Leveling



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

Management: FMT_PWD.1

The following actions could be considered for the management functions in FMT:
a) Management of ID and password configuration rules.

Audit: FMT_PWD.1

The following actions are recommended to record if FAU_GEN Security audit data generation is included
in the PP/ST:
a) Minimal: All changes of the password.

5.3.1.1 FMT_PWD.1 Management of ID and Password

Hierarchical to	No other components
Dependencies	FMT_SMF.1 Specification of Management Functions FMT_SMR.1 Security Roles
FMT_PWD.1.1	The TSF shall restrict its capability to manage passwords of [assignment: *list of functions*] to [assignment: *authorized roles*] as follows: 1. [assignment: *password combination rules and/or length*]; 2. [assignment: *other management such as management of special characters unusable for a password*].
FMT_PWD.1.2	The TSF shall restrict its capability to manage IDs of [assignment: *list of functions*] to [assignment: *authorized roles*] as follows*:* 1. [assignment: *ID combination rules and/or length*]; 2. [assignment : *other management such as management of special characters unusable for an ID*].
FMT_PWD.1.3	The TSF shall provide the function of [selection: *choose one of the following: setting up an ID and password during the installation process, setting up a password during installation, changing the ID and password during an authorized administrator's initial access, and changing the password during an authorized administrator's initial access*].

5.4 Production of the TSF (FPT)

5.4.1 Protection of TSF Data Stored

Family Overview

This family (FPT_PST, Protection of TSF Data Stored) defines rules to protect against unauthorized modification or disclosure of TSF data stored in repositories controlled by the TSF.

Component Leveling

FPT_PST.1 Basic Protection of TSF Data Stored requires the protection of TSF data stored in repositories controlled by the TSF.

Management: FPT_PST.1
There is no expected management requirement.

Audit: FPT_PST.1
There is no expected audit requirement.

5.4.1.1 FPT_PST.1 Basic Protection of TSF Data Stored

Hierarchical to	No other components
Dependencies	No dependencies

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

6 Security Requirements

This Chapter describes security functional requirements and assurance requirements that shall be met in the TOE.

6.1 Security Functional Requirements

Security functional requirements defined in this ST are specified from relevant security functional components selected from CC Part 2, in order to satisfy the security objectives identified in Chapter 3. The following [Table 6-1] summarizes security functional components used in this ST.

[Table 6-1] Security Functional Requirements

Security Functional Class	Security Functional Component	
Security Audit (FAU)	FAU_ARP.1	Security alarms
	FAU_GEN.1	Audit data generation
	FAU_SAA.1	Potential violation analysis
	FAU_SAR.1	Audit review
	FAU_SAR.3	Selectable audit review
	FAU_STG.1	Audit data storage location
Cryptographic Support (FCS)	FCS_CKM.1(1)	Cryptographic key generation (user data encryption)
	FCS_CKM.1(2)	Cryptographic key generation (TSF data encryption)
	FCS_CKM.6	Cryptographic key destruction timing and events
	FCS_COP.1(1)	Cryptographic operation (user data encryption)
	FCS_COP.1(2)	Cryptographic operation (TSF data encryption)
	FCS_RBG.1	Random bit generation (RBG)
User Data Protection (FDP)	FDP_UDE.1	User data encryption (extended)
	FDP_RIP.1	Subset residual information protection
Identification & Authentication (FIA)	FIA_AFL.1	Authentication failure handling
	FIA_IMA.1	Mutual authentication between TOE components (extended)
	FIA_SOS.1	Verification of secrets
	FIA_UAU.1	Authentication
	FIA_UAU.4	Single-use authentication mechanisms
	FIA_UAU.7	Protected authentication feedback
	FIA_UID.1	User identification
Security Management (FMT)	FMT_MOF.1	Management of security functions behaviour
	FMT_MTD.1	Management of TSF data
	FMT_PWD.1	ID and password management (extended)
	FMT_SMF.1	Specification of management functions
	FMT_SMR.1	Security roles
TSF Protection (FPT)	FPT_FLS.1	Failure with preservation of secure state
	FPT_ITT.1	Basic internal TSF data transfer protection
	FPT_PST.1	Protection of stored TSF data (extended)
	FPT_TST.1	TSF testing
TOE Access (FTA)	FTA_MCS.2	Limitation on multiple concurrent sessions
	FTA_TSE.1(1)	TOE session establishment

6.1.1 Security Audit (FAU)

6.1.1.1 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 action (see [Table 6-2] Actions against Security Violations] if any potential security violation is detected.

[Table 6-2] Actions against Security Violations

Security Functional Component	Security Violation	Action

FIA_AFL.1	- In case an administrator's authentication attempts fail consecutively for a defined number of times (default: 5 times)	- Inactivate the authentication function for a defined period (default: 5 mins.) - Send a warning message to the authorized administrator via email
FPT_TST.1	- In case the integrity verification fails - In case a self-test of the validated cryptographic module(KCMVP) fails	- Send a warning message to the authorized administrator via email
FAU_STG.3	- In case audit trail exceeds the threshold (default: 90%)	- Send a warning massage to the authorized administrator via email
FAU_STG.4	- In case audit trail is full	- Overwrite obsolete and audited event data - Send a warning massage to the authorized administrator via email

6.1.1.2 FAU_GEN.1 Audit Data Generation

Hierarchical to : No other components
Dependencies : FPT_STM.1 Reliable Timestamp

FAU_GEN.1.1 The TSF shall be able to generate audit records of the following auditable events:
a) start-up and shut-down of audit functions;
b) all auditable events with *not specified* level of audit;
c) ["Auditable Event" in [Table 6-3], None].

FAU_GEN.1.2 The TSF shall document at least the following information within each audit record:
a) Date and time of an event, a type of an event, subject identity (if applicable) and results (success or failure) of an event;
b) ["Additional Audit Record" in [Table 6-3], None] based on auditable event definition of functional components included in the PP/ST for each audit event type

[Table 6-3] Auditable Events

Security Functional Component	Auditable Event	Additional Audit Record
FAU_ARP.1	Security path	
FAU_SAA.1	Potential violation analysis	
FAU_STG.4	Actions in case of audit data loss	
FAU_STG.5	Prevention of audit data loss	
FCS_CKM.1(1)	Cryptographic key generation (user data encryption)	
FCS_CKM.1(2)	Cryptographic key generation (TSF data encryption)	
FCS_CKM.6	Cryptographic key destruction timing and events	
FCS_COP.1(1)	Cryptographic operation (user data encryption)	
FCS_COP.1(2)	Cryptographic operation (TSF data encryption)	
FDP_UDE.1 (extended)	User data encryption (extended)	Policy name, encryption/decryption algorithm
FIA_AFL.1	Authentication failure handling	
FIA_IMA.1	Mutual authentication between TOE components (extended)	Certificate Subject DN
FIA_UAU.1	Authentication	
FIA_UAU.4	Single-use authentication mechanism	
FIA_UID.1	Identification	
FMT_MOF.1	Security function management	
FMT_MTD.1	TSF data management	Modified TSF data value
FMT_PWD.1	ID and password management (extended)	
FPT_TST.1	TSF self-testing	Modified TSF data or executable code on integrity violation
FTA_MCS.2	Limitation on multiple concurrent sessions	
FTA_SSL.1	TSF-initiated session locking	
FTA_SSL.3	TSF-initiated session termination	
FAU_ARP.1	Action taken due to potential security violations	
FAU_SAA.1	Enabling and disabling of an analysis mechanism Automated responses performed by the tool	
FAU_STG.3	Action taken due to exceeding of a threshold	

Security Functional Component	Auditable Event	Additional Audit Record
FAU_STG.4	Action taken due to the audit storage failure	
FCS_CKM.1(1)	Cryptographic key generation (User data encryption)	
FCS_CKM.1(2)	Generation of TSF Data Encryption Key	
FCS_CKM.2	Success and failure of an action (only applicable to key distribution related to user data encryption/decryption)	
FCS_CKM.4	Success and failure of an action (only applicable to key destruction related to user data encryption/decryption)	
FCS_COP.1(1)	Cryptographic Operation (User Data Encryption)	
FCS_COP.112)	Cryptographic Operation (TSF Data Encryption)	
FDP_UDE.1	Success and failure of user data encryption/decryption	policy name, encryption/decryption algorithm
FIA_AFL.1	Reaching to the threshold for the unsuccessful authentication attempts and action taken, If appropriate, subsequent, restoration to the normal state	
FIA_IMA.1	Success and failure of mutual authentication Modification of authentication protocol	DN of certificate subject
FIA_UAU.1	All use of authentication mechanism	
FIA_UAU.4	Re-use attempt of authentication data	
FIA_UID.1	All use of user identification mechanism including user identity provided	
FMT_MOF.1	All modifications to the TSF functions	
FMT_MTD.1	All modifications to TSF data values	Modified values of TSF data
FMT_PWD.1	All modifications to passwords	
FPT_TST.1	Execution of the TSF self-tests and the result of the tests	Modified TSF data or executable codes in case of integrity violation
FTA_MCS.2	Denial of a new session based on the limitation of concurrent sessions	
FTA_SSL.1	TSF-initiated session locking	
FTA_SSL.3	TSF-initiated termination	

6.1.1.3 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 while monitoring audited events; and, based on these rules, point out a potential violation during enforcement of the SFRs.

FAU_SAA.1.2 The TSF shall enforce the following rules for monitoring audited events:
a) accumulation or combination of [authentication failure audit event among auditable events in FIA_UAU.1, integrity violation audit event and failure event of the validated cryptographic module's self-test among auditable events in FPT_TST.1, [excess of threshold among auditable events in FAU_STG.3 and full audit trail event among auditable events in FAU_STG.4]] known as a potential security violation;
b) [None].

6.1.1.4 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 capacity of reading [all the audit data] from audit records.

FAU_SAR.1.2 The TSF shall provide audit records in a way that is appropriate for the **authorized administrator** to interpret the information.

6.1.1.5 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 capacity to apply [sorting in a descending order based on the time/date of events] of audit data based on [the “time/date of an event” AND “subject IP” AND “subject ID” and “an event type” AND “an event result”].

6.1.1.6 FAU_STG.1 Protection of Audit Trail Storage

Hierarchical to : No other components
Dependencies : FAU_GEN.1 Audit Data Generation

FAU_STG.1.1 The TSF shall be capable of storing generated audit records in [*a local DBMS*].

6.1.2 Cryptographic Support (FCS)

6.1.2.1 FCS_CKM.1(1) Cryptographic Key Generation (User Data Encryption)

Hierarchical to : No other components
Dependencies: [FCS_CKM.2 Cryptographic Key Distribution or
FCS_CKM.5 Cryptographic Key Derivation or
FCS_COP.1 Cryptographic Operation]
FCS_CKM.3 Cryptographic Key Access
[FCS_RBG.1 Random Bit Generation or
FCS_RNG.1 Random Number Generation]
FCS_CKM.6 Cryptographic Key Destruction Timing and Events

FCS_CKM.1.1 The TSF shall generate **Data Encryption Keys (DEK)** in accordance with the specified cryptographic key generation algorithm [[Table 6-4] Cryptographic Algorithm] and the specified cryptographic key sizes [[Table 6-4] Cryptographic Key Size], conforming to [[Table 6-4] Referenced Standards].

[Table 6-4] List of Algorithm Standards

Type	Crypto Algorithm	Key Length (bits)	Reference Standard
ARIA	HASH_DRBG(SHA-256)	128/192/256	KS X 1213-1
SEED	HASH_DRBG(SHA-256)	128	TTAS.KO-12.0004/R1
LEA	HASH_DRBG(SHA-256)	128/192/256	TTAK.KO-12.0223

6.1.2.2 FCS_CKM.1(2) Cryptographic Key Generation (TSF Data Encryption)

Hierarchical to : No other components
Dependencies : [FCS_CKM.2 Cryptographic Key Distribution or
FCS_CKM.5 Cryptographic Key Derivation or
FCS_COP.1 Cryptographic Operation]
FCS_CKM.3 Cryptographic Key Access
[FCS_RBG.1 Random Bit Generation or
FCS_RNG.1 Random Number Generation]
FCS_CKM.6 Cryptographic Key Destruction Timing and Events

FCS_CKM.1.1 FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with the specified cryptographic key generation algorithm [[Table 6-5] Cryptographic Algorithm] and the specified cryptographic key sizes [[Table 6-5] Cryptographic Key Size], conforming to [[Table 6-5] Referenced Standards].

[Table 6-5] List of Random Bit Generation Standards

Component	Type	Crypto Algorithm	Key Length (bits)	Reference Standard
Management Server	KEK	PBKDF2(SHA-256)	256	KS X 1213-1
	DEK	HASH_DRBG(SHA-256)	128/192/256	ISO/IEC 18031

	Signature Key Pair	RSA-PSS(SHA-256)	2048	ISO/IEC 14888-2
	Integrity Verification Key	HMAC-SHA256	256	ISO/IEC 9797-2
Agent	KEK	PBKDF2(SHA-256)	256	KS X 1213-1
	DEK	HASH_DRBG(SHA-256)	128/192/256	ISO/IEC 18031
	Signature Key Pair	RSA-PSS(SHA-256)	2048	ISO/IEC 14888-2
	Integrity Verification Key	HMAC-SHA256	256	ISO/IEC 9797-2
Administrative Tool	KEK	PBKDF2(SHA-256)	256	KS X 1213-1
	DEK	HASH_DRBG(SHA-256)	128/192/256	ISO/IEC 18031
	Signature Key Pair	RSA-PSS(SHA-256)	2048	ISO/IEC 14888-2
	Integrity Verification Key	HMAC-SHA256	256	ISO/IEC 9797-2

6.1.2.3 **FCS_CKM.6 Cryptographic Key Destruction Timing and Events**

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 or
FCS_CKM.5 Cryptographic Key Derivation]

FCS_CKM.6.1 The TSF shall destroy [[Table 6-6] Cryptographic Key Destruction List] *when no longer needed.*

FCS_CKM.6.2 The TSF shall destroy the cryptographic keys and key material specified in FCS_CKM.6.1 in accordance with the specified cryptographic key destruction method [[Table 6-6] Method], conforming to [None].

