eXSignOn V4.0 Security Target
V1.7

TOMATO SYSTEM

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

[Revision History]

Version	Detail	Date	Created by	Reviewed by
1.0	Initial version	2025-05-14	Jo Ye-won	Kwon Hyuk-se
1.1	Reflected results of the 1st preliminary review by Korea System Assurance, Inc	2025-05-23	Jo Ye-won	Kwon Hyuk-se
1.2	Reflected results of the 1st formal review by Korea System Assurance, Inc	2025-06-12	Kim Min-soo	Kwon Hyuk-se
1.3	Reflected results of the 2nd formal review by Korea System Assurance, Inc	2025-07-11	Kim Min-soo	Kwon Hyuk-se
1.4	Reflected results of the 3rd formal review by Korea System Assurance, Inc	2025-07-30	Kim Min-soo	Kwon Hyuk-se
1.5	Reflected results of the 4rd formal review by Korea System Assurance, Inc	2025-08-01	Kim Min-soo	Kwon Hyuk-se
1.6	Reflected results of the 5rd formal review by Korea System Assurance, Inc	2025-08-05	Kim Min-soo	Kwon Hyuk-se
1.7	Reflected results of the 6rd formal review by Korea System Assurance, Inc	2025-08-06	Kim Min-soo	Kwon Hyuk-se

[Contents]

[Table of Contents]

[Figure of Contents]

1. ST introduction

1.1. ST reference

Table 1. ST introduction

Title	eXSignOn V4.0 Security Target
Version	V1.7
Ecaluation Assurance Level	EAL1+(ATE_FUN.1)
Developer	Tomato System Co., LTD.
Evaluation Criteria	Common Criteria for Information Technology Security Evaluation (Ministry of Science, ICT and Future Planning Notification No. 2013- 51, 2013.8.8.)
Common Criteria	Common Criteria for Information Technology Security Evaluation, Version CC:2022 R1 - Common Criteria for Information Technology Security Evaluation. Part 1 : Introduction and General Model, Version CC:2022 R1, (CCMB-2022-11-001, 2022. 11) - Common Criteria for Information Technology Security Evaluation. Part 2 : Security Functional Components, Version CC:2022 R1, (CCMB-2022-11-002, 2022. 11) - Common Criteria for Information Technology Security Evaluation. Part 3 : Security Assurance Component, Version CC:2022 R1, (CCMB-2022-11-003, 2022. 11) - Common Criteria for Information Technology Security Evaluation. Part 4 : Framework for the specification of evaluation methods and activities, Version CC:2022 R1, (CCMB-2022-11-004, 2022. 11) - Common Criteria for Information Technology Security Evaluation. Part 5 : Pre-defined packages of security requirements, Version CC:2022 R1, (CCMB-2022-11-005, 2022. 11) - Errata and Interpretation for CC:2022 (Release 1) and CEM:2022 (Release 1), Version 1.1, (CCMB-2024-07-002, 2024.07)
Keywords	Single Sign On, SSO
Release Date	August 6, 2025

1.2. TOE reference

Table 2. TOE reference

TOE	eXSignOn V4.0
Version	V4.0.005
TOE Components	eXSignOn Sever V4.0.005 (TMTEXS_SERVER_V4.0.005.zip) eXSignOn Agent V4.0.005 (TMTEXS_AGENT_V4.0.005.zip)
Manuals	eXSignOn V4.0 operating manual V1.8 (TMTEXS_OPE_V1.8.pdf) eXSignOn V4.0 preparation procedure V1.8 (TMTEXS_PRE_V1.8.pdf)
Developer	Tomato System Co., LTD.

1.3. TOE overview

1.3.1. Single Sign On overview

eXSignOn V4.0 (hereinafter referred to as the TOE) is used to provide users with services to various business systems through a one-time login (Single Sign-On), without requiring additional login actions. The TOE performs user identification and authentication, as well as the issuance and validation of authentication tokens, in accordance with user authentication policies.
The TOE provides a user login function using an ID/password authentication method. During the login process, the TOE issues an authentication token, and when the user accesses other business systems after logging in, the issued authentication token must be validated. At this time, the TOE requires ID and password-based authentication for both authorized administrators and authorized general users.
The main security functions provided by the TOE include user identification and authentication, and the issuance, storage, validation, and disposal of authentication tokens. For the generation of authentication tokens and integrated authentication based on authentication tokens, the TOE uses validated cryptographic modules whose security and implementation compliance have been verified through the Korea Cryptographic Module Validation Program (KCMVP).

1.3.2. TOE type and scope

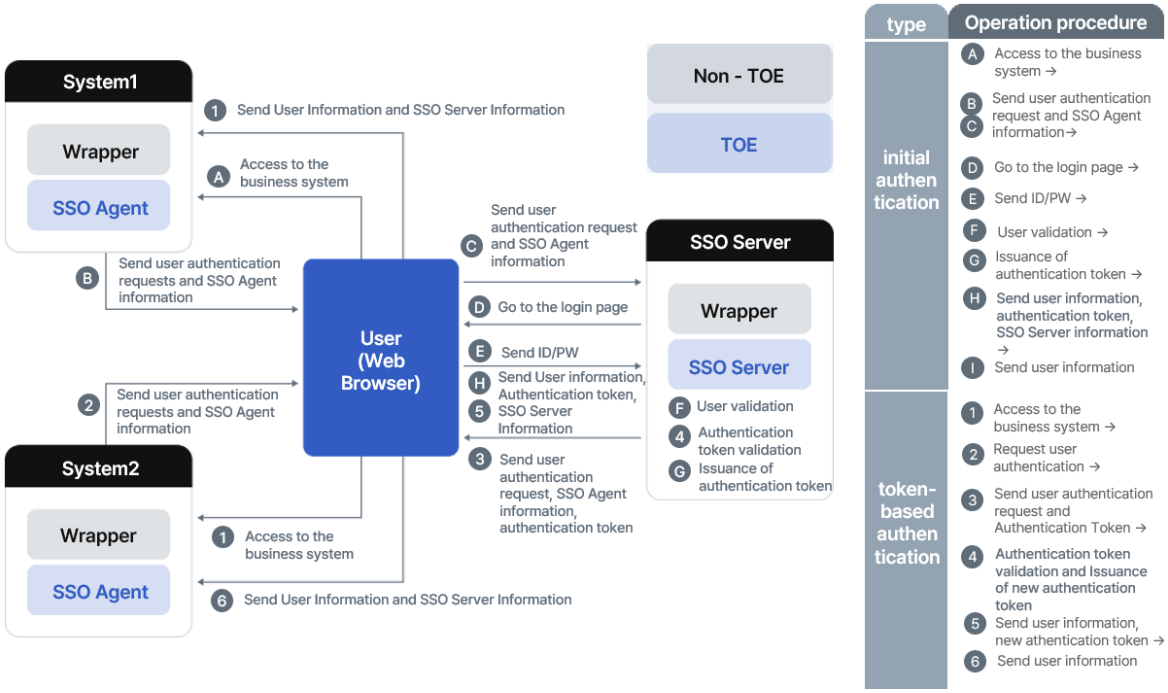
The TOE defined in this Security Target is an integrated authentication system (Single Sign-On, SSO) that allows users to access various business systems through a single login, without additional login actions. The TOE is provided in the form of software, and consists of the SSO server and the SSO agent.
The TOE is composed of a server that performs functions such as user login processing, authentication token issuance and validation, and policy configuration, and an agent that is installed in each business system to perform SSO interworking functions. The agent is provided in the form of an ‘API type’ composed of library files.
This Security Target defines the mandatory security functional requirements, conditionally mandatory security functional requirements, and optional security functional requirements to be provided by the indispensable TOE components — the SSO server and the SSO agent — and the TOE complies with these security functional requirements.

1.3.3. TOE usage and major security features

The TOE performs user identification and authentication to enable the user to access various business systems and use the service through a single user login without additional login action.
The TOE provides the security audit function that records and manages a critical events as audit data when activating the security functionality and management function, function of protecting the data that stored in the TSF controlled repository, and TSF protection function such as TSF self-testing. In addition, the TOE provides authentication failure handling, identification and authentication functions including mutual authentication between the TOE components, cryptographic support function such as cryptographic key management and cryptographic operation for issuing a token, security management function for management of security functions behaviour and configuration, and the TOE access function to manage the authorized administrator’s interacting session. In addition, the token requires confidentiality and integrity protection, and the TOE executable code requires integrity protection.
Figure 1 shows the user identification and authentication procedure of the general TOE. The detailed execution procedure can vary depending on the TOE implementation.
The user identification and authentication procedure can be grouped into the initial authentication phase using the ID/password, and the token-based authentication phase that accesses the business system using the token issued during the initial authentication procedure.
The execution procedure of the initial authentication phase is as follows. The user requests login using the ID/password, and the SSO agent that receives the login request message sends a login verification request to the SSO server, which in turn checks the authorized user status.
Upon receiving a login verification request, the SSO server authenticates the user by referencing credentials stored in the DBMS.
If the verification is successful, the SSO server issues an authentication token to the user's browser and transmits authenticated user information along with SSO server metadata to the SSO agent.
The SSO agent validates the received SSO server metadata and completes the authentication process for the business application using the verified user information.
The token-based authentication phase is only initiated when an authentication token has been successfully issued during the initial login process.

When a user attempts to access a business system service, the SSO agent forwards the authentication request and its own metadata to the SSO server via the user's browser, accompanied by the authentication token stored in the browser.
Upon receipt, the SSO server verifies both the SSO agent metadata and the authentication token. Immediately after verification, the token is invalidated and discarded.
Once verification is complete, the SSO server issues a new authentication token to the user's browser and transmits authenticated user information and updated SSO server metadata to the SSO agent.
The SSO agent then revalidates the SSO server metadata and finalizes the authentication process for the business system using the received user credentials.

Figure 1. User identification and authentication procedure



The user identification and authentication procedure can be executed with various procedures depending on the TOE implementation. The following table shows the example of operation by phase.

Table 3. Example of operation procedure by authentication phase

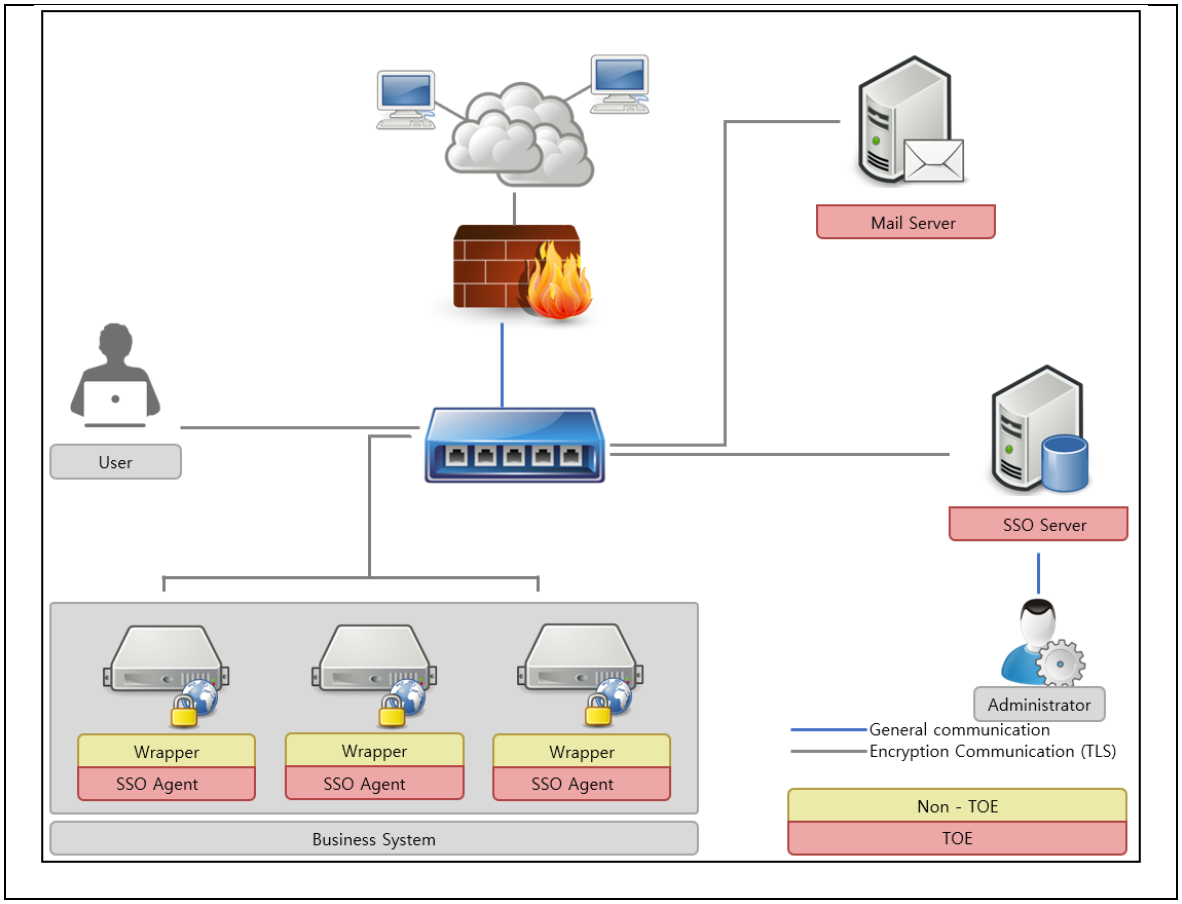
authentication phase	example of operation procedure
initial authentication	(A) Access to the business system ⟶ (B) Send user authentication request and SSO Agent information ⟶ (C) Send user authentication request and SSO Agent information ⟶ (D) Go to the login page ⟶ (E) Send ID/PW ⟶ (F) User validation ⟶ (G) Issuance of authentication token ⟶ (H) Send user information, authentication token, SSO Server information ⟶ (I) Send user information
token-based authentication	(1) Access to the business system ⟶ (2) Request user authentication ⟶ (3) Send user authentication request and authentication token ⟶ (4) Authentication Token validation and Issuance of new authentication token ⟶ (5) Send user information, new authentication token ⟶ (6) Send User Information

In addition, the subject who issues, stores, and verifies the token can be different, depending on the implementation. The following is an example of the subject who issues, stores, and verifies the token.