[Table 6-6] List of Cryptographic Key Destruction

Type	Method	Destruction Timing	Remarks
KEK Initial Password	Overwrite 3 times with 0x00	Immediately after use	
KEK (Key Encryption Key)	Overwrite 3 times with 0x00	Immediately after use	
DEK (Data Encryption Key)	Overwrite 3 times with 0x00	Immediately after use	
Signature Key	Overwrite 3 times with 0x00	Immediately after use	
Signature Verification Key	Overwrite 3 times with 0x00	Immediately after use	
Integrity Verification Key	Overwrite 3 times with 0x00	Immediately after use	

6.1.2.4 **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 [Cryptographic Operation List in [Table 6-7]] in accordance with a specified cryptographic algorithm [Cryptographic Algorithm in [Table 6-7]] and a specified cryptographic key length [Cryptographic Key Length in [Table 6-7]] that meets the following [[Table 6-7] List of Algorithm Standards].

[Table 6-7] List of Algorithm Standards

Cryptographic Operation List	Cryptographic Algorithm	Application Mode	Cryptographic Key Length (Bit)	Reference Standard
Block Cipher (Symmetric Key Cryptography) User data Encryption/ Decryption	ARIA	CBC	128/192/256	KS X 1213-1
	SEED	CBC	128	TTAS.KO-12.0004/R1
	LEA	CBC	128/192/256	TTAK.KO-12.0223
Hash Function (Uni-directional Cryptography)	SHA-256			ISO/IEC 10118-3
	SHA-384			ISO/IEC 10118-3

User Data Encryption	SHA-512			ISO/IEC 10118-3

6.1.2.5 **FCS_COP.1(2) Cryptographic Operation (TSF Data Encryption)**

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 [Cryptographic Operation List in [Table 6-8]] in accordance with a specified cryptographic key algorithm [Cryptographic Algorithm in [Table 6-8]] and a specified cryptographic key length [Cryptographic Key Length in [Table 6-8]] that meets the following [[Table 6-8] List of Cryptographic Operation Standards].

[Table 6-8] List of Cryptographic Operation Standards

Component	Cryptographic Operation List	Cryptographic Algorithm	Application Mode	Cryptographic Key Length(Bit)	Reference Standard
Management Server	Block Cipher (Symmetrical key cryptography) Encryption/decryption of user data cryptographic keys	ARIA	CBC	256	KS X 1213-1
	Block Cipher (Symmetrical key cryptography) Encryption/decryption of configuration information	ARIA	CBC	256	KS X 1213-1
	Hash function (Uni-directional cryptography) Encryption of administrator password	SHA-2		256	ISO/IEC 10118-3
	Block Cipher (Symmetrical key cryptography) Encryption/decryption of a session cryptographic key	ARIA	CBC	256	KS X 1213-1
	Block Cipher (Symmetrical key cryptography) Encryption/decryption of audit data	ARIA	CBC	256	KS X 1213-1
	Sign Mutual Authentication, TSF Data	RSA-PSS (SHA-256)		2048	PKCS #1 v2.1
	Sign verify Mutual Authentication, TSF Data	RSA-PSS (SHA-256)		2048	ISO/IEC 14888-2
	Integrity verification (TSF data, Encryption of TSF and User Data cryptographic keys)	HMAC-SHA256		256	ISO/IEC 9797-2
	Password-based derivation to generate a key (KEK) to encrypt the TSF data encryption key (DEK)	PBKDF2(SHA-256)		256	TTAS.KO-12.0334
Agent	Block Cipher (Symmetrical key cryptography) Encryption/decryption of configuration information	ARIA	CBC	256	KS X 1213-1
	Block Cipher (Symmetrical key cryptography) Encryption/decryption of a session cryptographic key	ARIA	CBC	256	KS X 1213-1
	Block Cipher (Symmetrical key cryptography) Encryption/decryption of user data cryptographic keys	ARIA	CBC	256	KS X 1213-1
	Sign	RSA-PSS		2048	ISO/IEC

	Mutual Authentication, TSF Data	(SHA-256)			14888-2
	Sign verify Mutual Authentication, TSF Data	RSA-PSS (SHA-256)		2048	ISO/IEC 14888-2
	Integrity verification (TSF data, Encryption of TSF and User Data cryptographic keys)	HMAC- SHA256		256	ISO/IEC 9797-2
	Password-based derivation to generate a key (KEK) to encrypt the TSF data encryption key (DEK)	PBKDF2(SH A-256)		256	TTAS.KO- 12.0334
Administrativ e Tool	Block Cipher (Symmetrical key cryptography) Encryption/decryption of configuration information	ARIA	CBC	256	KS X 1213-1
	Block Cipher (Symmetrical key cryptography) Encryption/decryption of a session cryptographic key	ARIA	CBC	256	KS X 1213-1
	Sign Mutual Authentication, TSF Data	RSA-PSS (SHA-256)		2048	ISO/IEC 14888-2
	Sign verify Mutual Authentication, TSF Data	RSA-PSS (SHA-256)		2048	ISO/IEC 14888-2
	Integrity verification (TSF data, Encryption of TSF Data cryptographic keys)	HMAC- SHA256		256	ISO/IEC 9797-2
	Password-based derivation to generate a key (KEK) to encrypt the TSF data encryption key (DEK)	PBKDF2(SH A-256)		256	TTAS.KO- 12.0334

6.1.2.6 **FCS_RBG.1 Random Bit Generation (extended)**

Hierarchical to : No other components
Dependencies : No dependencies

FCS_RBG.1.1 The TSF shall generate a random bit using a specified random bit generator that meets the following [[Table 6-11] List of Random Bit Generation Standards].

FCS_RBG.1.1 The TSF shall perform a deterministic random bit generation service using [[Table 6-9] Random Number Generation Algorithm] in accordance with [[Table 6-9] Referenced Standards] after initialization.

FCS_RBG.1.2 The TSF shall use the *TSF entropy source [[Table 6-9] Source Name]* for initialization and seeding.

FCS_RBG.1.3 The TSF shall update the DRBG state by *reseeding* using the *TSF entropy source [[Table 6-9] Source Name]* in accordance with [[Table 6-9] Random Number Generation Standards List] under the following circumstances:

O The following circumstances:

- *When the condition of [reseed count reaching 16,777,216] is met*

[Table 6-9] List of Random Bit Generation Standards

Category	Bits	RNG Algorithm	Source Name	Reference Standard	Remarks
Entropy	128 bits	HASH_DRBG(SHA-256)	/dev/urandom	ISO/IEC 18031	Linux
Entropy	128 bits	HASH_DRBG(SHA-256)	CryptGenRandom	ISO/IEC 18031	Windows

6.1.3 User Data Protection (FDP)

6.1.3.1 **FDP_UDE.1 User Data Encryption**

Hierarchical to : No other components

Dependencies : FCS_COP.1 Cryptographic Operation

FDP_UDE.1.1 The TSF shall provide TOE users with a function that enables encryption/decryption of user data in accordance with [Encryption/decryption method as per column, [none]] specified.

6.1.3.2 **FDP_RIP.1 Protection of Partial Residual Information**

Hierarchical to : No other components
Dependencies : No dependencies

FDP_RIP.1.1 The TSF shall ensure that all previous information of resources are not available in case the *resources are allocated and retrieved* to the following object [user data].

6.1.4 Identification and Authentication

6.1.4.1 **FIA_AFL.1 Authentication Failure Handling**

Hierarchical to : No other components
Dependencies : FIA_UAU.1 Authentication

FIA_AFL.1.1 In case there are *[1 ~ 5]* unsuccessful authentication attempts related to [administrator's authentication attempts], the TSF shall detect it.

FIA_AFL.1.2 When unsuccessful authentication attempts *have reached* to a defined number, the TSF shall [inactivate its identification and authentication function for five to ten minutes (default: fiveminutes)].

6.1.4.2 **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 through [self-implemented authentication protocol] that meets [None] between [the Management Server and Agent, the Management Server and Administrative Tool].

6.1.4.3 **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 verity that secrets meet the following [[Table 6-10] List of Secret Criteria].

[Table 6-10] List of Secret Criteria

Category	Defined Criteria	
Valid Characters	Digits (0-9), uppercase (English), lowercase (English), special characters (~, `, !, @, #, $, %, ^, &, *, (,), -, _, +, =)	Mandatory
Composition Rule	Must include at least one of each valid character type	Mandatory
Valid Length	9 to 16 characters	Mandatory
Prohibited	Password same as user account (ID)	Mandatory
	Consecutive repetition of same character/digit prohibited	Mandatory
	Sequential keyboard characters/digits prohibited	Mandatory

	Reuse of previously used password prohibited (either)	One of two required
	Reuse of password used within 3 months prohibited	

6.1.4.4 **FIA_UAU.1 Authentication**

Hierarchical to : No other components
Dependencies : FIA_UID.1 Identification

FIA_UAU.1.1 The TSF shall allow [set-up of the Management Server IP and Port, exchanges of Nonce values and session key agreement] to be enforced on behalf of an **authorized administrator** before the **authorized administrator** is authenticated.

FIA_UAU.1.2 The TSF shall successfully authenticate an **authorized administrator** prior to the permission of all other actions mediated by the TSF on behalf of the **authorized administrator** besides actions specified in FIA_UAU.1.1.

6.1.4.5 **FIA_UAU.4 Authentication Mechanism of Re-use Prevention**

Hierarchical to : No other components
Dependencies : No dependencies

FIA_UAU.4.1 The TSF shall prevent re-use of authentication data related to [password authentication].

6.1.4.6 **FIA_UAU.7 Protection of Authentication Feedback**

Hierarchical to : No other components
Dependencies : FIA_UAU.1 Authentication

FIA_UAU.7.1 The TSF shall provide users with only the following [characters entered are masked with * when the administrator keys in secret information (password), feedback such as “the administrator has failed to log in” or “login failed for user” without the mention of a reason for an authentication failure] while the authentication is in progress.

6.1.4.7 **FIA_UID.1 Identification**

Hierarchical to : No other components
Dependencies : No dependencies

FIA_UID.1.1 The TSF shall allow [set-up of the Management Server IP and Port] to be enforced on behalf of an **authorized administrator** before the **authorized administrator** is identified.

FIA_UID.1.2 The TSF shall successfully identify an **authorized administrator** prior to the permission of all other actions mediated by the TSF on behalf of the **authorized administrator** besides actions specified in FIA_UID.1.1.

6.1.5 Security Management (FMT)

6.1.5.1 **FMT_MOF.1 Management of Security Functions**

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 capacity of performing ***management actions*** regarding the functions in [[Table 6-11] List of Authorized Administrator Functions] to the [authorized administrator].

[Table 6-11] List of Authorized Administrator Functions

Function	Management Action
Set-up and modification of administrator password	Determine and modify an action

Function	Management Action
Set-up and modification of audit information thresholds	Determine and modify an action
Modification of the number of authentication failures and inactivity time	Determine and modify an action
Set-up and modification of the mail server	Determine and modify an action
Set-up and modification of administrator access IP	Determine and modify an action
Real-time validation of modules	Initiate an action
View of audit information	Determine an action
Set-up and modification of cryptographic keys	Determine and modify an action
Establishment and modification of policies	Determine and modify an action
Column encryption	Determine an action

6.1.5.2 **FMT_MTD.1 TSF Data Management**

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 capacity of ***managing*** [Table 6-12] List of Authorized Administrator's TSF Data] to the [authorized administrator].

[Table 6-12] List of Authorized Administrator's TSF Data

TSF Data List	Management
Audit information	Query
Modification of administrator password	Modification
Set-up of audit thresholds	Query, modification
Set-up of the mail server	Query, modification
Administrator information	Query, modification
Set-up of administrator access IP	Query, modification
Cryptographic keys (user data encryption)	Query
Policy list	Query, modification
Authentication failure information	Query, modification
Column list	Query, modification

6.1.5.3 **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 capacity of managing [None] to [None].
1. [None]
2. [None]

FMT_PWD.1.2 The TSF shall restrict the capacity of managing ID of [None] to [None].
1. [None]
2. [None]

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

[Table 6-13] List of Secret Criteria

Category	Defined Criteria	
Valid Characters	Digits (0-9), uppercase (English), lowercase (English), special characters (~, `, !, @, #, $, %, ^, &, *, (,), -, _, +, =)	Mandatory
Composition Rule	Must include at least one of each valid character type	Mandatory
Valid Length	9 to 16 characters	Mandatory

Prohibited	Password same as user account (ID)	Mandatory
	Consecutive repetition of same character/digit prohibited	Mandatory
	Sequential keyboard characters/digits prohibited	Mandatory
	Reuse of previously used password prohibited (either)	One of two required
	Reuse of password used within 3 months prohibited	

6.1.5.4 **FMT_SMF.1 Specification of Management Functions**

Hierarchical to : No other components
Dependencies : No dependencies

FMT_SMF.1.1 The TSF shall be able to perform the following management functions: [list of management functions provided by the TSF]

[
a) List of security functions specified in FMT_MOF.1
b) List of TSF data management specified in FMT_MTD.1
c) List of functions specified in FMT_PWD.1
]

6.1.5.5 **FMT_SMR.1 Security Roles**

Hierarchical to : No other components
Dependencies : FIA_UID.1 Identification

FMT_SMR.1.1 The TSF shall maintain the [following roles of authorized users]:

[
Set-up and modification of administrator password
Set-up and modification of audit thresholds
Modification of the number of authentication failure and inactivity time
Set-up and modification of the mail server
Set-up and modification of administrator access IP
Real-time validation of modules
View of audit information
Set-up and modification of cryptographic keys
Establishment and modification of policies
Encryption of columns
]

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

6.1.6 Protection of the TSF (FPT)

6.1.6.1 **FTP_FLS.1 Basic Protection of Internally-transmitted TSF Data**

Hierarchical to : No other components
Dependencies : No dependencies

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [[Table 6-14] TSF Failure Type List].

[Table 6-14] TSF Failure Type List

Category	Failure Type
Cryptographic Module	Validated cryptographic module (noise source health test) error

6.1.6.2 **FPT_ITT.1 Basic Protection of Internally-transmitted TSF Data**

Hierarchical to : No other components
Dependencies : No dependencies

FPT_ITT.1.1 The TSF shall protect the TSF data from *disclosure and modification* **through encryption and verification of message integrity** when they are transmitted between discrete parts of the TOE.

6.1.6.3 **FPT_PST.1 Basic Protection of TSF Data Stored (extended)**

Hierarchical to : No other components
Dependencies : No dependencies

FPT_PST.1.1 The TSF shall protect [the following TSF data] stored in repositories controlled by the TSF from unauthorized *disclosure and modification.*

[
Audit information
Modification of administrator password
Set-up of audit thresholds
Set-up of the mail server
Administrator information
Set-up of administrator access IP
Cryptographic keys (User data encryption)
Policy list
Authentication failure information
Column list
]

6.1.6.4 **FPT_TST.1 TSF Self-testing**

Hierarchical to : No other components
Dependencies : No dependencies

FPT_TST.1.1 The *TSF* shall run the following self-tests [[Table 6-15] TSF Self-Test List] at *start-up, periodically during normal operation, and upon request by an authorized* ***administrator***, in order to demonstrate the correct operation of the TSF.

FPT_TST.1.2 The TSF shall provide authorized **administrators** with the capability to verify the integrity of [*[Table 6-15] TSF Data*].

FPT_TST.1.3 The TSF shall provide authorized **administrators** with the capability to verify the integrity of [*[Table 6-15] TSF Data*].

[Table 6-15] List of Secret Criteria

Component	TSF Data	Description
Management Server	MagicDBPolicy.conf	Management server configuration file
	libMagicCrypto.so	Validated cryptographic module
	start.sh / stop.sh / restart.sh	Start/stop/restart scripts

	magicdb.dat	Policy management data file
	magicdb.audit.dat	Audit log data file
	Cryptographic key files	Key files
	Log files	Log files
Agent	MDBAgent.conf	Agent configuration file
	libMagicDB.so	DB encryption module
	libMagicCrypto.so	Validated cryptographic module
	start.sh / stop.sh / restart.sh	Start/stop/restart scripts
	Log files	Log files
	DEK	Data Encryption Key
Administrator Tool	MDBAdmin.xml.enc	Management tool configuration file
	MagicCryptoV23.dll	Validated cryptographic module
	install.ico / uninstall.ico / Uninstall.exe	Installation/uninstallation files

6.1.7 TOE Access (FTA)

6.1.7.1 **FTA_MCS.2 Limitation of Concurrent Session Number per User Attribute**

Hierarchical to : FTA_MCS.1 Basic Limitation of Concurrent Session Number
Dependencies : FIA_UID.1 Identification

FTA_MCS.2.1 The TSF shall restrict the maximum number of concurrent sessions for the same user to 1 according to [rule: maximum concurrent sessions for users with identical privileges is limited to 1]..

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

6.1.7.2 **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 establishment of the administrator's management access session based on [access IP, *None*].

6.1.8 Security Audit (FAU)

6.1.8.1 **FAU_STG.2 Protection of audit data repository**

Hierarchical to : No other components
Dependencies : : FAU_GEN.1 Audit Data Generation

FAU_STG.2.1 The TSF shall protect audit data stored within the audit records from unauthorized deletion.

FAU_STG.2.2 The TSF shall *prevent* unauthorized modification of audit data stored within the audit records.

6.1.8.2 **FAU_STG.4 Action taken due to the audit storage failure**

Hierarchical to : No other components
Dependencies : FAU_STG.2 Protection of audit data repository

FAU_STG.4.1 The TSF shall take [notification to an authorized administrator, [none]] when the audit records exceed [the threshold defined below]
a) Basis for threshold: The percentage (%) of used space out of the total audit record storage capacity; default value is 90%.
b) Permitted range for threshold input: Integer value between 10 and 90.

6.1.8.3 **FAU_STG.5 Prevention of audit data loss**

Hierarchical to : FAU_STG.4 Response actions when audit data loss is anticipated
Dependencies : Protection of audit data repository

FAU_STG.5.1 The TSF shall, when the audit records are full, *perform overwriting of the oldest audit records* and [send an email to an authorized administrator].

6.1.9 Cryptographic Support (FCS)

6.1.9.1 **FCS_CKM.5 Cryptographic Key Derivation**

Hierarchical to : No other components
Dependencies : [FCS_CKM.2 Cryptographic Key Distribution or FCS_COP.1 Cryptographic Operation]
FCS_CKM.6 Timing and Events for Cryptographic Key Destruction

FCS_CKM.5.1 TSF shall derive a cryptographic key [assignment: key type] from [assignment: input parameters] using a specified key derivation algorithm [see Table 6-14: Key Derivation Algorithms] and a specified cryptographic key length [see Table 6-14: Cryptographic Key Lengths] that conform to the standards listed in [Table 6-14: List of Key Derivation Standards].

[표 6-16] 암호키 유도 연산 표준 목록

Component	Cryptographic Operation List	Cryptographic Algorithm	Application Mode	Cryptographic Key Length(Bit)	Reference Standard
Management	Password-based derivation to	PBKDF2(SH		256	TTAS.KO-

Server Agent Administrative Tool	generate a key (KEK) to encrypt the TSF data encryption key (DEK)	A-256)			12.0334

6.1.9.2 **FCS_RBG.3 Random Bit Generation (Internal Seeding – Single Source)**

Hierarchical to : No other components
Dependencies : : FAU_GEN.1 Audit Data Generation

FCS_RBG.3.1 The TSF shall be able to seed the DRBG using the *TSF software-based entropy source* [[Table 6-17] Source Name] with a minimum entropy of at least [[Table 6-17] Number of Bits] bits.

[Table 6-17] List of Random Bit Generation Standards

Category	Bits	RNG Algorithm	Source Name	Reference Standard	Remarks
Entropy	128 bits	HASH_DRBG(SHA-256)	/dev/urandom	ISO/IEC 18031	Linux
Entropy	128 bits	HASH_DRBG(SHA-256)	CryptGenRandom	ISO/IEC 18031	Windows

6.1.10 Protection of the TSF(FPT)

6.1.10.1 **FPT_RCV.1 Manual recovery**

Hierarchical to : No other components
Dependencies : : AGD_OPE.1 User Operation Manual

FPT_RCV.1.1 Following [[Table 6-18] Failure/Service Discontinuity List], the TSF shall provide the capability to return the TOE to a secure state.

[Table 6-18] TSF Failure Type List

Category	Failure Type
Cryptographic Module	Validated cryptographic module (noise source health test) error

6.1.10.2 **FPT_TUD.1 TSF Security Patch Update (Extended)**

Hierarchical to : No other components
Dependencies : : No other components

FPT_TUD.1.1 The TSF shall provide [assignment: authorized roles] with the capability to query the unique identification information of the TOE.

FPT_TUD.1.2 The TSF shall verify the validity of the update file prior to installation by using [selection: verification of a published hash value, verification of a digital signature].

6.1.11 Identification and Authentication (FTA)

6.1.11.1 **FTA_SSL.1 TSF-initiated session locking**

Hierarchical to : No other components
Dependencies : FIA_UAU.1 Authentication

FTA_SSL.1.1 The TSF shall lock interactive sessions after a [user inactivity period of 10 minutes] by the following method.

a) Obscure or overwrite the display device so that the current content cannot be read.

b) Disable all activities regarding user data access and display device, rather than unlocking the session.

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

6.1.11.2 **FTA_SSL.3 TSF-initiated termination**

Hierarchical to : No other components
Dependencies : FMT_SMR.1 Security Roles

FTA_SSL.3.1 The TSF shall terminate interactive sessions after a [user inactivity period of 10 minutes].

6.1.11.3 **FTA_TSE.1(2) TOE Session Establishment**

Hierarchical to : No other components
Dependencies : No other components

FTA_TSE.1.1 The TSF shall be able to deny session establishment based on [connection IP, [none]].

6.1.12 Cryptographic Support (FCS)

Security Functional Class	Security Functional Component	
Cryptographic Support (FCS)	FCS_CKM.2	Cryptographic Key Distribution
Protection of the TSF (FPT)	FPT_STM.1	Reliable time stamps

6.1.12.1 **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 a cryptographic key in accordance with a cryptographic key distribution method specified in [Cryptographic Key Distribution Method in [Table 6-17]] that meets the following [[Table 6-17] List of Cryptographic Key Distribution Standards].

[Table 6-19] List of Cryptographic Key Distribution Standards

Usage	Cryptographic Algorithm	Cryptographic Key Length	Reference Standard
Public Key Cryptography	RSAES (SHA-256)	Public key 2048 bits	ISO/IEC 18033-2
Cryptographic Key Distribution Method			
- Distribution by encrypting an encryption key for information transmitted between the Management Server, Administrative Tool and Agent with the public key of the other server			

6.2 Security Assurance Requirements

Security assurance requirements in this ST are comprised of assurance components in CC Part 3 and the evaluation assurance level is EAL1+. Assurance components are summarized in [Table 6-20].

[Table 6-20] Assurance Requirements

Assurance Class	Assurance Component	
ST Evaluation	ASE_INT.1	ST Introduction
	ASE_CCL.1	Conformance Claim
	ASE_OBJ.1	Security Objectives for 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	Preparation Procedures
Life-cycle Support	ALC_CMC.1	TOE Leveling
	ALC_CMS.1	TOE Configuration Management (CM) Coverage
Tests	ATE_FUN.1	Functional Testing
	ATE_IND.1	Independent testing : conformance
Vulnerability Assessment	AVA_VAN.1	Vulnerability Survey

6.2.1 Security Target Evaluation

6.2.1.1 **ASE_INT.1 ST Introduction**

Dependencies : No dependencies

Developer Action Elements

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

Content and Presentation Elements

ASE_INT.1.1C The ST introduction shall contain a 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 summarize 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 For multi-assurance STs, the TOE overview shall describe the TSF composition with respect to the sub-TSFs defined in the PP-composition declared by the ST.

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

ASE_INT.1.9C 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**.**

6.2.1.2 **ASE_CCL.1 Conformance Claim**

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 to identify the version of the CC to which the ST and the TOE conform.

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

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

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

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

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

ASE_CCL.1.7C The rationale for conformance shall demonstrate that the TOE type identified in the ST is consistent with that specified in the claimed PPs.

ASE_CCL.1.8C The rationale for conformance shall demonstrate that the security problem definition in the ST is consistent with that in the claimed PPs.

ASE_CCL.1.9C The rationale for conformance shall demonstrate that the security objectives in the ST are consistent with those in the claimed PPs.

ASE_CCL.1.10C The rationale for conformance shall demonstrate that the security requirements in the ST are consistent with those in the claimed PPs and packages.

ASE_CCL.1.11C The rationale for conformance shall demonstrate that the security requirements in the ST are consistent with those in the claimed PPs and packages.

ASE_CCL.1.12C The PP or PP-combination conformance claim shall be expressed as strict conformance, exact conformance, demonstrable conformance, or as a list of conformance types.

ASE_CCL.1.13C If the conformance claim identifies a set of evaluation methods and activities derived from CEM work units to be used in the TOE evaluation, this set shall include all such methods and activities included in the claimed packages, PPs, or PP-modules of a PP-combination, and shall not include others.

Evaluator Action Elements

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

6.2.1.3 **ASE_SPD.1 Security Problem Definition**

Dependencies : No dependencies

Developer Action Elements:

ASE_SPD.1.1D The developer shall provide a security problem definition.

Content and Presentation Elements:

ASE_SPD.1.1C The security problem definition shall describe threats.

ASE_SPD.1.2C All threats shall be described in terms of a threat agent, an asset, and an adverse action.

ASE_SPD.1.3C The security problem definition shall describe OSPs.

ASE_SPD.1.4C The security problem definition shall describe assumptions about the operational environment of the TOE.

Evaluator Action Elements:

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

6.2.1.4 **ASE_OBJ.1 Security Objectives for Operational Environment**

Dependencies : No dependencies

Developer Action Elements

ASE_OBJ.1D The developer shall provide a statement of security objectives.

Content and Presentation Elements

ASE_OBJ.1C The statement of security objectives shall describe the security objectives for the operational environment.

ASE_OBJ.1.2C The rationale for the security objectives shall demonstrate that each security objective for the operational environment is traceable to the threat(s) addressed by that objective, the OSP(s) enforced by that objective, and the assumption(s) supported by that objective.