- Subject who issues the token: SSO Server
- location: User PC(Web Browser)
- Subject who verifies the token: SSO Server

1.3.4. Non-TOE and TOE operational environment

Figure 2. TOE operational environment



The TOE operating environment consists of an SSO server and SSO agents, as shown in <Figure 2>. The SSO server provides functions such as direct user login verification using user information stored in the DBMS, user login verification through system integration, authentication token management, and policy configuration. SSO agents request user login and authentication token validation from the SSO server and are installed on each business system. Additionally, SSO agents are installed in the form of an API, implemented as a library file within each business system.
Authorized administrators can access the SSO server via an administrator web browser to perform security management. For development convenience within the TOE operating environment, a wrapper composed of sample files may be used; however, the wrapper is excluded from the scope of the TOE.
An external entity required for operating the TOE includes a mail server used to notify authorized administrators in case of predicted audit data loss. Communication with the mail server is conducted using TLS-encrypted transmission.
The mail server is considered as part of the TOE operating environment.
The hardware and software requirements necessary for operating the TOE are as follows:

Table 4. Minimum Hardware and Software Requirements for Operating TOE

Item	Details	
Hardware	SSO Server	[CPU] Intel Core i5 2.5GHz or higher [RAM] 16GB or more [HDD] Minimum of 50GB of available space for TOE installation or more [Network] At least one 100/1000 Base-T Ethernet card (TCP/IP-based) or more
	SSO Agent	[CPU] Intel Core i5 2.5GHz or higher

		[RAM] 8GB or more [HDD] Minimum of 10GB of available space for TOE installation or more [Network] At least one 100/1000 Base-T Ethernet card (TCP/IP-based) or more
Software	SSO Server	[OS] WindowsServer2022 (64bit) [DBMS] MariaDB 11.8.2 (64bit) [WAS] Apache Tomcat 10.1.43 (64bit) [JRE] OpenJDK 11.0.28 (64bit)
	SSO Agent	[OS] WindowsServer2019 (64bit), RHEL 9.0 (Linux 5.14.0) (64bit) [WAS] Apache Tomcat/10.1.43 (64bit) [JRE] OpenJDK 11.0.28 (64bit)
	Web Browser	Chrome v139.0.7258.67

1.4. TOE operational environment

1.4.1. Software, Hardware, and Firmware Requirements for non-TOE

Table 5. Non-TOE Software for operating TOE

TOE Components	S/W	Specification and Purpose
SSO Server	[DBMS] MariaDB 11.8.2 (64bit)	The DBMS stores configuration values essential for the TOE's operation and authentication data required for user authentication. It also stores generated audit data and provides authorized administrators with audit data search and ordering methods, as well as audit trail protection functions.
	[WAS] Apache Tomcat/10.1.43 (64bit)	The TOE provides web-based SSO using WAS, offering SSO through HTTP requests and sessions. It also provides a web-based security management interface (GUI: Graphical User Interface).
	[JRE] OpenJDK 11.0.28 (64bit)	The TOE is developed using Java and provides the underlying framework for the standard SSO process and the security management interface operation.
SSO Agent	[WAS] Apache Tomcat/10.1.43 (64bit)	The TOE provides web-based SSO using WAS, offering SSO through HTTP requests and sessions. It also provides a web-based security management interface (GUI: Graphical User Interface).
	[JRE] OpenJDK 11.0.28 (64bit)	The TOE is developed using Java and provides the underlying framework for the standard SSO process and the security management interface operation.

Table 6. External entity supporting TOE

Item	TOE Support Function
Mail server	SMTP server Interfaces are provided to send alert emails to administrators when administrator authentication failures, audit storage saturation, or integrity violation events are detected The mail server operates on the internal network Connected to the SSO server via SSL

For data transmission between TOE components, a proprietary protocol is used for communication channel encryption and mutual authentication between components. Administrators and users communicate via a secure channel (HTTPS) supported by the operating environment when logging in through a web browser on their PCs, ensuring secure communication.

Table 7. Identification of validated cryptographic modules and cryptographic algorithms for TOE components

Item	Specification	Note
Validated Cryptographic Module	eXCryptoLib V1.0	Validation Date : April 16, 2025 Expiration Date : April 16, 2030 Validation Number : CM-268-2030.4
TOE Internal mutual authentication	Proprietary protocol Confidentiality/Faultiness: ARIA/CCM(128 bit) Key exchange: RSAES (2048 bit)	SSO Server <-> SSO Agent

Web channel data protection	HTTPS Protocol : TLSv1.3		- Administrator PC Web Browser <-> SSO Server - User PC Web Browser <-> SSO Agent
External entity Communication	Cipher Suite: TLS_AES_256_GCM_SHA384 Key exchange : ECDHE		SSO Server <-> Mail Server
Approved cryptographic algorithm	Symmetric	ARIA/CBC(128bit)	TSF Data, Encryption Key Encryption/Decryption
		ARIA/CBC(128bit)	Authentication Token, Transmission data Encryption/Decryption
	Random	HASH_DRBG (SHA-512)	Create Encryption Keys and IVs Random Number generated when Authentication Token is generated
	Message authentication	HMAC-SHA256	TSF file integrity verification
	Hash	SHA-256	User, Administrator Password Hash
	Key deriving	RSAES(2048 bit)	Encryption for SSO Server-to-SSO Agent Mutual Authentication Session Key
	Key deriving	PBKDF (HMAC(SHA-256))	KEK generation

1.5. TOE description

1.5.1. Physical scope of the TOE

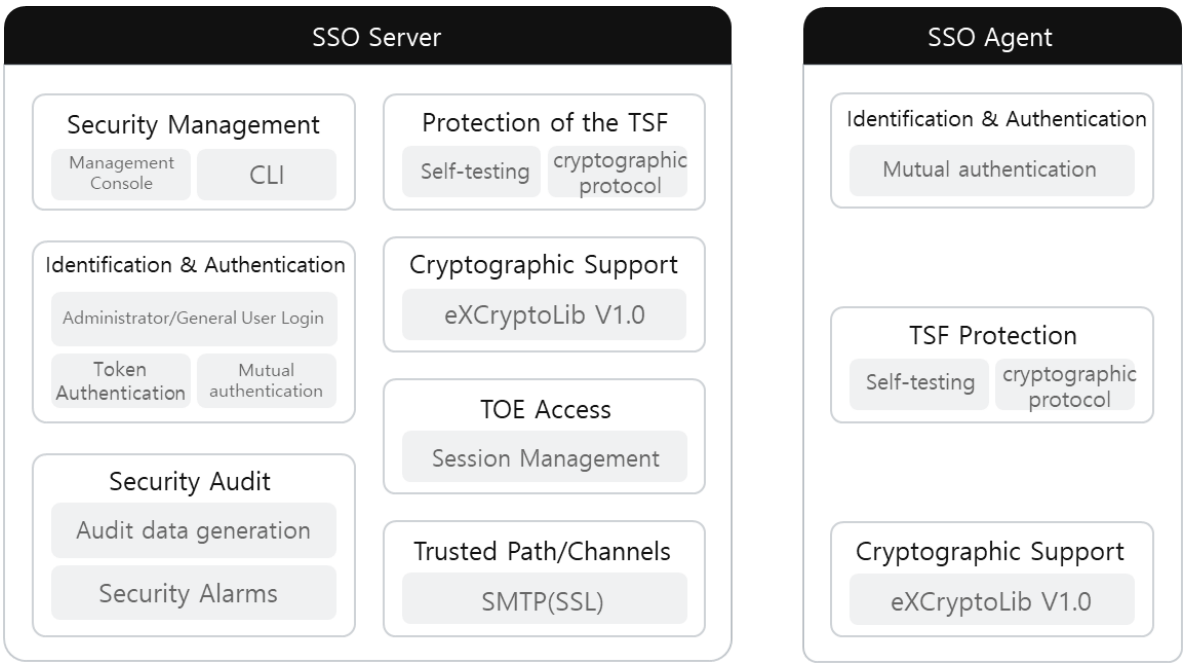
The TOE consists of the SSO server, SSO agents, and software required for the operating environment, which is outside the TOE scope.
The eXSignOn V4.0 product includes the SSO server and SSO agent programs developed by Tomato System Co., LTD., along the preparation guide and the operation manual required for administrative operations. The product CD contains the SSO server, SSO agent, wrapper, and documentation; however, it does not include third-party software such as WAS and DBMS, which are required for installation and operation.
In addition, the hardware, web application server, database, JDK, SSL, and web browser required for product operation fall outside the TOE scope and are excluded from the physical scope of the TOE. These components, which are outside the physical scope, must be separately prepared by the customer.
Accordingly, the components of the TOE provided to the customer are essential software for installing the SSO server and SSO agent, the preparation guide, the operation manual, and the wrapper.
The product is distributed in the form of a CD-ROM.

Table 8. Deployment Type and Format of the TOE

Scope		Distribution Status	Deployment Type	Deployment Format
TOE		eXSignOn V4.0		
TOE Detailed Version		V4.0.005		
TOE Components (1EA)	**SSO Server**	eXSignOn Server V4.0.005 (TMTEXS_SERVER_V4.0.005.zip)	S/W	CD-ROM
	SSO Agent	eXSignOn Agent V4.0.005 (TMTEXS_AGENT_V4.0.005.zip)		
Manuals		eXSignOn V4.0 operational guidance V1.8 (TMTEXS_OPE_V1.8.pdf)	Electronic Document (PDF)	
		eXSignOn V4.0 preparation procedure V1.8 (TMTEXS_PRE_V1.8.pdf)		

1.5.2. Logical scope of the TOE

Figure 3. Logical scope of the TOE



o Security audit (FAU)
The TOE provides functions for generating audit records of security-related events and detecting potential violations to track accountability for security-related actions. Audit records for all operations performed by users over time are stored, and the generated audit records are stored in the DBMS through the SSO server.
When the SSO server's disk capacity is saturated, an alert mail is sent to the authorized administrator in response to the loss of audit data, and a warning mail is sent to the authorized administrator in order to prevent the loss of audit data. If the loss prediction standard is exceeded, a saturation warning mail is sent, and if the saturation standard is exceeded, a saturation warning mail is sent.

Table 9. Audit data saturation and loss prediction criteria

Item	Setable range	Default	Note
Criteria for saturation	50%, 60%, 70%, 80%, 90%	80%	
Criteria for forecasting losses	40%, 50%, 60%, 70%, 80%	70%	It is automatically set to a value 10% lower from the saturation reference setting.

o Cryptographic support (FCS)
The TOE uses validated cryptographic algorithms within the validated cryptographic module (eXCryptoLib V1.0, CM-268-2030.4), whose security and implementation compliance have been confirmed through the Korea Cryptographic Module Validation Program (KCMVP), to manage encryption keys, perform encryption operations, and generate random bits for SSO server–SSO agent communication.
Additionally, to protect transmitted data and stored TSF data (e.g., SSO server configuration files, SSO server preferences stored in DB and files, files that store DEK/IV/TSF data integrity verification keys (HMACs), and files that store DBMS access information), it uses ARIA/CCM (128-bit) mode to ensure both confidentiality and integrity. It also performs functions such as the issuance, verification, storage, and disposal of authentication tokens, as well as the use of random bit generators for cryptographic key management (e.g., key generation).

o Identification and authentication (FIA)
The TOE performs secure encrypted communication between its components (SSO server and SSO agent) by implementing its own mutual authentication protocol. During mutual authentication, the following cryptographic algorithms are used:

- Generation of mutual authentication session keys and encrypted communication session keys: Hash_DRBG (random bit generator)
- Distribution (encryption/decryption) of session keys: RSAES (2048-bit)
- Authentication / encryption and decryption: ARIA/CCM (128-bit)
- SSO server identification information: SSO server domain
- SSO agent identification information: SSO agent domain and SPID

In addition, the SSO server performs user identification and authentication based on ID and password for both administrators and users, and user identification and authentication are required before any action is permitted. The password input field is masked with asterisks (*) to prevent it from being shown during input. Passwords must comply with the password policy defined by an authorized administrator. The TOE verifies the password used during an authentication attempt, does not provide feedback on failed authentication results, and prevents the reuse of authentication data.

To protect the TOE from improper authentication attempts, if the number of failed identification and authentication attempts exceeds the configured limit (default: 5 attempts, configurable from 1 to 5 attempts), the SSO server locks the account for the configured period of time (default: 5 minutes, configurable to 5/10/30/60 minutes). Locked accounts are automatically unlocked after the configured time period has elapsed.

Authentication
tokens used for user identification and authentication are generated using fixed values, the user ID, a timestamp, and random numbers produced by a random bit generator. To provide both confidentiality and integrity of the sensitive information contained in the token, the TOE uses a validated cryptographic module

applying ARIA/CCM (128-bit) mode for token encryption. Upon logout, the contents of the authentication token are securely deleted from memory using a triple overwrite method with the value ‘0x00’.

o Security management (FMT)
The SSO server can manage security functions and TSF data through a administrator web interface. The security management functions provided by the SSO server are as follows:

1. Security function management: Administrators can manage TSF functions. The TOE provides security policy management and monitoring, as well as audit data review functions.

2. Data management: The TOE manages TSF data. TSF data enables functions for managing security policies and viewing audit data.

3. Password management: Provides authorized administrators with functions to manage password length and combination rules, and enforces the change of default administrator passwords for new administrators upon first login.

4. Security role management: Authorized administrators of the TOE are classified as super administrators, general administrators, and monitoring administrators. Super administrators can manage all security functions of the TOE and add, delete, and grant privileges to general administrators, while general administrators can access some security functions or monitoring features as configured by the super administrator. Monitoring administrators can only monitor audit data. Only one administrator with modification privileges for security functions (excluding monitoring administrators) can be logged in at the same time to prevent simultaneous modifications to security functions.

o Protection of the TSF (FPT)
If a failure occurs in the entropy source (e.g., noise source health test failure), the TOE transitions to a critical error state and halts the operation of the validated cryptographic module to maintain a secure state. After mutual authentication between TOE components (SSO server and SSO agent) using its own protocol, the SSO server uses the distributed session key to perform encrypted communication with ARIA/CCM (128-bit), ensuring the confidentiality and integrity of transmitted TSF data. The SSO server also performs TSF self-tests, including integrity tests using the HMAC-SHA256 algorithm for libraries, configuration files, and cryptographic modules. TSF self-tests are performed during initial start-up, every 12 hours after start-up, and upon administrator request. If the TSF self-test fails, the SSO server notifies the authorized administrator of the failure details via email in order to provide TSF protection.

o TOE access (FTA)
The SSO server limits the maximum number of concurrent administrators with privileges to change settings (super administrators and general administrators) to one. If another administrator logs in while an existing administrator is connected, the existing administrator's session is terminated. However, administrators with only monitoring privileges can log in concurrently. Administrator sessions are restrictively accessible to authorized personnel based on IP addresses. User sessions are restrictively accessible based on IP address, SSO agent ID, and SSO agent domain. If a user or administrator does not perform any activity for the configured session timeout period (default: 600 seconds, configurable from 60 to 600 seconds) after login, the session is terminated.

o Trusted path/channels (FTP)
When the TOE communicates with external IT entities such as an Mail server(SMTP server), it provides a secure communication path/channel to protect transmitted data by using TLS v1.3 with the TLS_AES_256_GCM_SHA384 cipher suite and the ECDHE key exchange algorithm.