ASE_OBJ.1.3C The rationale for the security objectives shall demonstrate that the security objectives for the operational environment support all of the identified assumptions.

Evaluator Action Elements

ASE_OBJ.1.1E The evaluator shall confirm that the provided information meets all evidential requirements.

6.2.1.5 **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 and methodologies as a model for presentation.

ASE_ECD.1.5C The extended components shall consist of measurable and objective elements such that conformance or non-conformance 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.

6.2.1.6 **ASE_REQ.1 Stated Security Requirements**

Dependencies : ASE_ECD.1 Extended Components Definition

ASE_SPD.1 Security Problem Definition

ASE_OBJ.1 Security Objectives for the Operational Environment

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 rationale for the security requirements.

Content and Presentation Elements:

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

ASE_REQ.1.2C For a single-assurance ST, the statement of security requirements shall define a global set of SARs applicable to the entire TOE. The set of SARs shall be consistent with the PP or PP-Configuration for which the ST claims conformance.

ASE_REQ.1.3C For a multi-assurance ST, the statement of security requirements shall define a global set of SARs applicable to the entire TOE and sets of SARs applicable to sub-TSFs. The sets of SARs

shall be consistent with the multi-assurance PP-Configuration for which the ST claims conformance.

ASE_REQ.1.4C All subjects, objects, operations, security attributes, external entities, and other terms that are used in the SFRs and SARs shall be defined.

ASE_REQ.1.5C The statement of security requirements shall identify all operations on the security requirements.

ASE_REQ.1.6C All operations shall be performed correctly.

ASE_REQ.1.7C Each dependency of the security requirements shall be satisfied, or the security requirements rationale shall justify the non-satisfaction of the dependency.

ASE_REQ.1.8C The security requirements rationale shall trace each SFR to the threats countered by that SFR and the OSPs enforced by that SFR. The security requirements rationale shall demonstrate that the SFRs (together with the security objectives for the operational environment) counter all threats to the TOE.

ASE_REQ.1.9C The security requirements rationale shall demonstrate that the SFRs (together with the security objectives for the operational environment) enforce all OSPs applicable to the TOE.

ASE_REQ.1.10C The security requirements rationale shall explain why the SARs were selected.

ASE_REQ.1.11C The statement of security requirements shall be internally consistent.

ASE_REQ.1.12C If the ST defines a set of SARs that extends the set of SARs of the PP or PP-Configuration for which conformance is claimed, the security requirements rationale shall include an assurance requirements rationale that justifies the consistency of the extension and provides a rationale for the handling of the evaluation methods and evaluation activities identified in the conformance method affected by the extension of the set of SARs.

Evaluator Action Elements:

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

6.2.1.7 **ASE_TSS.1 TOE Summary Specification**

Dependencies : ASE_INT.1 ST Introduction
ASE_REQ.1 Stated Security Requirements
ADV_FSP.1 Basic Functional Specification

Developer Action Elements

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

Content and Presentation Elements

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

Evaluator Action Elements

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

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

6.2.2 Development

6.2.2.1 **ADV_FSP.1 Security-enforcing 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.

6.2.3 Guidance Documents

6.2.3.1 **AGD_OPE.1 Operational User Guidance**

Dependencies : ADV_FSP.1 Basic Functional Specification

Developer Action Elements

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

Content and Presentation Elements

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

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

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

AGD_OPE.1.4C The operational user guidance shall, for each user role, clearly present each type of security-

relevant event relative to the user-accessible functions that need to be performed, including a change in 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.

6.2.3.2 **AGD_PRE.1 Preparative Procedures**

Dependencies : No dependencies

Developer Action Elements

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

Content and Presentation Elements

AGD_PRE.1.1C 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_PRE.1.2C 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.

6.2.4 Life-cycle Support

6.2.4.1 **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 to 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 meets all requirements for content and presentation of evidence.

6.2.4.2 **ALC_CMS.1 TOE CM Coverage**

Dependencies: No dependencies

Developer Action Elements

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

Content and Presentation Elements

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

ALC_CMS.1.2 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.

6.2.5 Tests

6.2.5.1 **ATE_FUN.1 Functional Testing**

Dependencies : ATE_COV.1 Evidence of Test 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 described 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.

6.2.5.2 **ATE_IND.1 Independent Testing : Conformance**

Dependencies : ADV_FSP.1 Basic Functional Specification
AGD_OPE.1 Operational User Guidance
AGD_PRE.1 Preparative Procedures

Developer Action Elements

ATE_IND.1.1D The developer shall provide the TOE to be tested.

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.

6.2.6 Vulnerability Assessment

6.2.6.1 **AVA_VAN.1 Vulnerability Survey**

Dependencies : ADV_FSP.1 Basic Functional Specification
AGD_OPE.1 Operational User Guidance
AGD_PRE.1 Preparative Procedures

Developer Action Elements

AVA_VAN.1.1D The developer shall provide the TOE to be tested.

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 with basic attack potentials.

6.3 Security Requirement Rationale

6.3.1 Rationale for Security Functional Requirements

The rationale for the Security Functional Requirements demonstrates the following:

▷ Each threat and Organizational Security Policy is addressed by at least one Security Functional Requirement.

▷ Each Security Functional Requirement is traced back to at least one threat or Organizational Security Policy.

[Table 6-21] Mapping of Security Problem Definition to SFRs

SFR	T.SESSION_HIJACK	T.RETRY	T.IMPERSONATION	T.REPLAY	T.WEAK_PWD	T.INFO_LEAK	T.DATA_LEAK	T.TRANS_DAMAGE	T.WEAK_CRYPTO	T.TSF_COMPROMISE	P.AUDIT	P.SECURE_OP	P.CRYPTO
FAU_ARP.1										O			
FAU_GEN.1											O		
FAU_SAA.1										O			
FAU_SAR.1											O		
FAU_SAR.3											O		
FAU_STG.1											O		
FAU_STG.2											O		
FAU_STG.4											O		
FAU_STG.5											O		
FCS_CKM.1(1)						O							O
FCS_CKM.1(2)							O	O	O				O
FCS_CKM.2						O	O	O	O				O
FCS_CKM.5						O	O	O	O				O
FCS_CKM.6						O	O	O	O				O
FCS_COP.1(1)						O							O
FCS_COP.1(2)							O	O	O				O
FCS_RBG.1						O	O	O	O				O
FCS_RBG.3						O	O	O	O				O
FDP_UDE.1						O							
FDP_RIP.1						O							
FIA_AFL.1		O	O							O			
FIA_IMA.1			O										
FIA_SOS.1					O								
FIA_UAU.1			O							O			
FIA_UAU.4			O	O						O			
FIA_UAU.7			O		O					O			

FIA_UID.1			O							O			
FMT_MOF.1										O		O	
FMT_MTD.1										O		O	
FMT_PWD.1					O					O		O	
FMT_SMF.1										O		O	
FMT_SMR.1										O		O	
FPT_FLS.1						O	O	O	O				O
FPT_ITT.1								O					
FPT_PST.1							O						
FPT_RCV.1										O			
FPT_TST.1						O	O	O	O	O			O
FTA_MCS.2	O												
FTA_SSL.3	O												
FTA_TSE.1(1)	O												

T.Session_Hijacking — FTA_MCS.2, FTA_SSL.3, FTA_TSE.1(1)

FTA_MCS.2 counters T.Session_Hijacking by limiting concurrent sessions to the TOE using the same user account or equivalent privileges.
FTA_SSL.3 counters T.Session_Hijacking by ensuring termination of interactive sessions after a period of authorized user inactivity.
FTA_TSE.1(1) counters T.Session_Hijacking by ensuring that the establishment of authorized user access sessions is determined based on criteria such as IP address.

T.Consecutive_Authentication_Attempts — FIA_AFL.1

FIA_AFL.1 counters T.Consecutive_Authentication_Attempts by ensuring the capability to define the number of failed authentication attempts for authorized administrators and general users, and to take countermeasures upon reaching the defined threshold.

T.Masquerade — FIA_AFL.1, FIA_IMA.1, FIA_UAU.1, FIA_UAU.4, FIA_UAU.7, FIA_UID.1

FIA_AFL.1 counters T.Masquerade by ensuring the capability to define the number of failed authentication attempts for authorized administrators and to take countermeasures upon reaching the defined threshold.
FIA_IMA.1 counters T.Masquerade by ensuring that mutual authentication is performed between TOE components.
FIA_UAU.1 and FIA_UAU.4 counter T.Masquerade by ensuring the successful authentication of administrators and general users attempting to access the TOE.
FIA_UAU.7 counters T.Masquerade by ensuring that only masked values are displayed or not displayed to the user during authentication, and that no feedback regarding the reason for authentication failure is provided upon failure.
FIA_UID.1 counters T.Masquerade by ensuring the successful identification of administrators and general users attempting to access the TOE.

T.Replay — FIA_UAU.4

FIA_UAU.4 counters T.Replay by ensuring the capability to prevent the reuse of authentication data.

T.Weak_Password — FIA_SOS.1, FIA_UAU.7, FMT_PWD.1

FIA_SOS.1 counters T.Weak_Password by verifying that password complexity rules are satisfied.
FIA_UAU.7 counters T.Weak_Password by ensuring that only masked values are displayed or not displayed to the user during the authentication process.
FMT_PWD.1 counters T.Weak_Password by ensuring the capability to force a password change upon the first login of an authorized administrator using a default password.

T.Unauthorized_Information_Disclosure — FCS_CKM.1(1), FCS_CKM.2, FCS_CKM.5, FCS_CKM.6, FCS_COP.1(1), FCS_RBG.1, FCS_RBG.3, FDP_UDE.1, FDP_RIP.1, FPT_FLS.1, FPT_TST.1

FCS_CKM.1(1), FCS_CKM.2, FCS_CKM.5, FCS_CKM.6, FCS_RBG.1, FCS_RBG.3, FCS_COP.1(1), FPT_FLS.1, and FPT_TST.1 counter T.Unauthorized_Information_Disclosure by ensuring that cryptographic operations are performed in accordance with specified secure algorithms and specified cryptographic key sizes when encrypting and decrypting sensitive user information stored in the database.
FDP_RIP.1 counters T.Unauthorized_Information_Disclosure by ensuring that all original user data is deleted after encryption and decryption of sensitive user information stored in the database.
FDP_UDE.1 counters T.Unauthorized_Information_Disclosure by ensuring that encryption and decryption are performed when an authorized user stores sensitive information in the database.

T.Stored_Data_Disclosure — FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.5, FCS_CKM.6, FCS_COP.1(2), FCS_RBG.1, FCS_RBG.3, FPT_FLS.1, FPT_PST.1, FPT_TST.1

FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.5, FCS_RBG.1, FCS_RBG.3, FPT_FLS.1, and FPT_TST.1 counter T.Stored_Data_Disclosure by ensuring that cryptographic keys are generated and distributed in accordance with secure cryptographic algorithms and key sizes when encrypting stored data.
FCS_CKM.6 counters T.Stored_Data_Disclosure by ensuring that cryptographic keys and their associated information are destroyed in accordance with the specified key destruction method upon completion of stored data encryption.
FCS_COP.1(2) counters T.Stored_Data_Disclosure by ensuring that cryptographic operations are performed in accordance with specified secure algorithms and specified cryptographic key sizes when encrypting stored data.
FPT_PST.1 counters T.Stored_Data_Disclosure by ensuring that TSF data at rest is protected from disclosure threats through means such as encryption and access control.

T.Transmitted_Data_Compromise — FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.5, FCS_CKM.6, FCS_COP.1(2), FCS_RBG.1, FCS_RBG.3, FPT_FLS.1, FPT_ITT.1, FPT_TST.1

FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.5, FCS_RBG.1, FCS_RBG.3, FPT_FLS.1, and FPT_TST.1 counter T.Transmitted_Data_Compromise by ensuring that cryptographic keys are generated and distributed in accordance with secure cryptographic algorithms and key sizes during cryptographic communications.
FCS_CKM.6 counters T.Transmitted_Data_Compromise by ensuring that cryptographic keys and their associated information are destroyed in accordance with the specified key destruction method upon termination of cryptographic communications.
FCS_COP.1(2) counters T.Transmitted_Data_Compromise by ensuring that cryptographic operations are performed in accordance with specified secure algorithms and specified cryptographic key sizes during cryptographic communications.
FPT_ITT.1 counters T.Transmitted_Data_Compromise by ensuring the confidentiality and integrity of data transmitted between TOE components.

T.Weak_Cryptographic_Protocol — FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.5, FCS_CKM.6, FCS_COP.1(2), FCS_RBG.1, FCS_RBG.3, FPT_FLS.1, FPT_TST.1

FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.5, FCS_RBG.1, FCS_RBG.3, FPT_FLS.1, and FPT_TST.1 counter T.Weak_Cryptographic_Protocol by ensuring that cryptographic keys are generated and distributed in accordance with the required cryptographic algorithms and key sizes of standard cryptographic algorithms with a security strength of 112 bits or more when encrypting transmitted data.
FCS_CKM.6 counters T.Weak_Cryptographic_Protocol by ensuring that cryptographic keys and their associated information are destroyed in accordance with the specified destruction method.

FCS_COP.1(2) counters T.Weak_Cryptographic_Protocol by ensuring that cryptographic operations are performed in accordance with standard cryptographic algorithms with a security strength of 112 bits or more and the corresponding cryptographic key sizes when encrypting transmitted data.