1.6. Conventions

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

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

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

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

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

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

1.7. Terms and definitions

The terms used in this ST, which are the same as those in the CC, follow the definitions in the CC.

Application Programming Interface (API)
A set of software libraries that exist between the application layer and the platform system layer and facilitate the development of applications that run on the platform

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

Approved mode of operation
The mode of cryptographic module using approved cryptographic algorithm

Assets
Entities to which the TOE owner assigns value

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

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

Authentication Data
Information used to verify a user's claimed identity

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

Authorized Administrator
Authorized user to securely operate and manage the TOE

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

Business System
An application server that authorized end-users access through SSO.

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

Class
Set of CC families that share a common focus

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

Component
Smallest selectable set of elements on which requirements may be based

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

Database Management System (DBMS)

A software system composed to configure and apply the database.

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

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

Element
Indivisible statement of a security need

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

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

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

Evaluation Assurance Level (EAL)
Set of assurance requirements drawn from CC Part 3, representing a point on the CC predefined assurance scale, that form an assurance package

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

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

Identity
Representation uniquely identifying entities (e.g. user, process or disk) within the context of the TOE
Iteration
Use of the same component to express two or more distinct requirements

Korea Cryptographic Module Validation Program (KCMVP)
A system to validate the security and implementation conformance of cryptographic modules used for the protection of important but not classified information among the data communicated through the information and communication network of the government and public institutions.

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

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

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

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

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

Operation(on a component of the CC))
Modification or repetition of a component. Allowed operations on components are assignment, iteration, refinement and selection

Operation(on a subject)
Specific type of action performed by a subject on an object

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

Protection Profile (PP)
Implementation-independent statement of security needs for a TOE type

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

Public Key(asymmetric) cryptographic algorithm
A cryptographic algorithm that uses a pair of public and private key

Public Security Parameters (PSP)
security related public information whose modification can compromise the security of a cryptographic module

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

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

Refinement
Addition of details to a component

Role
Predefined set of rules establishing the allowed interactions between a user and the TOE

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

Security Attribute
Characteristic of subjects, users, objects, information, sessions and/or resources used in defining SFRs and their enforcement

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

Security Target (ST)
Implementation-dependent statement of security needs for a specific identified TOE

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

Self-test
Pre-operational and conditional tests performed by the cryptographic module

Sensitive Security Parameters (SSP)
critical security parameters (CSP) and public security parameters (PSP)

Session Key
Key generated from a validated cryptographic module, used for encryption communication for secure encryption communication between the SSO Server and the SSO Agent

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

SSL(Secure Sockets Layer)
Security protocol proposed by Netscape to provide security such as confidentiality and integrity in a computer network

Subject
Active entity in the TOE that performs operations on objects

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

Target of Evaluation (TOE)
Set of software, firmware and/or hardware possibly accompanied by guidance

Threat Agent
Entity that can adversely act on assets

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

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

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

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

SSO (Single Sign On)
The most basic authentication system, developed with the purpose of “all authentication in a single system.” This means that regardless of the number of systems, once authentication is successful in one system, access privileges to all other systems are also granted.

Server
A subsystem that provides centralized processing functions within a Local Area Network (LAN).

Agent
Software that communicates with the SSO server to assist with SSO authentication in the business systems that users will access.

1.8. ST organization

Chapter 1 introduces the ST, providing references, TOE references, TOE overview, and TOE description.

Chapter 2 declares the conformance claims to the CC, PP, and package, and describes the theoretical rationale for the conformance claims.

Chapter 3 defines the security problem to be addressed by the TOE and its operational environment in terms of threats, organizational security policies, and assumptions.

Chapter 4 defines the security objectives for the operational environment that support the TOE so that the TOE security functionality can be accurately provided.

Chapter 5 defines the extended components that need to be additionally defined according to the characteristics of the TOE.

Chapter 6 describes the security assurance requirements, including the security functional requirements and assurance requirements.

Chapter 7 summarizes the TOE functionality concisely as a TOE summary specification.

References describe the sources referred to for users who require more information than those described in this ST.

Abbreviations provide the abbreviations used in this ST.
Reference describes the references for users who need more information about the related information than those described in this ST.
Abbreviated terms are listed to define frequently used terms in the ST.

2. Conformance claim

2.1. CC conformance claim

2.1.1. CC, PP, and security requirement packages

Table 10. Conformance claim

CC		Common Criteria for Information Technology Security Evaluation, CC:2022, Revision 1 - Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model, CC:2022 R1 (CCMB-2022-11-001, 2022. 11.) - Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, CC:2022 R1 (CCMB-2022-11-002, 2022.11.) - Common Criteria for Information Technology Security Evaluation. Part 3: Security Assurance Components, CC:2022 R1 (CCMB-2022-11-003, 2022.11.) - Common Criteria for Information Technology Security Evaluation. Part 4: Framework for the specification of evaluation methods and activities, CC:2022 R1 (CCMB-2022-11-004, 2022.11.) - Common Criteria for Information Technology Security Evaluation. Part 5: Pre-defined packages of security requirements, CC:2022 R1 (CCMB-2022-11-005, 2022.11.) Errata and Interpretation for CC:2022 (Release 1) and CEM:2022 (Release 1), CCMB-2024-07-002 Version 1.1, July 2024
PP		Korean National Protection Profile for Signgle Sign-On V3.1 (2025. 6. 27.)
Conformance claim	**Part 2 Security functional components**	Extended : FIA_IMA.1, FIA_SOS.3, FMT_PWD.1, FPT_PST.1
	Part 3 Security assurance components	Conformant
	Package	Augmented : EAL1 augmented (ATE_FUN.1)

2.1.2. PP conformance statement

This security objective adheres to the principle of “Strict Compliance to Protection Profile”.

2.1.3. PP synthesis conformance claim

The Protection Profile that this security objective complies with is the “National Integrated Authentication Protection Profile V3.1”, and no additional composite Protection Profiles are included.

2.1.4. PP conformance claim

This Security Target strictly complies with the National Integrated Authentication Protection Profile V3.1.
- PP Title and Version: National Integrated Authentication Protection Profile V3.1
- Certification Number: KECS-PP-1348-2025
- Publication Date: June, 27, 2025
- Evaluation Assurance Level: EAL1+
- Conformance Type: Strict Compliance to Protection Profile

2.1.5. Package conformance claim

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

2.1.6. Conformance claim rationale

This Security Target adopts the same TOE type, security problem definition, security objectives, and security assurance requirements as those defined in the Protection Profile, thereby demonstrating strict compliance with the “National Integrated Authentication Protection Profile V3.1”.

[Conformance claim rationale]

Additional OEs have been added compared to the security objectives for the operational environment defined in the conforming PP:
- OE.SECURE_DBMS: Added OE in place of the conditionally required SFR FAU_STG.2 from the PP
- OE.TRUSTED_TIMESTAMP: Added OE in place of the optional SFR FPT_STM.1 from the PP
- OE.MANUAL_RECOVERY: Added OE in place of the conditionally required optional SFR FPT_RCV.1 from the PP
- OE.SECURE_ADMIN_ACCESS: Added OE in place of the conditionally required SFR FTP_TRP.1 from the PP, as the TOE provides this function using the operating environment

Some OEs defined in the conforming PP have been excluded:
- OE.AUTH_SYSTEM_SECURITY: Not added because the TOE does not use an external authentication system

2.2. Reference to assessment methods/activities

The security warranty component packages that this security objective complies with use the evaluation methods and evaluation activities defined in the following table, which includes all packages included in the protection profile.

Table 11. Security assurance requirements

Security Assurance Class	Security assurance component	
Security Target Evaluation	ASE_INT.1	ST introduction
	ASE_CCL.1	Conformance claims
	ASE_OBJ.1	Security objectives for the operational
	ASE_ECD.1	Extended components definition
	ASE_REQ.1	Direct rationale stated security requirements
	ASE_TSS.1	TOE summary specification
Development	ADV_FSP.1	Basic functional specification
Manual	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
Life cycle support	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM coverage
Testing	ATE_FUN.1	Functional testing
	ATE_IND.1	Independent testing - Conformance
Vulnerability survey	AVA_VAN.1	Vulnerability survey

3. Security problem definition

The security problem definition defines the threats, organizational security policies, and assumptions that the TOE and TOE operational environment are intended to handle.

3.1. Assets

The basic assets protected by Single Sign-On are as follows.
- Internal IT resources and services interacting with Single Sign-On
- Important data related to the TOE itself and TOE operation (e.g. TSF data)

3.2. Threats

Threat agents are IT entities and human users that cause harm to assets through unauthorized access or abnormal methods, and can generate various threats as follows. At this time, threat agents to the TOE have a basic level of expertise, resources, and motivation.

3.2.1. Unauthorized access

Table 12. Threats-Unauthorized access

Threat	Specification
T.SESSION_HIJACK	Threat agents can access user screens that are left unattended and logged in, or take advantage of user sessions that are not properly terminated while logged out to steal user authorization.
T.RETRY_AUTH_ATTEMPT	Using information gained from retrying authentication attempts, threat agents can successfully authenticate and then impersonate an authorized user to access the TOE.
T.IMPERSONATION	Threat agents can access the TOE by impersonating an authorized user, TOE components, etc.
T.REPLAY	Threat agents can find out and copy the authentication information, and replay it to access the TOE.
T.WEAK_PASSWORD	Threat agents can access the TOE by obtaining poorly managed passwords such as using the default values for passwords and then impersonating an authorized administrator. If low-level password rules are applied, threat agents can access the TOE by impersonating an authorized administrator.

3.2.2. Information leak

Table 13. Threats-Information leakage

Threat	Specification
T.STORED_DATA_LEAKAGE	Threat agents can leak important data (e.g. cryptographic keys, TOE settings, etc.) stored inside the TOE or in external entities (e.g. DBMS) that interact with the TOE in an unauthorized manner.
T.TRANSMISSION_DATA_DAMAGE	Threat agents can leak or modify transmission data between TOE components and with external IT entities in an unauthorized manner.
T.WEAK_CRYPTO_PROTOCOLS	Threat agents can analyze traffic that uses weak cryptographic communication protocols or low cryptographic strength to infer crypto key information or find out the content of communication ciphertext.

3.2.3. TOE functionality compromise

Table 14. Threats-TOE functionality compromise

Threat	Specification
T.TSF_COMPROMISE	Threat agents can compromise the TSF through unauthorized access, etc. to cause malfunction of the TOE functions or disable the TOE functions.

3.3. Organizational security policy

Table 15. OSP

Policy	Specification
P.AUDIT	To track accountability for security-related actions, security-related events shall be recorded and maintained, and the recorded data shall be reviewed. In addition, the available space on the disk for storing audit data shall be regularly checked to prevent the loss of audit data, and to protect the stored audit data from unauthorized modification or deletion.
P.SECURE_OPERATION	Management means must be provided so that administrators can securely set up the TOE to comply with the organization's Single Sign-On security policy and operate it accurately according to the TOE operation manual.
P.CRYPTO_STRENGTH	Organizations shall apply encryption measures for storage and transmission of important data, such as passwords for user authentication, and use secure cryptographic algorithms.

3.4. Assumptions

It is assumed that the following conditions exist in the TOE operational environment that accepts this ST.

Table 16. Assumptions

Assumption	Description
A.PHYSICAL_PROTECTION	The place where SSO agent and SSO server among the TOE components are installed and operated shall be equipped with access control and protection facilities so that only authorized administrator can access.
A.TRUSTED_ADMIN	The authorized administrator of the TOE is non-malicious, has been appropriately trained for the TOE management functions, and accurately fulfills their duties in accordance with administrator guidelines.
A.OPERATION_SYSTEM_REINFORCEMENT	The reliability and security of the operating system shall be ensured by reinforcing the latest vulnerabilities in the operating system on which the TOE is installed and operated.
A.SECURE_DEVELOPMENT	The developer who uses the TOE to interoperate with the user identification and authentication function in the operational environment of the business system shall ensure that the security functions of the TOE are securely applied in accordance with the requirements of the manual provided with the TOE.
A.SECURE_DBMS	Audit records, including audit trail stored in the DBMS and other components interacting with the TOE, must be protected against unauthorized deletion or modification.
A.TRUSTED_TIMESTAMP	The TOE must use a trusted timestamp provided by the TOE’s operational environment to accurately record security-related events.

A.MANUAL_RECOVERY	The administrator operation manual shall describe procedural manual recovery methods, such as TOE reinstallation, to enable administrators to recover information (e.g., configuration values, libraries) of the TOE agent in case it is tampered with.
A.SECURE_ADMIN_ACCESS	The TOE shall use a secure channel (SSL) to ensure the confidentiality and integrity of communications between the administrator's PC web browser and the user web server.

4. Security objectives

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

4.1. Security objectives for the operational environment

Table 17. TOE Security objectives for the operational environment

TOE Security objective	Specification
OE.LOG_BACKUP	The authorized administrator shall periodically check a spare space of audit data storage in case of the audit data loss, and carry out the audit data backup (e.g., external log server or separate storage device) to prevent audit data loss.
OE.PHYSICAL_CONTROL	The place where SSO agent and SSO server among the TOE components are installed and operated shall be equipped with access control and protection facilities so that only authorized administrators can access.
OE.TRUSTED_ADMIN	The authorized administrator of the TOE shall be non-malicious, has been appropriately trained for the TOE management functions, and accurately fulfills their duties in accordance with administrator guidelines.
OE.OPERATION_SYSTEM_REINFORCEMENT	The authorized administrators of the TOE shall reinforce the operating system where the TOE is installed and operated to address the latest vulnerabilities, thereby ensuring the reliability and security of the operating system.
OE.SECURE_DEVELOPMENT	The developer who uses the TOE to interoperate with the user identification and authentication function in the operational environment of the business system shall ensure that the security functions of the TOE are securely applied in accordance with the requirements of the manual provided with the TOE.
OE.SECURE_DBMS	The DBMS interacting with the TOE shall protect audit records that store audit trail, against unauthorized deletion or modification.
OE.TRUSTED_TIMESTAMP	The TOE shall accurately record security-related events using a trusted timestamp provided by the TOE operating environment.
OE.MANUAL_RECOVERY	The administrator operation manual shall describe procedural manual recovery methods, such as reinstallation of the TOE, to allow the administrator to recover in the event that the TOE agent's information (e.g., configuration values, libraries) is tampered with.
OE.SECURE_ADMIN_ACCESS	The TOE shall use a secure channel to ensure the confidentiality and integrity of communications between the administrator's PC web browser and the user web server.