T.TSF_Compromise — FAU_ARP.1, FAU_SAA.1, FIA_AFL.1, FIA_UAU.1, FIA_UAU.4, FIA_UAU.7, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_PWD.1, FMT_SMF.1, FMT_SMR.1, FPT_RCV.1, FPT_TST.1

FAU_ARP.1 counters T.TSF_Compromise by ensuring the capability to take countermeasures upon detection of security violations such as TOE integrity compromise.
FAU_SAA.1 counters T.TSF_Compromise by ensuring the capability to examine audited events and indicate security violations such as TOE integrity compromise.
FIA_AFL.1, FIA_UAU.1, FIA_UAU.4, FIA_UAU.7, and FIA_UID.1 counter T.TSF_Compromise by ensuring that access to the TOE is permitted only after successful identification and authentication of the user, thereby blocking unauthorized bypass access by threat agents.
FMT_MOF.1, FMT_MTD.1, FMT_PWD.1, FMT_SMF.1, and FMT_SMR.1 counter T.TSF_Compromise by distinguishing authorized user roles for accessing and configuring management functions into administrators and general users, and providing security policies and security functions according to role, thereby ensuring that unauthorized access by threat agents is blocked.
FPT_RCV.1 counters T.TSF_Compromise by ensuring the provision of a capability for TOE agents or clients to recover from tampered information.
FPT_TST.1 counters T.TSF_Compromise by ensuring TSF self-testing for correct TOE operation, and ensuring the capability for authorized administrators to verify the integrity of TSF data and the TSF.

P.Audit — FAU_GEN.1, FAU_SAR.1, FAU_SAR.3, FAU_STG.1, FAU_STG.2, FAU_STG.4, FAU_STG.5

FAU_GEN.1 satisfies P.Audit by ensuring that audit records are generated for auditable events such as start-up/shutdown of the audit function and success/failure of identification and authentication of administrators.
FAU_SAR.1 satisfies P.Audit by ensuring that authorized administrators are provided with the capability to review audit records, and that audit records are presented in a manner suitable for administrators to interpret the information.
FAU_SAR.3 satisfies P.Audit by providing a selective audit review capability based on logical relationship criteria applied to audit data.
FAU_STG.1 satisfies P.Audit by ensuring that, for TOE servers, audit data is stored in local storage or transmitted in real time to an external IT entity for storage using a secure channel.
FAU_STG.2 satisfies P.Audit by ensuring the capability to protect stored audit data from unauthorized modification and deletion.
FAU_STG.4 satisfies P.Audit by ensuring that appropriate countermeasures are taken when the audit trail of a TOE server exceeds the storage capacity threshold.
FAU_STG.5 satisfies P.Audit by ensuring the capability to take appropriate countermeasures when the audit trail of a TOE server reaches saturation.

P.Secure_Operation — FMT_MOF.1, FMT_MTD.1, FMT_PWD.1, FMT_SMF.1, FMT_SMR.1

FMT_MOF.1 satisfies P.Secure_Operation by ensuring that only authorized users have the capability to manage security functions.
FMT_MTD.1 satisfies P.Secure_Operation by ensuring that only authorized users have the capability to manage TSF data.
FMT_PWD.1 satisfies P.Secure_Operation by ensuring that only authorized administrators have the capability to manage password composition rules and length, and by providing functions such as mandatory password change upon first login by an authorized user.
FMT_SMF.1 satisfies P.Secure_Operation by requiring the TSF to specify management functions for security functions, security attributes, TSF data, and related items.
FMT_SMR.1 satisfies P.Secure_Operation by ensuring that authorized roles related to security management are specified.

P.Cryptographic_Strength — FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.5, FCS_CKM.6, FCS_COP.1(1), FCS_COP.1(2), FCS_RBG.1, FCS_RBG.3, FPT_FLS.1, FPT_TST.1

FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.5, FCS_CKM.6, FCS_RBG.1, FCS_RBG.3, FPT_FLS.1, and FPT_TST.1 satisfy P.Cryptographic_Strength by ensuring that cryptographic keys required for standard cryptographic algorithms with a security strength of 112 bits or more are securely generated and distributed when encrypting data.
FCS_COP.1(1) and FCS_COP.1(2) satisfy P.Cryptographic_Strength by ensuring that cryptographic operations are performed in accordance with standard cryptographic algorithms with a security strength of 112 bits or more and the corresponding cryptographic key sizes when encrypting data.

6.3.2 Rationale for Assurance Requirements

he assurance level for this Security Target has been selected as EAL1+(ATE_FUN.1).
EAL1 is applicable where some confidence in correct operation is required, but the threats to security are not considered serious. EAL1 does not require additional developer effort to prepare evaluation deliverables, provided that the TOE has been developed in accordance with development methodologies commonly applied by the developer. That is, it does not require additional investment of cost or time in preparation for evaluation.
EAL1 provides a basic level of assurance by analyzing the Security Functional Requirements contained in a limited Security Target using functional and interface specifications and guidance documentation to understand security behavior.
This analysis is supported by independent testing of the TSF and a search for potential vulnerabilities in the public domain (including functional testing and penetration testing).
Although EAL1 does not require evidence of testing performed by the developer based on functional specifications, this Security Target has augmented ATE_FUN.1 to enable the developer to independently test whether the TSF has been correctly implemented and whether any defects have occurred, and to document the results thereof.

6.3.3 Dependency of the SFRs

Rationale of the following functional components' dependency is shown in [Table 6-19] below.

[Table 6-22] Dependency Rationale

No.	Functional Component	Dependency	Reference No.
1	FAU_ARP.1	FAU_SAA.1	3
2	FAU_GEN.1	FPT.STM.1	Rationale (1)
3	FAU_SAA.1	FAU_GEN.1	2
4	FAU_SAR.1	FAU_GEN.1	2
5	FAU_SAR.3	FAU_SAR.1	4
6	FAU_STG.1	FAU_GEN.1	2
		FTP_ITC.1	Rationale (2)
7	FAU_STG.2	FAU_GEN.1	2
8	FAU_STG.4	FAU_STG.2	7
9	FAU_STG.5	FAU_STG.2	7
10	FCS_CKM.1(1)	[FCS_CKM.2 or FCS_CKM.5 or FCS_COP.1]	12, 13, 15
		[FCS_RBG.1]	17
		FCS_CKM.6	14
11	FCS_CKM.1(2)	[FCS_CKM.2 or FCS_CKM.5 or FCS_COP.1]	12, 13, 16
		FCS_RBG.1	17
		FCS_CKM.6	14
12	FCS_CKM.2	[FCS_CKM.1 or FCS_CKM.5]	10, 11, 13
13	FCS_CKM.5	[FCS_CKM.2 or FCS_COP.1]	12, 15, 16
		FCS_CKM.6	14
14	FCS_CKM.6	[FCS_CKM.1 or FCS_CKM.5]	10, 11, 13

15	FCS_COP.1(1)	[FCS_CKM.1 or FCS_CKM.5]	10, 13
		FCS_CKM.6	14
16	FCS_COP.1(2)	[FCS_CKM.1 or FCS_CKM.5]	11, 13
		FCS_CKM.6	14
17	FCS_RBG.1	[FCS_RBG.2 or FCS_RBG.3]	18
		FPT_FLS.1	33
		FPT_TST.1	37
18	FCS_RBG.3	FCS_RBG.1	17
19	FDP_UDE.1	FCS_COP.1	15, 16
20	FDP_RIP.1	-	-
21	FIA_AFL.1	FIA_UAU.1	24
22	FIA_IMA.1	-	-
23	FIA_SOS.1	-	-
24	FIA_UAU.1	FIA_UID.1	27
25	FIA_UAU.4	-	-
26	FIA_UAU.7	FIA_UAU.1	24
27	FIA_UID.1	-	-
28	FMT_MOF.1	FMT_SMF.1	31
		FMT_SMR.1	32
29	FMT_MTD.1	FMT_SMF.1	31
		FMT_SMR.1	32
30	FMT_PWD.1	FMT_SMF.1	31
		FMT_SMR.1	32
31	FMT_SMF.1	-	-
32	FMT_SMR.1	FIA_UID.1	27
33	FPT_FLS.1	-	-
34	FPT_ITT.1	-	-
35	FPT_PST.1	-	-
36	FPT_RCV.1	AGD_OPE.1	-
37	FPT_TST.1	-	-
38	FTA_MCS.2	FIA_UID.1	27
39	FTA_SSL.3	FMT_SMR.1	32
40	FTA_TSE.1(1)	-	-

Rationale (1): FAU_GEN.1 ensures the utility of auditing — including post-incident tracing and evidence preservation — by accurately recording the date and time of events in audit records. To this end, the audit data generation function has a dependency on the trusted time stamp function (FPT_STM.1). Time information is provided by the operating system, and this dependency may be defined as an operational environment (OE) requirement, thereby allowing the SFR dependency requirement to be relaxed.
Rationale (2): FAU_STG.1 has a dependency on FTP_ITC.1; however, since audit data is stored in local storage, this dependency has not been included in this Security Target.

6.3.4 Dependency Rationale of Security Assurance Requirements

As the dependency of EAL1 Assurance Package provided in the CC is already satisfied, its rationale is omitted herein.

The augmented assurance requirement, ATE_FUN.1, has a dependency on ATE_COV.1. ATE_FUN.1 has been augmented to ensure that the developer performs a test on test items and documents it in the test transcript in an accurate manner. However, ATE_COV.1 is not included in this ST since the PP considers that it is not necessarily required to include ATE_COV.1 that presents the consistency between test items and TSFI.

7 TOE Summary Specification

This Chapter describes security functionality required by the TOE and the list of security functionality is shown in [Table 7-1].

[Table 7-1] List of Security Functions

Security Functional Class	Security Functional Component	
Security Audit (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	Selective Audit Review
	FAU_STG.1	Audit Data Storage Location
	FAU_STG.2	Audit Data Storage Protection
	FAU_STG.4	Countermeasures upon Anticipated Audit Data Loss
	FAU_STG.5	Audit Data Loss Prevention
Cryptographic Support (FCS)	FCS_CKM.1(1)	Cryptographic Key generation (User Data Encryption)
	FCS_CKM.1(2)	Cryptographic Key Generation (TSF Data Encryption)
	FCS_CKM.2	Cryptographic Key Distribution
	FCS_CKM.5	Cryptographic Key Derivation
	FCS_CKM.6	Cryptographic Key Destruction Timing and Events
	FCS_COP.1(1)	Cryptographic Operation (User Data Encryption)
	FCS_COP.1(2)	Cryptographic Operation (TSF Data Encryption)
	FCS_RGB.1	Random Bit Generation
	FCS_RGB.3	Random Bit Generation (Internal Seeding – Single Source)
User data protection (FDP)	FDP_UDE.1	User Data Encryption
	FDP_RIP.1	Protection of Partial Residual Information
Identification and Authentication (FIA)	FIA_AFL.1	Authentication Failure Handling
	FIA_IMA.1	TOE Internal Mutual Authentication
	FIA_SOS.1	Verification of Secrets
	FIA_UAU.1	Authentication
	FIA_UAU.4	Authentication Mechanism for Re-use Prevention
	FIA_UAU.7	Protection of Authentication Feedback
	FIA_UID.1	Identification
Security Management (FMT)	FMT_MOF.1	Management of Security Functions
	FMT_MTD.1	TSF Data Management
	FMT_PWD.1	Management of ID and Password
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.1	Security Roles
Protection of the TSF (FPT)	FPT_ITT.1	Basic Protection of Internally-transmitted TSF Data
	FPT_PST.1	Basic Protection of TSF Data Stored
	FPT_RCV.1	Manual recovery
	FPT_TST.1	TSF Self-testing
TOE Access (FTA)	FTA_MCS.2	Limitation of Concurrent Session Number per User Attribute
	FTA_SSL.3	TSF-initiated termination
	FTA_TSE.1(1)	TOE Session Establishment

7.1 Security Audit

The TOE records and manages all audit data that can occur during operation in the Management Server such as start/end of an audit function, success/failure of identification and authentication of an administrator, details of the TOE configuration changes and security function execution.

The audit data information is defined in [Table 7-2] Auditable Events.

[Table 7-2] Auditable Events

Security Functional Component	Auditable Event	Additional Audit Record
FAU_ARP.1	Action taken due to potential security violations	
FAU_SAA.1	Enabling and disabling of an analysis mechanism Automated responses performed by the tool	
FAU_STG.4	Countermeasures upon Anticipated Audit Data Loss	
FAU_STG.5	Audit Data Loss Prevention	
FCS_CKM.1(1)	Cryptographic key generation (User data encryption)	
FCS_CKM.1(2)	Cryptographic key generation (TSF data encryption)	
FCS_CKM.6	Cryptographic Key Destruction Timing and Events	
FCS_COP.1(1)	Cryptographic operation (User data encryption)	
FCS_COP.1(2)	Cryptographic operation (TSF data encryption)	
FDP_UDE.1	Success and failure of user data encryption/decryption	policy name, encryption/decryption algorithm
FIA_AFL.1	Reaching to the threshold for the unsuccessful authentication attempts and action taken, If appropriate, subsequent restoration to the normal state	
FIA_IMA.1	Success and failure of mutual authentication Modification of authentication protocol	DN of certificate subject
FIA_UAU.1	All use of authentication mechanism	
FIA_UAU.4	Re-use attempt of authentication data	
FIA_UID.1	All use of user identification mechanism including user identity provided	
FMT_MOF.1	All modifications to the TSF functions	
FMT_MTD.1	All modifications to TSF data values	Modified values of TSF data
FMT_PWD.1	All modifications to passwords	
FPT_TST.1	Execution of the TSF self-tests and the result of the tests	Modified TSF data or executable codes in case of integrity violation
FTA_MCS.2	Per user attribute limitation on multiple concurrent sessions	
FTA_SSL.1	TSF-initiated session locking	
FTA_SSL.3	TSF-initiated termination	

The TOE blocks access from unauthorized administrator for protection of audit trail storage and safeguards audit data stored.