4.2. Security objectives rationale

Table 18. Security problem definition and corresponding security objectives for the operating environment.

	OE.LOG_BACKUP	OE. PHYSICAL_CONTROL	OE.TRUSTED_ADMIN	OE.SECURE_DEVELOPMENT	OE.OPERATION_SYSTEM_REINFORCEMENT	OE.SECURE_DBMS	OE.TRUSTED_TIMESTAMP	OE.MANUAL_RECOVERY	OE.SECURE_ADMIN_ACCESS
P.AUDIT	O								
P.SECURE_OPERATION			O						
A.PHYSICAL_PROTECTION		O							
A.TRUSTED_ADMIN	O		O						
A.SECURE_DEVELOPMENT				O					
A.OPERATION_SYSTEM_REINFORCEMENT					O				
A.SECURE_DBMS						O			
A.TRUSTED_TIMESTAMP							O		
A.MANUAL_RECOVERY								O	
A.SECURE_ADMIN_ACCESS									O

4.2.1. Security objectives rationale for operational environment

Table 19. Operational environment security objectives rationale

P.AUDIT	OE.LOG_BACKUP

P.AUDIT is performed by **OE.LOG_BACKUP**.
OE.LOG_BACKUP ensures that regular audit data storage space is checked by the administrator as well as the TOE function, and regular log backups or log transmission to an external log server re performed to prevent log records from being lost.

P.SECURE_OPERATION	OE.TRUSTED_ADMIN

P.SECURE_OPERATION is performed by **OE.TRUSTED_ADMIN**.
OE.TRUSTED_ADMIN ensures that the administrator operates TOE accurately in accordance with the organization’s Single Sign-On policy and operating manual.

A.PHYSICAL_PROTECTION	OE.PHYSICAL_PROTECTION

A.PHYSICAL_PROTECTION is supported by **OE.PHYSICAL_PROTECTION**.
OE.PHYSICAL_PROTECTION places the SSO server and the server with the SSO agent in a place equipped with protective equipment and controls access to ensure that only authorized administrators can enter.

A.TRUSTED_ADMIN	OE.TRUSTED_ADMIN, OE.LOG_BACKUP

A.TRUSTED_ADMIN is supported by **OE.TRUSTED_ADMIN**, **OE.LOG_BACKUP**.

OE.TRUSTED_ADMIN has no malicious intent, are properly trained in TOE management functions, and ensure that they perform their duties accurately according to administrator guidelines.
OE.LOG_BACKUP ensures that the authorized administrator periodically checks a spare space of audit data storage in case of the audit data loss, and carries out the audit data backup (external log server or separate storage device, etc.) to prevent audit data loss.

A.SECURE_DEVELOPMENT	OE.SECURE_DEVELOPMENT

A.SECURE_DEVELOPMENT is supported by **OE.SECURE_DEVELOPMENT**.
OE.SECURE_DEVELOPMENT ensures that developers who use the TOE to link user identification and authentication functions in the operating environment of the business system comply with the requirements of the operational user guidance provided with the TOE so that the security functions of the TOE can be applied safely.

A.OPERATION_SYSTEM_REINFORCEMENT	OE.OPERATION_SYSTEM_REINFORCEMENT

A.OPERATION_SYSTEM_REINFORCEMENT is supported by **E.OPERATION_SYSTEM_REINFORCEMENT**.
OE.OPERATION_SYSTEM_REINFORCEMENT ensures the reliability and safety of the operating system by reinforcing the latest vulnerabilities of the operating system in which the TOE is installed and operated.

A.SECURE_DBMS	OE.SECURE_DBMS

A.SECURE_DBMS is supported by **OE.SECURE_DBMS**.
OE.SECURE_DBMS protects audit records in which audit trail is stored, such as those stored in the DBMS interacting with the TOE, against unauthorized deletion or modification.

A.TRUSTED_TIMESTAMP	OE.TRUSTED_TIMESTAMP

A.TRUSTED_TIMESTAMP is supported by **OE.TRUSTED_TIMESTAMP**.
OE.TRUSTED_TIMESTAMP accurately records security-related events using a trusted timestamp provided by the TOE operating environment.

A.MANUAL_RECOVERY	OE.MANUAL_RECOVERY

A.MANUAL_RECOVERY is supported by **OE.MANUAL_RECOVERY**.
OE.MANUAL_RECOVERY ensures that the administrator can recover tampered information (e.g., configuration values, libraries) by describing procedural manual recovery methods, such as TOE reinstallation, in the administrator operation manual.

A.SECURE_ADMIN_ACCESS	OE.SECURE_ADMIN_ACCESS

A.SECURE_ADMIN_ACCESS is supported by **OE.SECURE_ADMIN_ACCESS**.
OE.SECURE_ADMIN_ACCESS ensures the confidentiality and integrity of communications between the administrator’s PC web browser and the user web server by using a secure channel.

5. Extended components definition

5.1. Identification and authentication

5.1.1. TOE Internal mutual authentication

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

Component leveling

FIA_IMA TOE Internal mutual authentication	1

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

Management: FIA_IMA.1
There are no management activities foreseen.

Audit: FIA_IMA.1
The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP, PP-Module, functional package or ST:
a) Minimal: Success and failure of mutual authentication

5.1.1.1. FIA_IMA.1 TOE Internal mutual authentication

Hierarchical to	No other components.
Dependencies	No dependencies.
FIA_IMA.1.1	The TSF shall perform mutual authentication between [assignment: different parts of TOE] using the [assignment: authentication protocol] that meets the following [assignment: list of standards].

5.1.2. Specification of Secrets

Family Behaviour

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

Component leveling

The specification of secrets family in CC Part 2 is composed of 2 components. It is now composed of three components, since this PP adds one more component as below.
※ The description on two components included in CC Part 2 is omitted.
FIA_SOS.3 Destruction of secrets requires, that the secret information be destroyed according to the specified destruction method, which can be based on the assigned standard.

Management: FIA_SOS.3

There are no management activities foreseen.

Audit: FIA_SOS.3

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP, PP-Module, functional package or ST:
a) Minimal : Success and failure of the activity

5.1.2.1. FOA_SOS.3 Destruction of Secrets

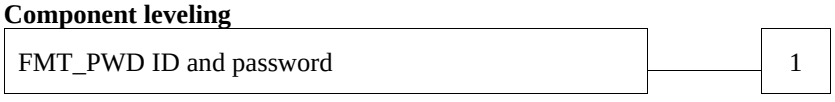
Hierarchical to	No other components.
Dependencies	FIA_SOS.2 TSF Generation of secrets
FIA_SOS.3.1	The TSF shall destroy secrets in accordance with a specified secrets destruction method [assignment: secret destruction method] that meets the following: [assignment: list of standards].

5.2. Security Management

5.2.1. ID and password

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

Component leveling



FMT_PWD.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 should be auditable if FAU_GEN Security audit data generation is included in the PP, PP-Module, functional package or ST:
a) Minimal: All changes of the password

5.2.1.1. FMT_PWD.1 Management of ID and password

Hierarchical to	No other components.
Dependencies	FMT_SMF.1 Specification of management functions FMT_SMR.1 Security roles
FMT_PWD.1.1	The TSF shall restrict the ability to manage the password of [assignment: list of functions] to [assignment: the authorized identified roles]. 1. [assignment: password combination rules and/or length] 2. [assignment: other management such as management of special characters unusable for password, etc.]
FMT_PWD.1.2	The TSF shall restrict the ability to manage the ID of [assignment: list of functions] to [assignment: the authorized identified roles]. 1. [assignment: ID combination rules and/or length] 2. [assignment: other management such as management of special characters unusable for ID, etc.]
FMT_PWD.1.3	The TSF shall provide the capability for [selection, choose one of: setting ID and password when installing, setting password when installing, changing the ID and password when the authorized administrator accesses for the first time, changing the password when the authorized administrator accesses for the first time].

5.3. Protection of the TSF

5.3.1. Protection of stored TSF data

Family Behaviour

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

Component leveling

FPT_PST Protection of stored TSF data	1

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

Management: FPT_PST.1

There are no management activities foreseen.

Audit: FPT_PST.1

There are no auditable events foreseen.

5.3.1.1. FPT_PST.1 Basic protection of stored TSF data

Hierarchical to	No other components.
Dependencies	No dependencies.
FPT_PST.1.1	The TSF shall protect [assignment: TSF data] stored in containers controlled by the TSF from the unauthorized [selection: disclosure, modification].

6. Security requirements

The security requirements specify security functional requirements and assurance requirements that must be satisfied by the TOE that claims conformance to this ST.
The security functional requirements included in this ST are derived from CC Part 2 and Chapter 5 Extended Components Definition.

6.1. Security functional requirements

All subjects, objects, operations, security attributes, and external entities used in the security functional requirements of this Security Target are defined as follows:

Table 20. Subjec, Object, Operationm Security attribute and External Entity definitions

Classification	Definition
Subject	Active entity in the TOE that performs operations on objects
Object	Passive entity in the TOE containing or receiving information and on which subjects perform operations
Operation	Specific type of action performed by a subject on an object
Security attribute	Attributes of subjects, users, objects, information, sessions, or resources that are used to define a Security Functional Requirement (SFR) (3.78) and are utilized in the enforcement of the SFR.
External Entity	Human or IT entity possibly interacting with the TOE from outside of the TOE boundary

All mandatory, conditional, and optional SFRs used in the security requirements of this Security Target are as follows.

Table 21. Security functional requirements

Functional class	Security functional components		Requirement Status	Compliance Status
Security Audit (FAU)	FAU_ARP.1	Security alarms	O	O
	FAU_GEN.1	Audit Data Generation	O	O
	FAU_SAA.1	Potential Violation Analysis	O	O
	FAU_SAR.1	Audit Review	O	O
	FAU_SAR.3	Selectable audit review	O	O
	FAU_STG.1	Audit data storage location	O	O
	FAU_STG.2	Protected Audit data storage	Conditional	X
	FAU_STG.4	Action in case of Possible Audit Data Loss	Conditional	O
	FAU_STG.5	Prevention of Audit Data Loss	Conditional	O
Cryptographic Support (FCS)	FCS_CKM.1	Cryptographic Key Generation	O	O
	FCS_CKM.2	Cryptographic Key Distribution	Optional	O

	FCS_CKM.5	Cryptographic Key Derivation	Conditional	O
	FCS_CKM.6	Timing and event of Cryptographic Key Destruction	O	O
	FCS_COP.1(1)	Cryptographic Operation (ARIA)	O	O
	FCS_COP.1(2)	Cryptographic Operation (HASH)	O	O
	FCS_COP.1(3)	Cryptographic Operation (HMAC)	O	O
	FCS_COP.1(4)	Cryptographic Operation (RSAES)	O	O
	FCS_RBG.1	Random Bit Generation (RBG)	O	O
	FCS_RBG.3	Random Bit Generation (Internal Seeding - Single Source)	Conditional	O
	FCS_RBG.4	Random Bit Generation (Internal Seeding - Multiple Source)	Conditional	X
	FCS_RBG.5	Random Bit Generation (Entropy Source Composition)	Conditional	X
Identification and Authentication (FIA)	FIA_AFL.1(1)	Authentication Failure Handling (General User)013	O	O
	FIA_AFL.1(2)	Authentication Failure Handling (Authorized Administrator)	O	O
	FIA_IMA.1(Extended)	TOE Internal mutual authentication	O	O
	FIA_SOS.1	Verification of secrets	O	O
	FIA_SOS.2	Generation of secrets	O	O
	FIA_SOS.3(Extended)	Destruction of secrets	O	O
	FIA_UAU.2	User Authentication Before Any Action	O	O
	FIA_UAU.4(1)	Single-use authentication mechanisms (general user login information)	O	O
	FIA_UAU.4(2)	Single-use authentication mechanisms (general user authentication tokens)	O	O
	FIA_UAU.7	Protected authentication feedback	O	O
	FIA_UID.2	User Identification Before Any Action	O	O
Security Management	FMT_MOF.1	Management of security functions behaviour	O	O

(FMT)	FMT_MTD.1	Management of TSF data	O	O
	FMT_PWD.1(Extended)	Management of ID and password	O	O
	FMT_SMF.1	Specification of management functions	O	O
	FMT_SMR.1	Security roles	O	O
TSF Protection (FPT)	FPT_FLS.1	Failure with preservation of secure state	O	O
	FPT_ITT.1	Basic internal TSF data transfer protection	O	O
	FPT_LEE.1(Extended)	External entity Association – Authentication	Conditional	X
	FPT_PST.1(Extended)	Basic protection of stored TSF data	O	O
	FPT_RCV.1	Manual Recovery	Conditional	X
	FPT_RCV.2	Automatic Recovery	Conditional	X
	FPT_STM.1	Trusted Timestamp	Optional	X
	FPT_TST.1	TSF Self-Testing	O	O
	FPT_TUD.1(Extended)	TSF Security Patch Update	Conditional	X
TOE Access (FTA)	FTA_MCS.2	Per user attribute limitation on multiple concurrent sessions	O	O
	FTA_SSL.1	Session Lock by TSF	Conditional	X
	FTA_SSL.3	TSF-initiated termination	Conditional	O
	FTA_TSE.1(1)	TOE session establishment (administrator)	O	O
	FTA_TSE.1(2)	TOE session establishment (user)	Conditional	X
Trusted Path/Channel (FTP)	FTP_ITC.1	Inter-TSF trusted channel	Conditional	O
	FTP_TRP.1	Secure Path	Conditional	X

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 [list of actions] upon detection of a potential security violation.

Table 22. Response Actions to Security Violations

Potential Security Violation Incident	Response Action

Self-test failure incident as specified in FAU_SAA.1	Send an email to the authorized administrator
Audit event for integrity verification failure as specified in FAU_SAA.1	
Administrator reaching the threshold of authentication attempts, as part of response to threshold reached for user authentication attempts specified in FAU_SAA.1	
Reaching the threshold for audit storage DB saturation monitoring as specified in FAU_SAA.1	

6.1.1.2. FAU_GEN.1 Audit data generation

Hierarchical to	No other components.
Dependencies	FPT_STM.1 Reliable time stamps
FAU_GEN.1.1	The TSF shall be able to generate an audit record of the following auditable events: a) Start-up and shutdown of the audit functions; b) All auditable events for the not specified level of audit; and c) [Refer to the "auditable events" in [Table 23, 24, 25] Audit events, [other specifically defined auditable events]].
FAU_GEN.1.2	The TSF shall record within each audit record at least the following information: a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST [Refer to the contents of "additional audit record" in [Table 23, 24, 25] Audit events, [other audit relevant information]].