If the storage where the audit data are stored exceeds the threshold defined by the administrator (defined as an integer between 10 and 90, default value: 90, unit: %), a warning message is sent to the authorized administrator via email.

If the audit data storage is full, the obsolete and audited event data are overwritten and a warning message is sent to the authorized administrator via email.

The authorized administrator can review audit data through the Administrative Tool (GUI) and retrieve audit data in accordance with date/time of an event, an event type, and an outcome of an event. Search results of the audit data can be sorted and displayed in a descending order based on the date/time of events. The function of modifying/deleting audit data is not provided.

If the TOE detects any potential security violation, an action against the security violation is conducted as shown in [Table 6-3].

[Table 7-3] Actions against Security Violations

Security Functional Component	Security Violation	Action
FIA_AFL.1	- In case an administrator's authentication attempts fail consecutively for a defined number of times (default: 5 times)	- Inactivate the authentication function for a defined period (default: 5 mins.) - Send a warning message to the authorized administrator via email

FPT_TST.1	- In case the integrity verification fails - In case a self-test of the validated cryptographic module(KCMVP) fails	- Send a warning message to the authorized administrator via email
FAU_STG.4	- In case audit trail exceeds the threshold (default: 90%)	- Send a warning message to the authorized administrator via email
FAU_STG.5	- In case audit trail is full	- Overwrite obsolete and audited event data - Send a warning message to the authorized administrator via email

Relevant SFR : FIA_AFL.1, FPT_TST.1, FAU_STG.1, FAU_ARP.1, FAU_GEN.1, FAU_SAA.1, FAU_SAR.1, FAU_SAR.3, FAU_STG.4, FAU_STG.5

7.2 Cryptographic Support

The validated cryptographic module (KCMVP) [Table 6-4] used by the TOE, MagicCrypto V2.3.0, supports random number generation algorithms and constructs entropy input from a noise source satisfying 128 bits or more of entropy that has passed the entropy test (TTAK.KO-12.0341/R1 Randomness Test). When entropy for the random bit generator is first collected or reseeded, the RCT (Repetition Count Test) and APT (Adaptive Proportion Test) health tests are performed on the collected entropy noise source.

After passing the noise source health tests, input/output data is processed using hexadecimal binary data as the default format for messages and hash values. When input is received via console, it is converted to hexadecimal, and additional seed values provided by the user (e.g., QRNG output) are accepted as input using algorithm parameters.

When performing "User Data Cryptographic Key Generation," the TOE generates random numbers using the random number generation algorithm in [Table 7-6], and generates cryptographic keys for user data using the symmetric key cryptographic algorithm in [Table 7-5] and the cryptographic key size in [Table 7-5].

[Table 7-4] Validated Cryptographic Module

Name of Cryptographic Module	Validation No.	Developer	Date Validated
MagicCrypto V2.3.0	CM-263-2030.1	Dream Security Co., Ltd.	2025-01-24

[Table 7-5] List of Algorithm Standards

Cls.	Symmetrical Key Cryptography Algorithm	Cryptographic Key Length (bit)	Reference Standard
Block cipher (symmetrical key cryptography) User data encryption/decryption	ARIA	128/192/256	KS X 1213-1
	SEED	128	TTAS.KO-12.0004/R1
	LEA	128/192/256	TTAK.KO-12.0223
Block cipher (symmetrical key cryptography) TSF data encryption/decryption	ARIA	128/192/256	KS X 1213-1

[Table 7-6] List of Random Bit Generation Standards

Cls.	Random Bit Generation Algorithm	Reference Standard	Remark
Random Bit Generator	HASH_DRBG(SHA-256)	ISO/IEC 18031	Pseudorandom function

The TOE generates a random bit using a random bit generation algorithm in [Table 7-4] while performing “cryptographic key generation of TSF data” and generates a TSF cryptographic key from TSF data in accordance with ARIA algorithm and a cryptographic bit length (256 bits) under a block cipher method (symmetrical key cryptography).

The TOE performs a cryptographic operation specified in the cryptographic operation list of [Table 7-7] in accordance with a cryptographic algorithm and a cryptographic key length specified in [Table 7-7] while performing “cryptographic operation of user data.”

[Table 7-7] List of Algorithm Standards

Cryptographic Operation List	Cryptographic Algorithm	Application Mode	Cryptographic Key Length (Bit)	Reference Standard
Block Cipher (Symmetric Key Cryptography) User data Encryption/ Decryption	ARIA	CBC	128/192/256	KS X 1213-1
	SEED	CBC	128	TTAS.KO-12.0004/R1
	LEA	CBC	128/192/256	TTAK.KO-12.0223
Hash Function (Uni-directional Cryptography) User Data Encryption	SHA-256			ISO/IEC 10118-3
	SHA-384			ISO/IEC 10118-3
	SHA-512			ISO/IEC 10118-3

The TSF performs a cryptographic operation in the cryptographic operation list of [Table 7-8] in accordance with a cryptographic algorithm and a cryptographic key length depending on a component specified in [Table 7-8] while carrying out “cryptographic operation of TSF data.”

[Table 7-8] List of Cryptographic Operation Standards

Component	Cryptographic Operation List	Cryptographic Algorithm	Application Mode	Cryptographic Key Length(Bit)	Reference Standard
Management Server	Block Cipher (Symmetrical key cryptography) Encryption/decryption of user data cryptographic keys	ARIA	CBC	256	KS X 1213-1
	Block Cipher (Symmetrical key cryptography) Encryption/decryption of configuration information	ARIA	CBC	256	KS X 1213-1
	Hash function (Uni-directional cryptography) Encryption of administrator password	SHA-2		256	ISO/IEC 10118-3
	Block Cipher (Symmetrical key cryptography) Encryption/decryption of a session cryptographic key	ARIA	CBC	256	KS X 1213-1
	Block Cipher (Symmetrical key cryptography) Encryption/decryption of audit data	ARIA	CBC	256	KS X 1213-1
	Sign Mutual Authentication, TSF Data	RSA-PSS (SHA-256)		2048	PKCS #1 v2.1
	Sign verify Mutual Authentication, TSF Data	RSA-PSS (SHA-256)		2048	ISO/IEC 14888-2
	Integrity verification (TSF data, Encryption of TSF and User Data cryptographic keys)	HMAC-SHA256		256	ISO/IEC 9797-2
	Password-based derivation to generate a key (KEK) to encrypt the TSF data encryption key (DEK)	PBKDF2(SHA-256)		256	TTAS.KO-12.0334
Agent	Block Cipher (Symmetrical key cryptography) Encryption/decryption of configuration information	ARIA	CBC	256	KS X 1213-1
	Block Cipher (Symmetrical key cryptography) Encryption/decryption of a session cryptographic key	ARIA	CBC	256	KS X 1213-1

	Block Cipher (Symmetrical key cryptography) Encryption/decryption of user data cryptographic keys	ARIA	CBC	256	KS X 1213-1
	Sign Mutual Authentication, TSF Data	RSA-PSS (SHA-256)		2048	PKCS #1 v2.1
	Sign verify Mutual Authentication, TSF Data	RSA-PSS (SHA-256)		2048	ISO/IEC 14888-2
	Integrity verification (TSF data, Encryption of TSF and User Data cryptographic keys)	HMAC-SHA256		256	ISO/IEC 9797-2
	Password-based derivation to generate a key (KEK) to encrypt the TSF data encryption key (DEK)	PBKDF2(SHA-256)		256	TTAS.KO-12.0334
Administrative Tool	Block Cipher (Symmetrical key cryptography) Encryption/decryption of configuration information	ARIA	CBC	256	KS X 1213-1
	Block Cipher (Symmetrical key cryptography) Encryption/decryption of a session cryptographic key	ARIA	CBC	256	KS X 1213-1
	Sign Mutual Authentication, TSF Data	RSA-PSS (SHA-256)		2048	PKCS #1 v2.1
	Sign verify Mutual Authentication, TSF Data	RSA-PSS (SHA-256)		2048	ISO/IEC 14888-2
	Integrity verification (TSF data, Encryption of TSF Data cryptographic keys)	HMAC-SHA256		256	ISO/IEC 9797-2
	Password-based derivation to generate a key (KEK) to encrypt the TSF data encryption key (DEK)	PBKDF2(SHA-256)		256	TTAS.KO-12.0334

The TOE distributes cryptographic keys according to the reference standard in [Table 7-9] by encrypting cryptographic keys of information transmitted among the Management Server, Administrative Tool and Agent with a public key in accordance with the cryptographic algorithm and cryptographic key length specified in [Table 6-9] while performing “distribution of cryptographic keys.”

[Table 7-9] List of Cryptographic Key Distribution Standards

Usage	Cryptographic Algorithm	Cryptographic Key Length	Reference Standard
Public key cryptography	RSAES (SHA-256)	Public key 2048 bits	ISO/IEC 18033-2

Upon shutdown of the Administrative Tool, the symmetric key memory value used for KEK (Key Encryption Key) purposes is destroyed by overwriting it three times with zeros. Upon session termination with the Management Server, the private key memory value used for mutual authentication and TSF data protection purposes is destroyed by overwriting it three times with zeros. The integrity verification key memory value used for TSF data integrity verification purposes is destroyed by overwriting it three times with zeros. In addition, when an authorized administrator deletes a cryptographic key via the

Administrative Tool, the user data encryption key file is overwritten three times with zeros and the file is subsequently deleted. Upon shutdown of the Management Server, the symmetric key memory value used for KEK (Key Encryption Key) purposes is destroyed by overwriting it three times with zeros. Upon session termination with an Agent or the AdministratorTool, the private key memory value used for mutual authentication and TSF data protection purposes is destroyed by overwriting it three times with zeros. The integrity verification key memory value used for TSF data integrity verification purposes is destroyed by overwriting it three times with zeros.

Upon shutdown of the Agent, the symmetric key memory value used for KEK (Key Encryption Key) purposes is destroyed by overwriting it three times with zeros. Upon completion of user data encryption, the symmetric key memory value used for DEK (Data Encryption Key) purposes is destroyed by overwriting it three times with zeros. Upon session termination with the Management Server, the private key memory value used for mutual authentication and TSF data protection purposes is destroyed by overwriting it three times with zeros. The integrity verification key memory value used for TSF data integrity

verification purposes is destroyed by overwriting it three times with zeros.

Relevant SFR : FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.5, FCS_CKM.6, FCS_COP.1(1), FCS_COP.1(2), FCS_RGB.1, FCS_RGB.3

7.3 Function of User Data Protection

The TOE provides the function of user data encryption/decryption under the encryption/decryption method as per column and does not generate the same cryptogram for the same plain texts.

Using the validated cryptographic module, user data are encrypted depending on the usage such as a block cipher (symmetrical key cryptography) and a hash function (uni-directional cryptography) specified in [Table 7-10] in accordance with a cryptographic algorithm and a cryptographic key length in [Table 7-10] List of Algorithm Standards.

[Table 7-10] List of Algorithm Standards

Usage	Cryptographic Algorithm	Application Mode	Cryptographic Key Length(Bit)	Reference Standard
Block cipher (symmetrical key cryptography) User data encryption/decryption	ARIA	CBC	128/192/256	KS X 1213-1
	SEED	CBC	128	TTAS.KO-12.0004/R1
	LEA	CBC	128/192/256	TTAK.KO-12.0223
Hash function (uni-directional cryptography) User data encryption	SHA-256			ISO/IEC 10118-3
	SHA-384			ISO/IEC 10118-3
	SHA-512			ISO/IEC 10118-3

Using the validated cryptographic module during “TSF data encryption”, the TOE performs encryption of DEKs and configuration files with encryption of cryptographic keys and KEKs in the Agent in accordance with a cryptographic algorithm and a cryptographic key length specified in [Table 7-11] List of Cryptographic Algorithm Standards.

[Table 7-11] List of Cryptographic Key Generation Standards

Cls.	Cryptographic Algorithm	Cryptographic Key Length (Bit)	Reference Standard
Block cipher (Symmetrical key cryptography)	ARIA	256	KS X 1213-1

The TOE generates a random bit from the validated cryptographic module in [Table 7-12], using a random bit generator specified in the following [Table 7-13] List of Random Bit Generation Standards while performing “the function of generating random bits.”

[Table 7-12] Validated Cryptographic Module

Name of Cryptographic Module	Validation No.	Developer	Date Validated
MagicCrypto V2.3.0	CM-263-2030.1	Dream Security	2025-01-24

[Table 7-13] List of Random Bit Generation Standards

Cls.	Random Bit Generation Algorithm	Reference Standard	Remark
Random Bit Generator	HASH_DRBG(SHA-256)	ISO/IEC 18031	Pseudorandom function

While performing encryption/decryption per column specified, the TOE provides users with its capacity of encrypting/decrypting user data through a cryptographic operation in accordance with an algorithm and a cryptographic key length specified in [Table 7-11].

Using the Administrative Tool, the authorized administrator designates columns of the DBMS table protected, establish the application of an algorithm needed and performs encryption/decryption of user data.

If the Administrative Tool requests encryption/decryption of user data for the column established, the Agent performs encryption/decryption with a DEK used for encryption of user data distributed from the Management Server.

Once the authorized administrator completes encryption/decryption of user data, the original user data used are all deleted. If a hash algorithm is used, only encryption can be performed.

Once the authorized administrator completes encryption/decryption of user data, the user data are deleted from the DBMS protected and residual information is destroyed.

Relevant SFR : FDP_UDE.1, FDP_RIP.1

7.4 Identification and Authentication

If there are [ont to five, (default: five)] unsuccessful authentication attempts by an unauthorized administrator, the TOE shall detect it. In case that the unsuccessful authentication attempts reach to a defined number, the TSF shall inactivate the authentication function for five to ten minutes (default: five minutes), deny any authentication and send a warning email to the authorized administrator.

In the TOE, the identification and authentication of the administrator are performed at once. The information provided through the screen GUI for identification/authentication of the administrator is an ID and a password, which are used to identify/authenticate the administrator. An action that can be taken before the administrator is identified/authenticated is a communication check. The administrator manages the security functions after he or she is successfully identified/authenticated.

TOE performs mutual authentication using TLS communication between the Management Server and the Agent, as well as additional self-implemented authentication.

The Management Server performs mutual authentication of users between the Agent and management tool though a self-implemented authentication protocol based on a certificate, ID and password.

1. The Agent or Administrative Tool conducts mutual authentication by installing their certificates respectively issued from the certificate of the Management Server in advance.
2. When an agent or management tool connects to the management server, the management server generates SignedData (PKCS#7) containing the management server certificate and transmits it to the agent or management tool.
3. The Agent or Administrative Tool performs a chain verification on the certificate of the Management Server included in the SignedData received and a certificate of the Agent or Administrative Tool.
4. The Agent and Administrative Tool generate random Key (1) and IV (1) (HASH_DRBG(SHA-256)).
5. Encrypt (RSAES (SHA-256)) the Key(1) and IV(1) with the received management server certificate. And SignedData containing encrypted data and management tool certificate is created and transmitted to the management server.
6. The Management Server performs mutual authentication by validating and verifying a chain of the Agent or Administrative Tool’s certificate.
7. The Key (1) and IV (1) encrypted in the SignedData that are received from the Agent or Administrative Tool are decrypted with a private key of the Management Server.
8. The Management Server generates Key (2) and IV (2) randomly (HASH_DRBG(SHA-256)).
9. Encrypt the Key(2) and IV(2) with the received management tool certificate (RSAES (SHA-256)) And Create SignedData, which includes encryption data and management server certificate, and send it to the management tool or agent.
10. The SignedData including Key (2) and IV (2) encrypted (RSAES(SHA-256)), using the private key of the Agent or Administrative Tool received, are transmitted to the Agent or Administrative Tool.
11. The Agent or Administrative Tool decrypts cryptographic Key (2) and IV(2) received, using a

private key of the Agent or Administrative Tool, and relies on cryptographic communications based on a session key for communication afterwards.

The TOE shall enforce password rules and shall provide a mechanism to verify that passwords satisfy the following rules: passwords shall contain at least one character from each of the following categories: digits (0–9), uppercase letters (A–Z), lowercase letters (a–z), special characters (~, `, !, @, #, $, %, ^, &, *, (,), -, _, +, =), and valid characters, with a valid password length of 9 to 16 characters. The TOE shall prohibit the setting of a password identical to the user account (ID), consecutive repetition of identical characters or digits, sequential input of consecutive characters or digits on the keyboard, and the reuse of any password that has been used within the previous three (3) months.

Password verification shall be performed each time the management tool logs into the management server, each time the agent is started, and each time the management tool is started.

The TOE shall allow the following actions to be performed on behalf of the administrator prior to administrator authentication: management server IP and management server port configuration, Nonce value exchange, and session key agreement. The TOE shall require the administrator to authenticate successfully before allowing any other TSF-mediated actions to be performed on behalf of the administrator.

The TOE shall prevent re-use of authentication data related to user authentication.

1. The Administrative Tool or Agent generates Nonce (R1) and transmits the ID and Nonce (R1) to the Management Server.
2. The server generates Nonce (R2), and performs an XOR operation of Nonce (R1), Nonce (R2) and PWD (1) after viewing the PWD (1) and Salt (1) with the subject ID of the Administrative Tool or Agent subject ID, followed by a hash (SHA256) to generate AuthValue (A1).
3. The Salt (1) viewed using the subject ID of the Administrative Tool or Agent and Nonce (R2) generated by the Management Server are transmitted to the Agent.
4. The Administrative Tool or Agent generates PWD (2) by associating Password (1) with Salt (1) received from the Management Server.
5. Nonce (R2) received and PWD (2) that previously generated Nonce (R1) are used to perform an XOR operation and hash to create AuthValue (A2). The AuthValue (A2) generated is transmitted to the Management Server.
6. The Management Server verifies AuthValue (A2) received by the Agent by comparing it with AuthValue (A1). The Agent performs authentication and transmits answers including the outcome of a login request if the authentication succeeds and a new session key in case of login success.
7. Re-use of authentication data is prevented by comparing AuthValue (A1) and AuthValue (A2).

A. With the Administrative Tool that uses a password-based authentication method, the TOE receives an ID and password and allows the authorized administrator to manage security functions.

B. An action that can be performed before an administrator is authenticated includes establishment of Management Server IP and port and, after the authentication, the administrator can manage security functions.

C. The Management Server performs mutual authentication of users between the Agent and Administrative Tool through a self-implemented protocol based on a certificate, ID and password.

1. The Agent or Administrative Tool conducts mutual authentication by installing their certificates respectively issued from the certificate of the Management Server in advance.

2. Once the Agent or Administrative Tool is accessed to the Management Server, the server signs on (RSA-PSS(SHA-256)) with a private key of the Management Server, generates SignedData (PKCS#7) including the certificate of the server and sends it to the Agent or Administrative Tool.
3. The Agent or Administrative Tool performs a chain verification on the certificate of the Management Server included in the SignedData received and a certificate of the Agent or Administrative Tool.
4. The Agent and Administrative Tool generate random Key (1) and IV (1) (HASH_DRBG(SHA-256)).
5. SignedData are generated using a private key of the Agent or Administrative Tool including cryptographic data encrypted (RSAES(SHA-256)) with the certificate of the Management Server that is the recipient of Key (1) and IV (1) randomly generated, and the SignedData generated are transmitted to the Management Server.
6. The Management Server performs mutual authentication by validating and verifying a chain of the Agent or Administrative Tool's certificate.
7. The Key (1) and IV (1) encrypted in the SignedData that are received from the Agent or Administrative Tool are decrypted with a private key of the Management Server.
8. The Management Server generates Key (2) and IV (2) randomly (HASH_DRBG(SHA-256)).
9. The Key (2) and IV (2) randomly generated are encrypted (RSAES(SHA-256)), using the certificate of the Agent or Administrative Tool received, and are transmitted to the Agent or Administrative Tool.
10. The SignedData including Key (2) and IV (2) encrypted (RSAES(SHA-256)), using the private key of the Agent or Administrative Tool received, are transmitted to the Agent or Administrative Tool.
11. The Agent or Administrative Tool decrypts cryptographic Key (2) and IV(2) received, using a private key of the Agent or Administrative Tool, and relies on cryptographic communications based on a session key for communications afterwards.

The TOE masks with * characters revealed to users while the administrator enters the password and provides feedback in the following texts in case of authentication failure with no mention of a failure reason: "Administrator has failed to log in."

The TOE shall allow establishment of the Management Server IP and port to be enforced on behalf of the authorized administrator before the administrator is identified, and successfully identify each authorized administrator prior to permission of all other actions mediated by the TSF on behalf of the authorized administrator. The Agent does not allow any additional IP.

Relevant SFR : FIA_AFL.1, FIA_IMA.1, FIA_SOS.1, FIA_UAU.1, FIA_UAU.4, FIA_UAU.7, FIA_UID.1

7.5 Security Management

The TOE restricts management actions to the authorized administrator with the capacity of managing security functions specified in [Table 7-14].

[Table 7-14] List of Authorized Administrator's Security Functions

Security Function	Management Action
Set-up and modification of administrator password	Determine and modify an action
Set-up and modification of audit information thresholds	Determine and modify an action
Modification of the number of authentication failures and inactivity time	Determine and modify an action
Set-up and modification of the mail server	Determine and modify an action
Set-up and modification of administrator access IP	Determine and modify an action

Security Function	Management Action
Real-time validation of modules	Initiate an action
View of audit information	Determine an action
Set-up and modification of cryptographic keys	Determine and modify an action
Establishment and modification of policies	Determine and modify an action
Column encryption	Determine an action

The TOE restricts its capacity of managing TSF data specified in [Table 6-15] to the authorized administrator.

[Table 7-15] List of Authorized Administrator's TSF Data

TSF Data List	Management
Audit information	Query
Modification of administrator password	Modification
Set-up of audit thresholds	Query, modification
Set-up of the mail server	Query, modification
Administrator information	Query, modification
Set-up of administrator access IP	Query, modification
Cryptographic keys (user data encryption)	Query
Policy list	Query, modification
Authentication failure information	Query, modification
Column list	Query, modification

The TOE shall provide a function to require the authorized administrator to change the password upon initial login. The password rules shall require passwords to contain at least one character from each of the following categories: digits (0–9), uppercase letters (A–Z), lowercase letters (a–z), special characters (~, `, !, @, #, $, %, ^, &, *, (,), -, _, +, =), and valid characters, with a valid password length of 9 to 16 characters. The TOE shall prohibit the setting of a password identical to the user account (ID), consecutive repetition of identical characters or digits, and sequential input of consecutive characters or digits on the keyboard. The authority to create and modify the ID/password of administrators and agents shall be restricted to the authorized administrator.

The authorized administrator of the TOE shall perform the following roles: administrator password configuration and modification, audit threshold configuration and modification, authentication failure count and inactivity timeout modification, mail server configuration and modification, administrator access IP configuration and modification, real-time module verification, audit information inquiry, cryptographic key configuration and modification, policy configuration and modification, and column encryption management.

Relevant SFR : FMT_MOF.1, FMT_MTD.1, FMT_PWD.1, FMT_SMF.1, FMT_SMR.

7.6 Protection of the TSF

During data transmission between the Management Server and the Agent, the TOE protects data from disclosure and modification by employing ARIA-128-CBC encryption of a validated cryptographic module (KCMVP). Encrypted communication is performed over TLS using a proprietary authentication protocol, and the procedure is as follows.

1. The Agent or Administrative Tool conducts mutual authentication by installing their certificates respectively issued from the certificate of the Management Server in advance.
2. When an agent or management tool connects to the management server, the management server generates SignedData (PKCS#7) containing the management server certificate and transmits it to the agent or management tool.
3. The Agent or Administrative Tool performs a chain verification on the certificate of the Management Server included in the SignedData received and a certificate of the Agent or Administrative Tool.
4. The Agent and Administrative Tool generate random Key (1) and IV (1) (HASH_DRBG(SHA-256)).
5. Encrypt (RSAES (SHA-256)) the Key(1) and IV(1) with the received management server certificate.

And SignedData containing encrypted data and management tool certificate is created and transmitted to the management server.

6. The Management Server performs mutual authentication by validating and verifying a chain of the Agent or Administrative Tool's certificate.
7. The Key (1) and IV (1) encrypted in the SignedData that are received from the Agent or Administrative Tool are decrypted with a private key of the Management Server.
8. The Management Server generates Key (2) and IV (2) randomly (HASH_DRBG(SHA-256)).
9. Encrypt the Key(2) and IV(2) with the received management tool certificate (RSAES (SHA-256)) And Create SignedData, which includes encryption data and management server certificate, and send it to the management tool or agent.
10. The Administrative Tool decrypts cryptographic Key (2) and IV (2) received, using a private key of the Administrative Tool, and relies on cryptographic communications (ARIA-128-CBC) with a session key for communication afterwards.

With a validated cryptographic module that performs encryption, the TOE protects against unauthorized disclosure and modification.

[Table 7-16] Methods of TSF Data Protection

Component	To be Encrypted	Cryptographic Key	Cryptographic Algorithm	Storage Location
Management Server	User data encryption key	Generated key (DEK)	ARIA/CBC/256	File
	TSF data encryption key	Generated key (KEK)	ARIA/CBC/256	File
	Integrity verification key	Generated key (KEK)	ARIA/CBC/256	File
	Configuration information	Generated key (DEK)	ARIA/CBC/256	File
	Administrator password	Random value (SALT)	SHA256	File
	Session encryption key	Generated key (DEK)	ARIA/CBC/256	Memory
	CSP	Generated key (DEK)	ARIA/CBC/256	File
	Audit data	Generated key (DEK)	ARIA/CBC/256	File
	Mail account information	Generated key (DEK)	ARIA/CBC/256	File
Agent	Configuration information	Generated key (DEK)	ARIA/CBC/256	File
	TSF data encryption key	Generated key (KEK)	ARIA/CBC/256	File
	Integrity verification key	Generated key (KEK)	ARIA/CBC/256	Memory
	Session encryption key	Generated key (DEK)	ARIA/CBC/256	Memory
	User data encryption key	Generated key (DEK)	ARIA/CBC/256	Memory
	CSP	Generated key (DEK)	ARIA/CBC/256	Memory
Administrative Tool	Configuration information	Generated key (DEK)	ARIA/CBC/256	File
	Integrity verification key	Generated key (KEK)	ARIA/CBC/256	Memory
	TSF data encryption key	Generated key (KEK)	ARIA/CBC/256	File
	Session encryption key	Generated key (DEK)	ARIA/CBC/256	Memory

User data cryptographic keys of the Management Server randomly (HASH_DRBG(SHA-256)) generate a DEK and IV for a DEK to encrypt the user data cryptographic keys when a cryptographic key is generated. The DEK and IV for DEK randomly generated encrypt user data cryptographic keys (ARIA/CBC/256). The DEK and IV for DEK that utilize user data cryptographic keys for encryption are encrypted (ARIA/CBC/256) with a KEK and stored in a separate file. The IV for DEK used herein is an IV for a KEK randomly generated as per user data cryptographic key in the file DB stored along with critical security parameters. User data cryptographic keys encrypted (ARIA/CBC/256) and the DEK and IV for DEK that encrypt (ARIA/CBC/256) user data cryptographic keys ensure integrity with HMAC-SHA256 by using the KEK as a key.