Table 23. Audit record generation for integrated authentication

Security functional component	Auditable event	Additional audit record
FIA_SOS.2	Rejection by the TSF of any tested secret	-
FIA_SOS.3(Extended)	Success and failure of the activity(applicable to the destruction of SSO token only)	-

Table 24. Other auditable event

Security functional component	Auditable event	Additional audit record
FCS_CKM.1	Cryptographic key generation failure	-
FCS_COP.1	Cryptographic operation failures (including type of cryptographic operation)	-
FIA_IMA.1	Mutual Authentication Result (Success, Failure)	-
FIA_AFL.1	Response Upon Reaching User Authentication Attempt Limit	-
FIA_SOS.2	SSO Authentication token generation failure	-
FIA_SOS.3	SSO Authentication token destruction success	-
FIA_UAU.2	User login Success or Fail	-
FIA_UAU.4	Response actions upon detection of duplicate login attempts using the same account	-
FMT_MOF.1	All security management actions related to the [assignment: list of functions] specified in FMT_MOF.1.1 (e.g., enabling/disabling security functions, executing security functions, modifying mechanisms of security functions, adding/deleting/modifying conditions or rules that determine the behavior of security functions, and	Modified security attribute data

	adding/modifying/deleting response actions of security functions) shall be supported. ** When the administrative access service is enabled or disabled, audit records shall be generated for all implemented protocols.	
FMT_MTD.1	User registration, modification, and deletion	-
	All password changes	-
	Security management actions related to the "TSF data list" specified in FMT_MTD.1.1 (e.g., changing default values, querying, modifying, deleting, erasing, etc.) ** However, the functions "audit log viewing" and "TOE version information viewing" are excluded. ** Authentication information (e.g., passwords), cryptographic keys, and similar data must not be stored in audit records. ** Sensitive data in audit records (e.g., passwords, resident registration numbers) shall not be recorded, or if recording is unavoidable, must be masked before being generated.	Modified TSF data
	Change in SSO agent registration status	-
FMT_PWD.1	Default account (ID) and password changes	-
FPT_TST.1	Execution of self-tests (Success, Failure)	Failed security functions
	Execution of TOE self-integrity verifications (Success, Failure)	Component(s) that failed the integrity check
	Cryptographic module self-tests result (Success, Failure)	-
FTA_MCS.2	Denial of new sessions due to concurrent session limit	-
FTA_SSL.3	User session termination	-
FTA_TSE.1	Blocking of Management Terminal Access IP	-
Etc	User logout Success or Fail	-

Table 25. Auditable events for SSO agent

Security functional component	Auditable event	Additional audit record
Self-protection	Execution and result of integrity checks	-
Security management	Agent start-up	-

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 in monitoring the audited events and based upon these rules indicate a potential violation of the enforcement of the SFRs.

FAU_SAA.1.2	The TSF shall enforce the following rules for monitoring audited events: a) Accumulation or combination of [self-test failure events specified in FAU_GEN.1, audit events for integrity verification failure, administrator reaching the threshold of authentication attempts as part of the response to user authentication attempt threshold being reached, threshold reached for audit storage DB saturation monitoring] known to indicate 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 capability to read [all the audit data] from the audit records.
FAU_SAR.1.2	The TSF shall provide the audit records in a manner suitable for the **authorized administrator** to interpret the information.

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 ability to apply “Methods of Selection and/or Ordering” of audit data in the table below based on “Criteria with Logical Relations” in the table below.

Table 26. Audit data search criteria

Criteria with Logical Relations		Methods of Selection and/or Ordering
User audit log	①AND: (Period, User ID, Event type) ②OR: Event success/failure ③SORTING CRITERIA Select ascending or descending	AND of “Criteria with Logical Relations” (①,②,③)
Administrator audit log	①AND: (Period, User ID, Event type) ②OR: Event success/failure ③SORTING CRITERIA Select ascending or descending	
System audit log	①AND: (Period, Event type) ②OR: Event success/failure ③SORTING CRITERIA Select ascending or descending	
Configuration change audit log	①AND: (Period, Event type, Changed menu type) ②OR: Event success/failure ③SORTING CRITERIA Select ascending or descending	

6.1.1.6. FAU_STG.1 Audit data storage location

Hierarchical to	No other components.
Dependencies	FAU_GEN.1 Audit data generation FTP_ITC.1 Inter-TSF trusted channel
FAU_STG.1.1	The TSF shall be able to store generated audit data on the [*local DBMS*.]

6.1.1.7. FAU_STG.4 Action in case of possible audit data loss

Hierarchical to No other components.

Dependencies FAU_STG.2 Protected audit data storage

FAU_STG.4.1 The TSF shall perform [send a warning email to the authorized administrator] when the audit data storage exceeds [the database capacity limit designated by the authorized administrator (50, 60, 70, 80, 90% (initial value 80%))].

6.1.1.8. FAU_STG.5 Prevention of audit data loss

Hierarchical to FAU_STG.4 Action in case of possible audit data loss

Dependencies FAU_STG.2 Protected audit trail storage

FAU_STG.5.1 The TSF *shall ignore audited events* and [“send a warning email to the authorized administrator”] if the audit trail is full.

6.1.2. Cryptographic support (FCS)

6.1.2.1. 6.1.2.1. FCS_CKM.1 Cryptographic key generation

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_RBG.1 Random bit generation, or
FCS_RNG.1 Generation of random numbers]
FCS_CKM.6 Timing and event of cryptographic key destruction

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [see “Key Generation Algorithm (Standard)” in the table below] and specified cryptographic key sizes [see “Key Length (bit)” in the table below] that meet the following: [see “Key Generation Algorithm (Standard)” in the table below].

Table 27. Key Generation Details

Generated Key	Key Length (bit)	Key Generation Entity	Key Generation Algorithm (standard)	Time of Generation
DEK	128	SSO Server	HASH_DRBG (SHA-512) [TTAK.KO-12.0331-Part1](2018), [TTAK.KO-12.0331-Part2](2018)	- At initial startup of the SSO server
TSF data integrity verification key	256	SSO Server	HASH_DRBG (SHA-512) [TTAK.KO-12.0331-Part1](2018), [TTAK.KO-12.0331-Part2](2018)	- Upon SSO server startup
Private key	2048	SSO Server	RSAES ([KS X ISO/IEC 18033-2](2017))	- At initial startup of the SSO server
Public key	2048	SSO Server	RSAES ([KS X ISO/IEC 18033-2](2017))	- At initial startup of the SSO server
Session key for mutual authentication	128	SSO Agent	HASH_DRBG (SHA-512) [TTAK.KO-12.0331-Part1](2018), [TTAK.KO-12.0331-Part2](2018)	- Upon SSO agent startup
Session key for encrypted communication	128	SSO Agent	HASH_DRBG (SHA-512) [TTAK.KO-12.0331-Part1](2018), [TTAK.KO-12.0331-Part2](2018)	- Upon successful mutual authentication between SSO server and SSO agent

6.1.2.2. 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 or FCS_CKM.5 Cryptographic key derivation] FCS_CKM.3 Cryptographic key access
FCS_CKM.2.1	The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [the RSAES public key encryption method provided by a validated cryptographic module] that meets the following: [KS X ISO/IEC 18033-2(2017)].

6.1.2.3. 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 event of cryptographic key destruction
FCS_CKM.5.1	The TSF shall derive cryptographic keys [master key] from [password, sort, about the bit length of the master key] in accordance with a specified key derivation algorithm [PBKDF] and specified cryptographic key sizes [128 bit] that meet the following: [[TTAK.KO-12.0334-Part1(2018), TTAK.KO-12.0334-Part2(2018)]].

Table 28. Key Generation Details

Generated Key	Cryptographic Algorithm (standard)	Key Length (bit)	Key Derivation Entity	Key Derivation Algorithm	Time of Derivation
KEK	ARIA/CBC ([KS X 3254](2016), [KS X 1213-1](2019))	128	SSO Server	PBKDF	- At initial startup of the SSO server - as needed

6.1.2.4. FCS_CKM.6 Timing and event of cryptographic key destruction

Hierarchical to	No other components.
Dependencies	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation]
FCS_CKM.6.1	The TSF shall destroy [authentication token, KEK, DEK, HMAC key for integrity verification, private key for mutual authentication, session key for mutual authentication, session key for encrypted communication] when [*immediately after SSO integration is completed, immediately after use, immediately after SSO agent shutdown, immediately after mutual authentication is completed, or upon termination of encrypted communication*].
FCS_CKM.6.2	The TSF shall destroy cryptographic keys and keying material specified by FCS_CKM.6.1 in accordance with a specified cryptographic key destruction method ["0x00" Overwrite 3 Times] that meets the following: [No document].

Table 29. Key Destruction Details

Generated Key	Time of Destruction
KEK	- Upon SSO server shutdown - Immediately after use
DEK	- Immediately after use
TSF data integrity verification key	- Upon SSO server or SSO agent shutdown - Immediately after use
Private key	- Immediately after use

Public key	- N/A
Session key for mutual authentication	- Upon SSO server or SSO agent shutdown - Immediately after use - Upon completion of mutual authentication - Upon termination of encrypted communication
Session key for encrypted communication	- Upon termination of encrypted communication

6.1.2.5. FCS_COP.1(1) Cryptographic operation (ARIA)

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 Timing and event of cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [data communication encryption/decryption and integrity verification between physically separated TOE components, and TSF data encryption] in accordance with a specified cryptographic algorithm. [ARIA/CCM(128 bit), ARIA/CBC(128 bit)] and cryptographic key sizes [128 bit] that meet the following: [KS X 3254](2016), [KS X 1213-1](2019)]

Table 30. Usage of Each Cryptographic Operation-ARIA

Cryptographic Algorithm	Key Used	Usage
ARIA/CBC (128bit)	KEK	Encryption and decryption of DEK, TSF data integrity verification key, and private key for mutual authentication
	DEK	Encryption and decryption of TSF data
ARIA/CCM (128bit)	DEK	Encryption and decryption of authentication tokens
	Session key for mutual authentication	Encryption and decryption of authentication information for mutual authentication
	Session key for encrypted communication	Encryption and decryption of transmitted data

6.1.2.6. FCS_COP.1(2) Cryptographic operation (HASH)

Hierarchical to No other components.

Dependencies [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation, or
FCS_CKM.5 Cryptographic key derivation]
FCS_CKM.6 Timing and event of cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [user and administrator password encryption] in accordance with a specified cryptographic algorithm. [SHA-256] and cryptographic key sizes [No data.] that meet the following: [[KS X ISO/IEC 10118-3:2001](2018)]

Table 31. Usage of Each Cryptographic Operation-HASH

Cryptographic Algorithm	Key Used	Usage
SHA-256	N/A	Hashing of general user and administrator passwords

6.1.2.7. FCS_COP.1(3) Cryptographic operation (HMAC)

Hierarchical to No other components.

Dependencies [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation, or
FCS_CKM.5 Cryptographic key derivation]
FCS_CKM.6 Timing and event of cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [TSF data integrity verification] in accordance with a specified cryptographic algorithm.[HMAC(SHA256)] and cryptographic key sizes [256 bit] that meet the following: [[TTAK.KO-12.0330-Part1](2018), [TTAK.KO-12.0330-Part2](2018)]

Table 32. Usage of Each Cryptographic Operation-HMAC

Cryptographic Algorithm	Key Used	Usage
HMAC-SHA256	TSF data integrity verification key	Extraction and verification of MAC values to verify the integrity of TSF data

6.1.2.8. FCS_COP.1(4) Cryptographic operation (RSAES)

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 Timing and event of cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [key distribution during mutual authentication between the SSO server and the SSO agent] in accordance with a specified cryptographic algorithm [RSAES] and cryptographic key sizes [2048 bit] that meet the following: [KS X ISO/IEC 18033-2]

Table 33. Usage of Each Cryptographic Operation-RSAES

Cryptographic Algorithm	Key Used	Usage
RSAES(2048bit)	DEK	Mutual authentication between the SSO server and the SSO agent

6.1.2.9. FCS_RBG.1 Random bit generation (RBG)

Hierarchical to No other components

Dependencies [FCS_RBG.2 Random bit generation (external seeding), or
FCS_RBG.3 Random bit generation (internal seeding – single source)]
FPT_FLS.1 Failure with preservation of secure state FPT_TST.1 TSF self-testing

FCS_RBG.1.1 The TSF shall perform deterministic random bit generation services using [Hash_DRBG(SHA-512)] in accordance with [TTAK.KO-12.0331-Part1](2018), [TTAK.KO-12.0331-Part2](2018)] after initialization with a seed.

FCS_RBG.1.2 The TSF shall use a [*getSecureRandom()*] for initialized seeding.

FCS_RBG.1.3 The TSF shall update the state of the DRBG by *re-seeating* using a [getSecureRandom()]in the following situations:
o Under the following conditions
— *[When the Reseed Counter value reaches the update interval] condition :*
in accordance with [TTAK.KO-12.0331-Part1 (2018), TTAK.KO-12.0331-Part2 (2018)]

6.1.2.10. FCS_RBG.3 Random bit generation (internal seeding – single source)

Hierarchical to	No other components.
Dependencies	FCS_RBG.1 Random bit generation (RBG)
FCS_RBG.3.1	The TSF shall be able to seed the RBG using a *TSF software-based noise source* [getSecureRandom()] with a minimum of [384] bits of min-entropy.

6.1.3. Identification and authentication (FIA)

6.1.3.1. 6.1.3.1. FIA_AFL.1(1) Authentication failure handling (general user)

Hierarchical to	No other components.
Dependencies	FIA_UAU.1 Timing of authentication
FIA_AFL.1.1	The TSF shall detect when an *administrator-configurable positive integer within [1–5 (default: 5 times)]* unsuccessful authentication attempts occurs related to [general user authentication failure].
FIA_AFL.1.2	When the defined number of unsuccessful authentication attempts has been *met*, the TSF shall perform [administrator-configurable deactivation of the identification and authentication function for [5, 10, 30, 60 minutes] (default: 5 minutes)].

6.1.3.2. FIA_AFL.1(2) Authentication failure handling (administrator)

Hierarchical to	No other components.
Dependencies	FIA_UAU.1 Timing of authentication
FIA_AFL.1.1	The TSF shall detect when an *administrator configurable positive integer within [1–5(default: 5 times)]* unsuccessful authentication attempts occur related to [administrator authentication failure].
FIA_AFL.1.2	When the defined number of unsuccessful authentication attempts has been *met*, the TSF shall [deactivate the identification and authentication functions (default: 5 minutes, configurable from 5 to 60 minutes)].