The Agent performs encryption (ARIA/CBC/256) with a cryptographic key (DEK and IV) shared from mutual authentication of user data cryptographic keys generated in the Management Server, to be delivered in real time during start-up of the Agent or generation of user data cryptographic keys, while the user data cryptographic keys delivered are stored in shared memories that keep user data cryptographic keys, policy names and critical security parameters of user data encryption that have been encrypted (ARIA/CBC/256) with the DEK and IV for DEK randomly generated (HASH_DRBG(SHA-256)). The DEK and IV for DEK are encrypted (ARIA/CBC/256) with a KEK while IV for KEK is randomly generated (HASH_DRBG(SHA-256) and stored in shared memories.

The configuration files used in the TOE are comprised of DEKs and IV for DEKs randomly generated (HASH_DRBG(SHA-256)) which encrypt (ARIA/CBC/256) configuration detail when the configuration files are created. The DEKs and IV for DEKs are encrypted(ARIA/CBC/256) into KEKs and IV for KEKs to store encrypted configuration detail in the configuration files that store Slat used for generation of a KEK, encrypted DEKs and IV for DEKs following the configuration detail encrypted and cryptographic module files and a hash of each executable shell script during installation. All details stored ensure integrity based on HMAC-SHA256 using the KEK as a key.

The section between the Management Server and Administrative Tool and that between the Management Server and Agent of the TOE are connected through an encryption session. An encryption session key for the encryption session is distributed through a self-implemented mutual authentication protocol, using a certificate. When a session key is generated or distributed, the encryption session key randomly (HASH_DRBG(SHA-256)) generates a DEK for encryption (ARIA/CBC/256) to be encrypted (ARIA/CBC/256) with a KEK and IV for a KEK and stored.

The IV for DEK, critical security parameters and audit data needed for encryption of user data cryptographic key files in the management files are encrypted and stored in the file DB which is comprised of multiple pages in a unit of 8192 bytes. On each page, critical TSF data such as critical parameters and audit data are encrypted and stored as a DEK and IV for a DEK randomly generated (HASH_DRBG(SHA-256)). The IV for KEK and encrypted DEK and IV for DEK are stored on each page. Integrity is ensured based on HMAC-SHA256 using the KEK as a key.

The Management Server and Agent generate a log file for tracking of execution and the log file ensures integrity as per each line based on a hash (SHA-256). The hash of each line is associated with the hash value of the previous line, generating association among the previous, current and next lines and ensuring integrity. As for the first line, a hash value is generated in association with a KEK. The integrity of the entire log file is ensured by verifying the first line whose hash value is associated with the KEK, followed by subsequent lines in association with the hash value of the previous line.

The KEK shall be derived using the PBKDF2(SHA-256) algorithm of the verified cryptographic module (KCMVP), based on the password entered at startup, the Salt (16 bytes) stored in the configuration file, and an iteration count of 1024. The derived KEK shall then be XOR-operated with a separately generated random vector (Random Vector, 32 bytes) produced by HASH_DRBG(SHA-256), and the result shall be encoded using Base64 encoding. The entered password shall be destroyed by overwriting with zeros three (3) times..

A self-test is performed to maintain secure and accurate operation conditions during the start-up of components and normal operation on a periodical basis, and upon the request of the administrator, using a TSF data protection method specified in [Table 7-16] for protection of the TSF.
The self-test is conducted during the initial start-up or every three hours after loading.
The self-test is conducted on test items in [Table 7-17] depending on the TOE classification in [Table 7-17]; and

in case of a self-test failure, a warning message is notified to the administrator via email along with details including the time of self-test failure (reliable OS time of the operational environment) and the failed file(s).

[Table 7-17] Items of Self-test

TOE Classification	Self-test Item	Testing
Management Server	Validated Cryptographic Module MagicCrypto V2.3.0	Self-test internally performed by the validated cryptographic module
	Process Name MagicDBPolicy	Check if main processes needed to run the Management Server are in normal operation and send the result to audit logs
Agent	Validated Cryptographic Module MagicCrypto V2.3.0	Self-test internally performed by the validated cryptographic module
	Process Name MDBAgent	Check if main processes needed to run the Agent are in normal operation and send the result to audit logs
Administrative Tool	Validated Cryptographic Module MagicCrypto V2.3.0	Self-test internally performed by the validated cryptographic module
	Process Name MagicDBPlus_v2.1_Admin	Check if main processes needed to run the Administrative Tool are in normal operation and send the result to audit logs

The following [Table 7-18] describes TSF integrity test methods under which integrity is tested for test items in [Table 7-18] depending on the TOE classification in [Table 7-18] in accordance with the testing detail specified in [Table 7-18]. In case integrity verification fails due to data damage from an unauthorized user, a warning message is sent to the administrator via email.

[Table 7-18] Items of TSF Integrity Test

TOE Classification	Item of Integrity Test	Testing
Management Server	MagicDBPolicy.conf	The entire configuration information is verified with a signature value using the certificate issued during the initial installation, and HMAC generated with a KEK is stored when the configuration file is created. An integrity test is performed to see if there is any falsification due to an unauthorized user during the start-up of the Management Server, upon the request of the administrator using the Administrative Tool and every three hours since the start-up of the Management Server. The test results are transmitted to audit logs and, if falsified, notified to the administrator via email together with the name of a falsified file. In case of configuration damage that keeps the Management Server from being in normal operation, the audit log and email are sent during the server's next normal operation.
	libMagicCrypto.so start.sh stop.sh restart.sh	Hash (SHA256) values from libMagicCrypto.so, start.sh, stop.sh, restart.sh installed during generation of the configuration file are stored in the configuration file. An integrity test is performed to see if there is any falsification due to an unauthorized user during the start-up of the Management Server, upon the request of the administrator using the Administrative Tool and every three hours since the start-up of the Management Server. The test results are transmitted to audit logs and, if falsified, notified to the administrator via email together with the name of a falsified file. In case of the cryptographic module's damage that keeps the Management Server from being in normal operation, the audit log and email are sent during the server's next normal operation.
	magicdb.dat magicdb.audit.dat	HMAC generated with a KEK for each block in a certain size is stored during data recording. An integrity test is performed to see if there is any falsification due to an unauthorized user during the start-up of the Management Server, upon the request of the administrator using the Administrative Tool, and every three hours since the start-up of the Management Server. The test results are transmitted to audit log and, if falsified, notified to the administrator via email together with the name of a falsified file. In case of file DB damage that keeps the administrator's email

TOE Classification	Item of Integrity Test	Testing
		address and email server information from being successfully uploaded, the email is sent, using the information in the memories stored by the Management Server during the start-up and a change in information of the administrator's email address and email server.
	Cryptographic Key File	HMAC generated with a KEK is stored in the file when a cryptographic key is generated. An integrity test is performed to see if there is any falsification due to an unauthorized user during the start-up of the Management Server, upon the request of the administrator using the Administrative Tool and every three hours since the start-up of the Management Server. The test results are transmitted to audit log and, if falsified, notified to the administrator via email together with the name of a falsified file
	Log File	If a log is added to the log file, a chain verification is needed for each line of logs using a hash (SHA256). When the initial log file is generated, its first line and a KEK are associated together to generate a hash value which is recorded on the last line of the log. The following line is associated together with the previous line to generate a hash value. The entire file can be verified by associating the first line with a KEK and performing a hash operation to compare and verify the hash value recorded on the corresponding line, followed by the subsequent lines which go through a hash operation and verification. An integrity test is performed to see if there is any falsification due to an unauthorized user during the start-up of the Management Server, upon the request of the administrator using the Administrative Tool and every three hours since the start-up of the Management Server. The test results are transmitted to audit logs and, if falsified, notified to the administrator via email together with the name of a falsified file.
Agent	MDBAgent.conf	The entire configuration information is verified with a signature value using the certificate issued during the initial installation, and HMAC generated with a KEK is stored when the configuration file is created. An integrity test is performed to see if there is any falsification due to an unauthorized user during the start-up of the Management Server, upon the request of the administrator using the Administrative Tool and every three hours since the start-up of the Management Server. The test results are transmitted to audit logs and, if falsified, notified to the administrator via email together with the name of a falsified file. In case of configuration damage that keeps the Agent from being in normal operation, the audit log and email are sent during next normal operation.
	libMagicDB.so libMagicCrypto.so start.sh stop.sh restart.sh	Hash (SHA256) values from libMagicDB.so, libMagicCrypto.so, start.sh, stop.sh, restart.sh installed during generation of the configuration file are stored in the configuration file. An integrity test is performed to see if there is any falsification due to an unauthorized user during the start-up of the Management Server, upon the request of the administrator using the Administrative Tool and every three hours since the start-up of the Management Server. The test results are transmitted to audit logs and, if falsified, notified to the administrator via email together with the name of a falsified file. In case of the cryptographic module's damage that keeps the Agent from being in normal operation, the audit log and email are sent during next normal operation.
	Log File	If a log is added to the log file, a chain verification is needed for each line of logs using a hash (SHA256). When the initial log file is generated, its first line and a KEK are associated together to generate a hash value which is recorded on the last line of the log. The following line is associated together with the previous line to generate a hash value. The entire file can be verified by associating the first line with a KEK and performing a hash

TOE Classification	Item of Integrity Test	Testing
		operation to compare and verify the hash value recorded on the corresponding line, followed by the subsequent lines which go through a hash operation and verification. An integrity test is performed to see if there is any falsification due to an unauthorized user during the start-up of the Management Server, upon the request of the administrator using the Administrative Tool and every three hours since the start-up of the Management Server. The test results are transmitted to audit logs and, if falsified, notified to the administrator via email together with the name of a falsified file.
	DEK	A DEK for user data delivered to the Management Server is encrypted (ARIA-256-CBC) and stored in the memories and, while being stored, its integrity is ensured using HMAC values generated with a KEK. An integrity test is performed to see if there is any falsification due to an unauthorized user during the start-up of the Management Server, upon the request of the administrator using the Administrative Tool and every three hours since the start-up of the Management Server. The test results are sent to the administrator via email.
Administrative Tool	MDBAdmin.xml.enc	HMAC generated with a KEK is stored when the configuration file is created. An integrity test is performed to see if there is any falsification due to an unauthorized user during the start-up of the Management Server, upon the request of the administrator using the Administrative Tool and every three hours since the start-up of the Management Server. The test results are transmitted to audit logs and, if falsified, notified to the administrator via email together with the name of a falsified file. In case of configuration damage that keeps the Administrative Tool from being in normal operation, the audit log and email are sent during next normal operation.
	MagicCryptoV23.dll install.ico uninstall.ico Uninstall.exe	Hash (SHA256) values from MagicCryptoV23.dll, install.ico, uninstall.ico, Uninstall.exe installed during generation of the configuration file are stored in the configuration file. An integrity test is performed to see if there is any falsification due to an unauthorized user during the start-up of the Management Server, upon the request of the administrator using the Administrative Tool, and every three hours since the start-up of the Management Server. The test results are transmitted to audit logs and, if falsified, notified to the administrator via email together with the name of a falsified file. In case of the cryptographic module's damage that keeps the Administrative Tool from being in normal operation, the audit log and email are sent during next normal operation.

When entropy is reseeded into the random bit generator during integrity verification of the validated cryptographic module in the TOE's Management Tool and Agent, if an error occurs in the RCT (Repetition Count Test) or APT (Adaptive Proportion Test) health tests performed on the collected entropy noise source, or if integrity compromise is detected, the Management Tool and Agent shall be manually reinstalled to restore the TOE to a secure state.

Relevant SFR : FPT_ITT.1, FPT_TST.1, FPT_PST.1, FPT_RCV.1, FPT_FLS.1

7.7 TOE Access

The TOE limits the maximum number of concurrent sessions of management access by the same administrator with the same authority to one. If there is another attempt to access using the same account after the administrator has logged in, the first comer takes the priority and the new access is blocked.

The TOE is only accessible from the Administrative Tool with a pre-registered IP address, and as for the

management access, any access not from the access IP defined by the authorized administrator shall be denied. By default, there are two access IPs defined by the authorized administrator and they can be added or deleted using the Administrative Tool.

- The access IP is restricted to IPv4's address system.
- The IPv4's address system is comprised of 12 digits in total which can be divided into four sections with three digits per each. Each section is expressed as an integer of 1 to 255 in one to three digits and is divided with '.' as in A.B.C.D.

Once the administrator has logged in, the access session performs session termination after inactivity time without any action for a certain period. The administrator's inactivity time is defined as 10 minutes and the absence of following requests that communicate with the Management Server is considered as inactivity time:

- Request to view a list of cryptographic keys
- Request to register a cryptographic key
- Request to modify a cryptographic key
- Request to search a policy list
- Request to register a policy
- Request to view a list of users
- Request to register a user
- Request to modify a user
- Request to search logs

Relevant SFR : FTA_MCS.2, FTA_SSL.3, FTA_TSE.11