6.1.3.3. FIA_IMA.1 TOE Internal mutual authentication

Hierarchical to	No other components.
Dependencies	No dependencies.
FIA_IMA.1.1	The TSF shall perform mutual authentication between [SSO Server and SSO Agent] in accordance with a specified [Self-Implemented Authentication Protocol] that meets the following: [None].

6.1.3.4. FIA_SOS.1 Verification of secrets

Hierarchical to	No other components.
Dependencies	No dependencies.
FIA_SOS.1.1	The TSF shall provide a mechanism to verify that secrets meet [Table 26. Password security criteria].

Table 34. Password security criteria

Sub-category	Specification	
Requirements	Allowed Characters	52 English letters (uppercase and lowercase)
		10 numeral digits (0–9)
		Special characters (! @ # $ % ^ & * ( ) - _ = + [{] } | ; : , < . > / ? ` ~)
	Minimum/Maximum Length	Password minimum/maximum length: 9 to 20 characters (configurable from 9 to 20)
		Must include at least one of each: numeral digit, uppercase letter, lowercase letter, and special character (default: 1 of each, configurable from 1 to 2)

Prohibitions	Password must not be the same as the user account (ID)
	Consecutive repetition of the same character or number is not allowed (default: allowed up to 2 times, configurable from 1 to 2)
	Sequential input of 4 or more characters or digits in keyboard order is not allowed
	Reuse of the previously used password is prohibited

6.1.3.5. FIA_SOS.2 TSF Generation of secrets

Hierarchical to No other components.

Dependencies No dependencies.

FIA_SOS.2.1 TSF shall provide a mechanism to generate **an authentication token** that meet [Table 35].

FIA_SOS.2.1 The TSF shall be able to enforce the use of TSF-generated **authentication tokens** for [general user login].

Table 35. Authentication token definitions

Defined acceptance criteria	content
Authentication token Configuration method	Self-configuration method
Authentication token algorithm	ARIA/CCM(128bit)
Authentication token length	44 byte

6.1.3.6. FIA_SOS.3 Destruction of Secrets(Extended)

Hierarchical to No other components.

Dependencies FIA_SOS.2 TSF Generation of secrets

FIA_SOS.3.1 The TSF shall destroy **authentication tokens** in accordance with a specified **authentication token** destruction method [triple overwrite of the variable with '0x00'] that meets the following: [None].

6.1.3.7. FIA_UAU.2 User authentication before any action

Hierarchical to FIA_UAU.1 Timing of authentication

Dependencies FIA_UID.1 Timing of identification

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

6.1.3.8. FIA_UAU.4(1) Single-use authentication mechanisms (administrator and general user login information)

Hierarchical to No other components.

Dependencies No dependencies.

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to [Onetime CSRF token].

Table 36. CSRF Token definitions

Defined acceptance criteria	content
CSRF Token Configuration method	Self-configuration method
CSRF Token length	44 byte

6.1.3.9. FIA_UAU.4(2) Single-use authentication mechanisms (general user authentication tokens)

Hierarchical to No other components.

Dependencies	No dependencies.
FIA_UAU.4.1	The TSF shall prevent reuse of authentication data related to [authentication token].

6.1.3.10. FIA_UAU.7 Protected authentication feedback

Hierarchical to	No other components.
Dependencies	FIA_UAU.1 Timing of authentication
FIA_UAU.7.1	The TSF shall provide only [masking with "*" during administrator/general user identification and authentication] to the user while authentication is in progress.

6.1.3.11. FIA_UID.2 User identification before any action

Hierarchical to	FIA_UID.1 Timing of identification
Dependencies	No dependencies.
FIA_UID.2.1	The TSF shall require each **administrator and user** to be successfully identified before allowing any TSF-mediated actions on behalf of that **administrator and user**.

6.1.4. Security management (FMT)

6.1.4.1. FMT_MOF.1 Management of security functions behaviour

Hierarchical to No other components.

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

FMT_MOF.1.1 The TSF shall restrict the ability to ***conduct management actions*** of the functions [see "Management Actions for Security Functions" in the table below] to [see "Classification of Administrator Authorization" in the table below].

Table 37 Specification of Security Functions Management

Subclass	Management Actions for Security Functions	Classification of Administrator Authorization
FIA_UAU.2 FIA_UID.2	Register, delete, modify users; assign user privileges	Super Administrator, General Administrator
FMT_PWD.1	Configure password composition and length policy	Super Administrator, General Administrator
FIA_AFL.1	Configure allowed number of user authentication failures	Super Administrator, General Administrator
	Configure response method to user authentication failure	Super Administrator, General Administrator
	Configure time before reactivation after authentication function is deactivated	Super Administrator, General Administrator
FTA_TSE.1	Register, delete, and modify IP addresses of management terminals	Super Administrator, General Administrator
FMT_MTD.1	View agent information – status, version, applied security policy	Super Administrator, General Administrator
	Manage agent security policy – configure and distribute policy	Super Administrator, General Administrator
	Configure authentication information for accessing external IT entities	Super Administrator, General Administrator
FPT_TST.1	Request self-test of TOE server	Super Administrator, General Administrator
	Request integrity check of TOE configuration and the TOE itself	Super Administrator, General Administrator
FTA_SSL.3	Configure session timeout for users	Super Administrator, General Administrator
	TOE Version Information Retrieval	Super Administrator, General Administrator, Monitoring Administrator
FTA_MCS.2	Configure number of concurrent user sessions	Super Administrator, General Administrator
FAU_SAR.1	View audit records	Super Administrator, General Administrator, Monitoring Administrator
FAU_STG.4	Configure response settings for audit record loss	Super Administrator, General Administrator
FAU_STG.5		Super Administrator, General Administrator

6.1.4.2. FMT_MTD.1 Management of TSF data

Hierarchical to No other components.

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

FMT_MTD.1.1 The TSF shall restrict the ability to ***manage*** [see "TSF Data Management Actions" in the table below] to [see "Administrator Privilege Classification" in the table below].

Table 38. TSF data management actions as covered by FMT_MTD.1

Security Function Component	TSF Data Management Actions	Classification of Administrator Authorization
FIA_UAU.2 FIA_UID.2	Granting **privileges** to general administrator and monitoring administrator accounts (IDs) Adding, deleting, and modifying general administrator and monitoring administrator **IDs** Modifying **passwords and user information (name, role, email, phone number)** for general administrator and monitoring administrator accounts	Super Administrator
	Adding, deleting, and modifying general user **IDs**	Super Administrator, General Administrator
FMT_PWD.1(Extended)	Configure **password length and combination** policies for administrators and general users	Super Administrator, General Administrator
FMT_PWD.1(Extended) FMT_MTD.1	Modify own **password and personal information (name, role, email, phone number)**	Super Administrator, General Administrator, Monitoring Administrator
	Add or delete **password and personal information (name, email, phone number)** of general users	Super Administrator, General Administrator
FIA_AFL.1(1) FIA_AFL.1(2)	Setting the **allowable number of authentication failures**	Super Administrator, General Administrator
	Configure **the reactivation time after authentication function deactivation**	Super Administrator, General Administrator
FTA_TSE.1	Register, delete, and modify **management terminal IP addresses**	Super Administrator, General Administrator
FMT_MTD.1	Agent inquiry - Required inquiry information: **agent version, security policy applied to the agent, agent operation status** (activated/deactivated), **agent integrity verification result** (success/failure)	Super Administrator, General Administrator
	Managing **agent security policy**	Super Administrator, General Administrator
	Configuring **authentication information for accessing external IT entities (e.g., SMTP)**	Super Administrator, General Administrator
	Change **default passwords used to access internal TOE components (DBMS) or external IT entities**	Super Administrator, General Administrator
	View **identification information of the TOE and TOE components (e.g., server, agent)**	Super Administrator, General Administrator
FAU_STG.4	**Setting audit trail threshold** for notifying administrators of potential audit record loss	Super Administrator, General Administrator
FTA_SSL.3	Configuring **automatic session termination time for users**	Super Administrator, General Administrator
FAU_SAR.1	Viewing **audit records**	Super Administrator, General Administrator, Monitoring Administrator

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

Hierarchical to	No other components.
Dependencies	FMT_SMF.1 Specification of Management Functions FMT_SMR.1 Security roles
FMT_PWD.1.1	The TSF shall restrict the ability to manage the password of [the following function list] to [**Super Administrator, General Administrator**]. 1. [1) Allowed Characters o 52 English letters (uppercase or lowercase) o 10 numeral digits (0–9) o Special characters (! @ # $ % ^ & * ( ) - _ = + [{] } | ; : , < . > / ? ` ~) 2) Minimum/Maximum Length o Password minimum/maximum length: 9 to 20 characters (configurable from 9 to 20) o Must include at least one of each: numeral digit, uppercase letter, lowercase letter, and special character (default: 1 of each, configurable from 1 to 2)] 2. [o Password must not be the same as the user account (ID) o Consecutive repetition of the same character or number is not allowed (default: allowed up to 2 times, configurable from 1 to 2) o Sequential input of 4 or more characters or digits in keyboard order is not allowed o Reuse of the previously used password is prohibited]
FMT_PWD.1.2	The TSF shall restrict the ability to manage the ID of [no function] to [the authorized administrator].
FMT_PWD.1.3	The TSF shall provide the capability for *changing the password when the authorized administrator accesses for the first time*.

6.1.4.4. FMT_SMF.1 Specification of Management Functions

Hierarchical to	No other components.
Dependencies	No dependencies.
FMT_SMF.1.1	The TSF shall be capable of performing the following management functions: [list of management functions to be provided by the TSF]. [TSF function management: Management functions specified in FMT_MOF.1 TSF data management: Management functions specified in FMT_MTD.1 ID and password management: Management functions specified in FMT_PWD.1]

6.1.4.5. FMT_SMR.1 Security roles

Hierarchical to	No other components.
Dependencies	FIA_UID.1 Timing of identification
FMT_SMR.1.1	The TSF shall maintain the roles [super administrator, general administrator, monitoring administrator].
FMT_SMR.1.2	The TSF shall be able to associate users and their roles **defined in FMT_SMR.1.1**.

6.1.5. Protection of the TSF (FPT)

6.1.5.1. FPT_FLS.1 Failure with preservation of secure state

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: [failure of noise source health test].

6.1.5.2. FPT_ITT.1 Basic internal TSF data transfer protection

Hierarchical to No other components.

Dependencies No dependencies.

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

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

Hierarchical to No other components.

Dependencies No dependencies.

FPT_PST.1.1 The TSF shall protect [TSF data listed in the table below] stored in containers controlled by the TSF from unauthorized *disclosure*.

Table 39. TSF Data Requiring Encryption Upon Save

Category	TSF Data Requiring Encryption Upon Save	Remarks
TSF data stored by TOE server	General user password	User passwords for user identification and authentication are stored using the one-way hash algorithm SHA-256 to prevent decryption. - A randomly generated secret value, called a salt, is applied when hashing passwords. - The salt is generated using a random number generator based on the HASH-DRBG (SHA-512) algorithm, with a size of 128 bits. - The iteration count is 1,000. - As a one-way hash algorithm is used, there is no encryption key, and the encrypted password is stored in the database.
	Administrator password	
	DEK, IV, TSF Data Integrity Verification Key (HMAC)	DEK, IV, and the TSF Data Integrity Verification Key (HMAC) are encrypted using the Key Encryption Key (KEK) with ARIA/CBC (128-bit) and stored securely. The encryption key is not hardcoded into the TOE; instead, it is encrypted with the KEK and stored as a file.
	Integrity verification value	The integrity verification values are generated by hashing each library and configuration file that constitutes the TOE using the TSF Data Integrity Verification Key (HMAC) with HMAC (SHA-256), and are stored in the form of (filename)=(hash value).
	public key	The public key is encoded with a proprietary encoding method and stored.
	private key	The private key is encrypted using the Key Encryption Key (KEK) with ARIA/CBC (128-bit) and stored securely. The encryption key is not hardcoded into the TOE; instead, it is encrypted with the KEK and stored as a file.
	KEK salt	The KEK salt is encoded with a proprietary encoding method and stored.
	KEK Integrity Verification Value	The KEK integrity verification value is generated by encrypting the string “tomato” with ARIA/CBC (128-bit) using the Key Encryption Key (KEK), encoding it with a proprietary encoding method, and stored as a file.

		It is used to verify whether the KEK password matches the value previously entered.
	Passwords in WAS logs	Passwords are not recorded in WAS logs.
	Encryption keys in WAS logs	Encryption keys do not remain as plain text in WAS logs.
	DBMS access information	Encrypted using DEK with ARIA/CBC (128-bit) and stored as a file.
	SMTP password	Encrypted using DEK with ARIA/CBC (128-bit) and stored securely.
	Authentication tokens	Authentication tokens are automatically destroyed upon successful validation.

Table 40. TSF Data That Must Be Securely Protected via Encryption, Access Control

Category	TSF Data That Must Be Securely Protected via Encryption, Access Control, etc.	Storage location	Remarks
TSF data stored by TOE server	TOE configuration values (security policies, environment configuration parameters, etc.)	DB	The TOE does not support backing up configuration values as external files.
	Audit records	DB	Audit records can only be viewed when logged in as an authorized administrator.

Table 41. Protection for Cryptographic Key Storage

Category	TSF Data to Be Securely Protected According to the Conditions Described in the Remarks	Remarks
KEK	Key Encryption Key (KEK)	The KEK (Key Encryption Key) is not stored; instead, a KEK password is entered each time the SSO server starts. This password is stored in memory using a proprietary encoding method. Whenever the KEK is needed, it is newly generated by deriving it from the password stored in memory using a PBKDF (Password-Based Key Derivation Function) algorithm. The salt is stored in a file during the initial startup of the SSO server and is loaded from that file during subsequent startups.

Table 42. Protection of TSF Data (Sensitive Information) Stored by the SSO Agent

Category	TSF Data Requiring Encryption Upon Save	Remarks
TSF data stored in files and systems by the SSO agent	TSF data integrity verification key (HMAC)	The SSO agent uses symmetric key encryption when communicating securely with the SSO server.
	Integrity verification value	The integrity verification values are generated by hashing each library and configuration file that constitutes the TOE using the TSF Data Integrity Verification Key (HMAC) with HMAC (SHA-256), and are stored in the form of (filename)=(hash value).

Table 43. Protection of TSF Data (Configuration Values, Audit Data) Stored by the SSO Agent

Category	TSF Data Requiring Encryption Upon Save	Storage location	Remarks
TSF data stored in files and systems by the SSO agent	TOE configuration values (security policies, environment configuration parameters, etc.)	N/A	The TOE does not support backing up configuration values as external files.
	Audit records	Memory	Audit records are encrypted using the communication session key with ARIA/CCM (128-bit) and immediately transmitted to the SSO server through a proprietary encrypted communication channel. If transmission is not possible, up to 1,000 records are stored in the memory of the SSO agent to ensure stability.

Table 44. Protection of Stored TSF Data Related to Authentication Tokens (Cryptographic Keys, Critical Security Parameters)

Category	TSF Data Related to Authentication Tokens	Remarks
Data Encryption Key	DEK	The Data Encryption Key (DEK) is encrypted using the symmetric key encryption algorithm ARIA/CBC (128-bit) with the KEK and stored as an internal file of TOE.

6.1.5.4. FPT_TST.1 TSF testing

Hierarchical to No other components.

Dependencies No dependencies.

FPT_TST.1.1 The TSF shall run a suite of the following self-tests [self-tests of the SSO server and SSO agent, integrity tests, cryptographic module self-tests] *at startup, periodically during normal operation, and upon request by an authorized* ***administrator***, to demonstrate the correct operation of *the TSF*.
The TSF shall run a suite of the following self-tests *at startup, periodically during normal operation, and upon request by an authorized* ***administrator*** to demonstrate the correct operation of *the TSF*: [self-tests of the SSO server and SSO agent, integrity tests, cryptographic module self-tests].

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

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

6.1.6. TOE access (FTA)

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

Hierarchical to FTA_MCS.1 Basic limitation on multiple concurrent sessions

Dependencies FIA_UID.1 Timing of identification

FTA_MCS.2.1 The TSF shall restrict the maximum number of concurrent sessions belonging to the same **administrator and user** according to the following rules: [The maximum number of concurrent sessions for users with the same privileges and for the same user is limited to 1; [For management access sessions of both super administrators and general administrators, they are treated as having the same privileges. Therefore, the maximum number of administrator sessions—excluding the monitoring administrator—is limited to 1]].

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

6.1.6.2. FTA_TSE.1(1) TOE session establishment

Hierarchical to	None
Dependencies	None
FTA_TSE.1.1	The TSF shall be able to deny **the administrator's management access session** establishment based on [access IP, *none*].

6.1.6.3. 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 an interactive session after a [session timeout period configured by an authorized administrator (60 to 600 seconds, default 600 seconds)].

6.1.7. Truested path/channels (FTP)

6.1.7.1. FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to	No other components.
Dependencies	No dependencies.
FTP_ITC.1.1	The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2	The TSF shall permit *trusted IT product* to initiate communication via the trusted channel.
FTP_ITC.1.3	The TSF shall initiate communication via the trusted channel for [transmitting TOE audit data and sending administrator notification emails].

6.2. Security assurance requirements

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

Table 45. Security assurance requirements

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

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 an ST introduction.
Content and presentation elements	
ASE_INT.1.1C	The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description.
ASE_INT.1.2C	The ST reference shall uniquely identify the ST
ASE_INT.1.3C	The TOE reference shall uniquely identify the TOE.
ASE_INT.1.4C	The TOE overview shall summarise the usage and major security features of the TOE.
ASE_INT.1.5C	The TOE overview shall identify the TOE type.
ASE_INT.1.6C	The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE.
ASE_INT.1.7C	For a multi-assurance ST, the TOE overview shall describe the TSF organization in terms of the sub-TSFs defined in the PP-Configuration the ST claims conformance to.
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 claims

Dependencies	ASE_INT.1 ST introduction ASE_ECD.1 Extended components definition ASE_REQ.1 Stated security requirements
Developer action elements	
ASE_CCL.1.1D	The developer shall provide a conformance claim.
ASE_CCL.1.2D	The developer shall provide a conformance claim rationale.
Content and presentation elements	
ASE_CCL.1.1C	The conformance claim shall contain a CC conformance claim that identifies the version of the CC to which the ST and the TOE claim conformance.
ASE_CCL.1.2C	The CC conformance claim shall describe the conformance of the ST to CC Part 2 as either CC Part 2 conformant or CC Part 2 extended.
ASE_CCL.1.3C	The CC conformance claim shall describe the conformance of the ST to CC Part 3 as either CC Part 3 conformant or CC Part 3 extended.
ASE_CCL.1.4C	The CC conformance claim shall be consistent with the extended components definition.
ASE_CCL.1.5C	The conformance claim shall identify all PPs and security requirement packages to which the ST claims conformance.
ASE_CCL.1.6C	The conformance claim shall describe any conformance of the ST to a package as either package-conformant or package-augmented.
ASE_CCL.1.7C	The conformance claim shall describe the conformance of the ST to the PP as PP-conformant.
ASE_CCL.1.8C	The conformance claim rationale shall demonstrate that the TOE type is consistent with the TOE type in the PPs for which conformance is being claimed.
ASE_CCL.1.9C	The conformance claim rationale shall demonstrate that the statement of the security problem definition is consistent with the statement of the security problem definition in the PPs for which conformance is being claimed.
ASE_CCL.1.10C	The conformance claim rationale shall demonstrate that the statement of security objectives is consistent with the statement of security objectives in the PPs for which conformance is being claimed.

ASE_CCL.1.11C	The conformance claim rationale shall demonstrate that the statement of security requirements is consistent with the statement of security requirements in the PPs for which conformance is being claimed.
ASE_CCL.1.12C	The conformance claim for the PP(s) or PP-configuration shall be exact conformance, strict conformance, or demonstrable conformance, or a list of conformance types.
ASE_CCL.1.13C	If the conformance claim identifies a set of evaluation methods and evaluation activities derived from CEM work units to be used for TOE evaluation, this set shall include all those contained in the packages, PPs, or PP-modules of the PP-configuration to which the ST claims conformance, with no additional methods permitted.
Evaluator action elements	
ASE_CCL.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.1.3. ASE_OBJ.1 Security objectives for the operational environment

Dependencies	ASE_SPD.1 Security problem definition
Developer action elements	
ASE_OBJ.1.1D	The developer shall provide a statement of security objectives for the operational environment.
ASE_OBJ.1.2D	The developer shall provide a security objectives rationale for the operational environment.
Content and presentation elements	
ASE_OBJ.1.1C	The statement of security objectives shall describe the security objectives for the operational environment.
ASE_OBJ.1.2C	The security objectives rationale shall trace each security objective for the operational environment back to threats countered by that security objective, OSPs enforced by that security objective, and assumptions upheld by that security objective.
ASE_OBJ.1.3C	The security objectives rationale shall demonstrate that the security objectives for the operational environment uphold all assumptions.
Evaluator action elements	
ASE_OBJ.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.1.4. ASE_ECD.1 Extended components definition

Dependencies	No dependencies.
Developer action elements	

ASE_ECD.1.1D	The developer shall provide a statement of security requirements.
ASE_ECD.1.2D	The developer shall provide an extended components definition.
Content and presentation elements	
ASE_ECD.1.1C	The statement of security requirements shall identify all extended security requirements.
ASE_ECD.1.2C	The extended components definition shall define an extended component for each extended security requirement.
ASE_ECD.1.3C	The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes.
ASE_ECD.1.4C	The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation.
ASE_ECD.1.5C	The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements may 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 may be clearly expressed using existing components.

6.2.1.5. ASE_REQ.1 Direct rationale 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 security requirements rationale.
Content and presentation elements	
ASE_REQ.1.1C	The statement of security requirements shall describe the SFRs and the SARs.
ASE_REQ.1.2C	For a single-assurance ST, the statement of security requirements shall define the global set of SARs that apply to the entire TOE. The sets of SARs shall be consistent with the PPs or PP-Configuration to which the ST claims conformance.
ASE_REQ.1.3C	For a multi-assurance ST, the statement of security requirements shall define the global set of SARs that apply to the entire TOE and the sets of SARs that apply to each sub-TSF. The sets of SARs shall be consistent with the multi- assurance PP-Configuration to 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 the 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 either be satisfied, or the security requirements rationale shall justify the dependency not being satisfied.
ASE_REQ.1.8C	The security requirements rationale shall demonstrate that the SFRs (in conjunction with the security objectives for the environment) counter all threats for the TOE.
ASE_REQ.1.9C	The security requirements rationale shall demonstrate that the SFRs (in conjunction with the security objectives for the environment) enforce all OSPs.
ASE_REQ.1.10C	The security requirements rationale shall explain why the SARs were chosen.
ASE_REQ.1.11C	The statement of security requirements shall be internally consistent.
ASE_REQ.1.12C	If the ST defines sets of SARs that expand the sets of SARs of the PPs or PP-Configuration it claims conformance to, the security requirements rationale shall include an assurance rationale that justifies the consistency of the extension and provides a rationale for the disposition of any Evaluation methods and Evaluation activities identified in the conformance statement that are affected by the extension of the sets of SARs.
Evaluator action elements	
ASE_REQ.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.1.6. ASE_TSS.1 TOE summary specification

Dependencies	ASE_INT.1 ST introduction ASE_REQ.1 Direct rationale 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 Basic functional specification

Dependencies	No dependencies.
Developer action elements	
ADV_FSP.1.1D	The developer shall provide a functional specification.
ADV_FSP.1.2D	The developer shall provide a tracing from the functional specification to the SFRs.
Content and presentation elements	
ADV_FSP.1.1C	The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.
ADV_FSP.1.2C	The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.
ADV_FSP.1.3C	The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.
ADV_FSP.1.4C	The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.
Evaluator action elements	
ADV_FSP.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_FSP.1.2E	The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

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 shall be controlled in a secure processing environment, including appropriate warnings.
AGD_OPE.1.2C	The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.
AGD_OPE.1.3C	The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.
AGD_OPE.1.4C	The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.
AGD_OPE.1.5C	The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.
AGD_OPE.1.6C	The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST.
AGD_OPE.1.7C	The operational user guidance shall be clear and reasonable.
Evaluator action elements	
AGD_OPE.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

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_PRE1.1C	The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.
AGD_PRE1.2C	The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.
Evaluator action elements	
AGD_PRE.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AGD_PRE.1.2E	The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

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 for the TOE.
Content and presentation elements	
ALC_CMC.1.1C	The TOE shall be labelled with its unique reference.

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

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.1.2C	The configuration list shall uniquely identify the configuration items
Evaluator Requirements	
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 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 document must include a test plan, expected test results, and actual test results.

ATE_FUN.1.2C	The test plan must identify the test items to be performed and describe the scenarios for each test. These scenarios must include order dependencies for other test results.
ATE_FUN.1.3C	The expected test results must present the outcomes expected from successful execution of the test.
ATE_FUN.1.4C	The actual test results must be consistent with the expected test results.
Evaluator action elements	
ATE_FUN.1.1E	The evaluator must verify that the provided information satisfies all evidence requirements.

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 for testing.
Content and presentation elements	
ATE_IND.1.1C	The TOE shall be suitable for testing.
Evaluator action elements	
ATE_IND.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ATE_IND.1.2E	The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

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 for testing.
Content and presentation elements	
AVA_VAN.1.1C	The TOE shall be suitable for testing.
Evaluator action elements	
AVA_VAN.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_VAN.1.2E	The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.
AVA_VAN.1.3E	The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6.3. Security requirements rationale

6.3.1. Security functional requirements rationale

Table 46. correspondence with the 'security problem definition' and the 'security functional requirements'

SFRs	T.SESSION_HIJACK	T.RETRY_AUTH_ATTEMPT	T.IMPERSONATION	T.REPLAY	T.WEAK_PASSWORD	T.STORED_DATA_LEAKAGE	T.TRANSMISSION_DATA_DAMAGE	T.WEAK_CRYPTO_PROTOCOLS	T.TSF_COMPROMISE	P.AUDIT	P.SECURE_OPERATION	P.CRYPTO_STRENGTH
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.4										O		
FAU_STG.5										O		
FCS_CKM.1						O	O	O				O
FCS_CKM.2						O	O	O				O
FCS_CKM.5						O	O	O				O
FCS_CKM.6						O	O	O				O
FCS_COP.1(1)						O	O	O				O
FCS_COP.1(2)						O	O	O				O
FCS_COP.1(3)						O	O	O				O
FCS_COP.1(4)						O	O	O				O
FCS_RBG.1						O	O	O				O
FCS_RBG.3						O	O	O				O
FIA_AFL.1(1)		O	O						O			
FIA_AFL.1(2)		O	O						O			
FIA_IMA.1			O									
FIA_SOS.1					O							
FIA_SOS.2				O								
FIA_SOS.3	O											
FIA_UAU.2			O						O			
FIA_UAU.4(1)			O	O					O			
FIA_UAU.4(2)			O						O			
FIA_UAU.7			O		O				O			
FIA_UID.2			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
FPT_ITT.1							O					
FPT_PST.1						O						

FPT_TST.1						O	O	O	O			O
FTA_MCS.2	O											
FTA_SSL.3	O											
FTA_TSE.1(1)	O											
FTA_TSE.1(2)	O											
FTP_ITC.1							O					

Table 47. Security Functions Addressing Threats

Threats	SFRs	Description
T.SESSION_HIJACK	FIA_SOS.3	FIA_SOS.3 responds to T.SESSION_HIJACK by ensuring safe destruction of the authentication token when the TOE session ends.
	FTA_MCS.2	FTA_MCS.2 responds to T.SESSION_HIJACK by restricting concurrent access to the TOE with the same user account or same privileges.
	FTA_SSL.3	FTA_SSL.3 repond to T.SESSION_HIJACK by ensuring session locking or session termination for interactive sessions after a period of inactivity by authorized users.
	FTA_TSE.1(1)	FTA_TSE.1(1) respond to T.SESSION_HIJACK by ensuring that it determines whether to establish an authorized user access session based on IP.
T.RETRY_AUTH_ATTEMPT	FIA_AFL.1(1) FIA_AFL.1(2)	FIA_AFL.1(1), FIA_AFL1(2) responds to T.RETRY_AUTH_ATTEMPT by defining the number of failed authentication attempts by authorized users and ensuring the ability to take responsive action when the defined number is reached.
T.IMPERSONATION	FIA_AFL.1(1) FIA_AFL.1(2)	FIA_AFL.1(1), FIA_AFL1(2) responds to T.IMPERSONATION by defining the number of failed authentication attempts by authorized users and ensuring the ability to take responsive action when the defined number is reached.
	FIA_IMA.1	FIA_IMA.1 responds to T.IMPERSONATION by ensuring that mutual authentication is conducted between TOE components.
	FIA_SOS.2 FIA_SOS.3 FIA_UAU.2 FIA_UAU.4(1) FIA_UAU.4(2)	FIA_SOS.2, FIA_SOS.3, FIA_UAU.2, FIA_UAU.4(1), FIA_UAU.4(2) respond to T.IMPERSONATION by ensuring that users attempting to access the TOE are successfully authenticated.
	FIA_UAU.7	FIA_UAU.7 responds to T.IMPERSONATION by ensuring that only masked values will be output or no display to users during authentication and not providing feedback on the reason for failure in case of authentication failure.
	FIA_UID.2	FIA_UID.2 responds to T.IMPERSONATION by ensuring that users attempting to access the TOE are successfully identified.

T.REPLAY	FIA_SOS.2	FIA_SOS.2 responds to T.REPLAY by ensuring that authentication tokens are not reused when generating authentication tokens.
	FIA_UAU.4(1) FIA_UAU.4(2)	FIA_UAU.4(1), FIA_UAU.4(2) responds to T.REPLAY by ensuring the ability to prevent reuse of authentication data.
T.WEAK_PASSWORD	FIA_UAU.7	FIA_UAU.7 responds to T.WEAK_PASSWORD by ensuring that only masked values will be output or no display to users during authentication.
	FIA_SOS.1	FIA_SOS.1 responds to T.WEAK_PASSWORD by verifying that password complexity rules are satisfied.
	FMT_PWD.1	FMT_PWD.1 responds to T.WEAK_PASSWORD by ensuring the ability to force a change of the default password when the authorized administrator first connects
T.STORED_DATA_LEAKAGE	FCS_CKM.1 FCS_CKM.2 FCS_CKM.5 FCS_RBG.1 FCS_RBG.3 FPT_FLS.1 FPT_TST.1	FCS_CKM.1, FCS_CKM.2, FCS_CKM.5, FCS_RBG.1, FCS_RBG.3, FCS_RBG.5, FPT_FLS.1, FPT_TST.1 respond to T.STORED_DATA_LEAKAGE by ensuring that a cryptographic key is created and distributed according to a secure cryptographic algorithm and key length when encrypting stored data.
	FCS_CKM.6	FCS_CKM.6 responds to T.STORED_DATA_LEAKAGE by ensuring that the cryptographic keys and their related information are destroyed according to the specified cryptographic key destruction method upon completion of storage data encryption.
	FCS_COP.1(1) FCS_COP.1(2) FCS_COP.1(3) FCS_COP.1(4)	FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4) responds to T.STORED_DATA_LEAKAGE by ensuring that cryptographic operations are performed according to the specified secure algorithm and specified cryptographic key length when encrypting stored data.
	FPT_PST.1	FPT_PST.1 responds to T.STORED_DATA_LEAKAGE by ensuring that the stored TSF data is protected from being leaked by means of encryption, access control, etc.
T.TRANSMISSION_DATA_DAMAGE	FCS_CKM.1 FCS_CKM.2 FCS_CKM.5 FCS_RBG.1 FCS_RBG.3 FPT_FLS.1 FPT_TST.1	FCS_CKM.1, FCS_CKM.2, FCS_CKM.5, FCS_RBG.1, FCS_RBG.3, FCS_RBG.5, FPT_FLS.1, and FPT_TST.1 respond to T.TRANSMISSION_DATA_DAMAGE by ensuring that a cryptographic key is created and distributed according to a secure cryptographic algorithm and key length during cryptographic communication.
	FCS_CKM.6	FCS_CKM.6 responds to T.TRANSMISSION_DATA_DAMAGE by ensuring that cryptographic keys and their related information are destroyed according to the

		specified cryptographic key destruction method at the end of cryptographic communication.
	FCS_COP.1(1) FCS_COP.1(2) FCS_COP.1(3) FCS_COP.1(4)	FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4) responds to T.TRANSMISSION_DATA_DAMAGE by ensuring that cryptographic operations are performed according to the specified secure algorithm and specified cryptographic key length during cryptographic communication.
	FPT_ITT.1	FPT_ITT.1 responds to T.TRANSMISSION_DATA_DAMAGE by ensuring the confidentiality and integrity of transmission data between TOE components.
	FTP_ITC.1	FTP_ITC.1 responds to T.TRANSMISSION_DATA_DAMAGE by ensuring the confidentiality and integrity of transmission data between the TOE and external IT entities.
T.WEAK_CRYPTO_PROTOCOLS	FCS_CKM.1 FCS_CKM.2 FCS_CKM.5 FCS_RBG.1 FCS_RBG.3 FPT_FLS.1 FPT_TST.1	FCS_CKM.1, FCS_CKM.2, FCS_CKM.5, FCS_RBG.1, FCS_RBG.3, FCS_RBG.5, FPT_FLS.1, and FPT_TST.1 respond to T.WEAK_CRYPTO_PROTOCOLS by ensuring that the cryptographic key is created and distributed according to the standard cryptographic algorithm and key length with a security strength of 112 bits or more when encrypting transmission data.
	FCS_CKM.6	FCS_CKM.6 responds to T.WEAK_CRYPTO_PROTOCOLS by ensuring that the cryptographic keys and their related information are destroyed according to the specified destruction method.
	FCS_COP.1(1) FCS_COP.1(2) FCS_COP.1(3) FCS_COP.1(4)	FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4) responds to T.WEAK_CRYPTO_PROTOCOLS by ensuring that cryptographic operations are performed according to the standard cryptographic algorithm and cryptographic key length with a security strength of 112 bits or more when encrypting transmission data.
T.TSF_COMPROMISE	FAU_ARP.1	FAU_ARP.1 responds to T.TSF_COMPROMISE by ensuring the ability to take response actions when detecting security violations such as TOE integrity compromise, etc.
	FAU_SAA.1	FAU_SAA.1 responds to T.TSF_COMPROMISE by ensuring the ability to review audited events to point out security violations, such as TOE integrity compromise.
	FIA_AFL.1(1) FIA_AFL.1(2) FIA_UAU.2 FIA_UAU.4(1) FIA_UAU.4(2) FIA_UAU.7 FIA_UID.2	FIA_AFL.1, FIA_UAU.2, FIA_UAU.4(1), FIA_UAU.4(2), FIA_UAU.7, and FIA_UID.2 respond to T.TSF_COMPROMISE by allowing access to the TOE only after successful user identification and authentication, ensuring the blocking of bypass access by threat agents.

P.AUDIT	FMT_MOF.1 FMT_MTD.1 FMT_PWD.1 FMT_SMF.1 FMT_SMR.1	FMT_MOF.1, FMT_MTD.1, FMT_PWD.1, FMT_SMF.1, and FMT_SMR.1 respond to T.TSF_COMPROMISE by dividing authorized user roles into administrator and end user when accessing and configuring management functions, and by providing security policies and functions based on those roles to ensure blocking of unauthorized access by threat agents.
	FPT_TST.1	FPT_TST.1 responds to T.TSF_COMPROMISE by ensuring the TSF self-testing for accurate operation of the TOE and ensuring that authorized administrators can verify the integrity of TSF data and the TSF itself.
	FAU_GEN.1	FAU_GEN.1 satisfies P.AUDIT by ensuring that audit records are generated for auditable events such as the startup/termination of the audit function and the success/failure of the identification and authentication of the administrator.
	FAU_SAR.1	FAU_SAR.1 satisfies P.AUDIT by providing the authorized administrator with the ability to retrieve audit records and ensuring that the audit records are presented in a manner suitable for the administrator to interpret the information.
	FAU_SAR.3	FAU_SAR.3 satisfies P.AUDIT by providing a selective audit review function based on logical relationship criteria for audit data.
	FAU_STG.1	FAU_STG.1 satisfies P.AUDIT by providing the ability to store audit data in local storage or transmit it to an external IT entity for storage in real time using a trusted channel for the TOE server.
	FAU_STG.4	FAU_STG.4 satisfies P.AUDIT by ensuring that appropriate response actions are taken if the audit trail on the TOE server exceeds the storage limit.
	FAU_STG.5	FAU_STG.5 satisfies P.AUDIT by ensuring the ability to take appropriate response actions when the audit trail of the TOE server is full.
P.SECURE_OPERATION	FMT_MOF.1	FMT_MOF.1 satisfies P.SECURE_OPERATION by ensuring that only authorized users have the ability to manage security functions.
	FMT_MTD.1	FMT_MTD.1 satisfies P.SECURE_OPERATION by ensuring that only authorized users have the ability to manage the TSF data.
	FMT_PWD.1	FMT_PWD.1 satisfies P.SECURE_OPERATION by ensuring that only authorized administrators have the ability to manage the combination rules and length of IDs and passwords, and by providing functions such as changing passwords when authorized administrators first access.
	FMT_SMF.1	FMT_SMF.1 satisfies P.SECURE_OPERATION by requiring management functions such as security functions to be performed by the TSF, the TSF data, etc. to be specified.

	FMT_SMR.1	FMT_SMR.1 satisfies P.SECURE_OPERATION by ensuring that authorized roles related to security management are specified.
P.CRYPTO_STRENGTH	FCS_CKM.1 FCS_CKM.2 FCS_CKM.5 FCS_CKM.6 FCS_RBG.1 FCS_RBG.3 FPT_FLS.1 FPT_TST.1	FCS_CKM.1, FCS_CKM.2, FCS_CKM.5, FCS_CKM.6, FCS_RBG.1, FCS_RBG.3, FCS_RBG.5, FPT_FLS.1, and FPT_TST.1 satisfy P.CRYPTO_STRENGTH by ensuring that the cryptographic keys required for standard cryptographic algorithms with a security strength of 112 bits or more are securely generated and distributed during data encryption.
	FCS_COP.1(1) FCS_COP.1(2) FCS_COP.1(3) FCS_COP.1(4)	FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4) satisfies P.CRYPTO_STRENGTH by ensuring that cryptographic operations are performed according to standard cryptographic algorithms with a security strength of 112 bits or more and the cryptographic key length during data encryption.

6.3.2. Security assurance requirements rationale

The evaluation assurance level of this ST was selected as EAL1+(ATE_FUN.1).

EAL1 can be applied in cases where a certain degree of trust in correct operation is required, but the threat to security is not serious. If EAL1 is developed according to the development methodology commonly applied by the developer, no additional effort is required from the developer to prepare the evaluation submissions. In other words, there is no need to invest more money or time to prepare for the evaluation.

EAL1 provides a basic level of assurance by analyzing the security functional requirements included in the limited security target using function and interface specifications and documentation to understand security behavior.

This analysis is supported by independent testing of the TSF and searching for potential vulnerabilities in the public domain(functional testing and penetration testing).

EAL1 does not require evidence of testing conducted by the developer based on functional specifications, but ATE_FUN.1 was added in this PP to allow the developer to independently test whether the TSF has been implemented correctly and whether defects have occurred, etc. and document the results.

6.3.3. Dependency of the security functional requirements

The following table shows dependency of security functional requirements.

Table 48. Theoretical Basis for Dependencies

No.	Security functional requirements	Dependency	Reference No.
1	FAU_ARP.1	FAU_SAA.1	3
2	FAU_GEN.1	FPT_STM.1	OE.TRUSTED_TIMESTAMP
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 FTP_ITC.1	2 43
7	FAU_STG.4	FAU_STG.2	-
8	FAU_STG.5	FAU_STG.2	-
9	FCS_CKM.1	[FCS_CKM.2 or FCS_CKM.5 or FCS_COP.1] FCS_CKM.3 [FCS_RBG.1 or FCS_RNG.1] FCS_CKM.6	10 11 13,14,15,16 17 12
10	FCS_CKM.2	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5] FCS_CKM.3	9
11	FCS_CKM.5	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.6	13,14,15,16 12
12	FCS_CKM.6	[FDP_ITC.1 or	9

		FDP_ITC.2 orFCS_CKM.1 or FCS_CKM.5]	11
13	FCS_COP.1(1)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5] FCS_CKM.6	9 11 12
14	FCS_COP.1(2)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5] FCS_CKM.6	-
15	FCS_COP.1(3)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5] FCS_CKM.6	9 11 12
16	FCS_COP.1(4)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5] FCS_CKM.6	9 11 12
17	FCS_RBG.1	[FCS_RBG.2 or FCS_RBG.3] FPT_FLS.1 FPT_TST.1	18 35 38
18	FCS_RBG.3	FCS_RBG.1	17
19	FIA_AFL.1(1)	FIA_UAU.1	25
20	FIA_AFL.1(2)	FIA_UAU.1	25
21	FIA_IMA.1	-	-
22	FIA_SOS.1	-	-
23	FIA_SOS.2	-	-
24	FIA_SOS.3	FIA_SOS.2	23
25	FIA_UAU.2	FIA_UID.1	29
26	FIA_UAU.4(1)	-	-
27	FIA_UAU.4(2)	-	-
28	FIA_UAU.7	FIA_UAU.1	25
29	FIA_UID.2	-	-
30	FMT_MOF.1	FMT_SMF.1 FMT_SMR.1	33 34
31	FMT_MTD.1	FMT_SMF.1 FMT_SMR.1	33 34
32	FMT_PWD.1	FMT_SMF.1 FMT_SMR.1	33 34
33	FMT_SMF.1	-	-
34	FMT_SMR.1	FIA_UID.1	29

35	FPT_FLS.1	-	-
36	FPT_ITT.1	-	-
37	FPT_PST.1	-	-
38	FPT_TST.1	-	-
39	FTA_MCS.2	FIA_UID.1	29
40	FTA_SSL.3	FMT_SMR.1	34
41	FTA_TSE.1(1)	-	-
42	FTP_ITC.1	-	-

FAU_GEN.1 depends on FPT_STM.1, but since a trusted timestamp is provided by the security objective OE.TRUSTED_TIMESTAMP for the TOE operating environment in this Security Target, it is not necessary to satisfy the dependency.

FIA_UAU.2 hierarchically includes FIA_UAU.1. Therefore, by conforming to FIA_UAU.2, the requirement of FIA_UAU.1 is satisfied through the hierarchical relationship.

FIA_UID.2 hierarchically includes FIA_UID.1. Therefore, by conforming to FIA_UID.2, the requirement of FIA_UID.1 is satisfied through the hierarchical relationship.

FCS_CKM.3 Cryptographic key access component is intended to allow the requirements for using keys outside of the TOE (e.g. . backup, archival, escrow, recovery) be specified and to require the method used to access the cryptographic key be specified. Since this function is not required in the Security Requirements for Government, it has not been added in this ST.

The FCS_COP.1(2) cryptographic operation component specifies a hash algorithm, but since this algorithm is not used for key, it is not necessary to satisfy FDP_ITC.1, FDP_ITC.2, FCS_CKM.1, or FCS_CKM.5.

FAU_STG.4 and FAU_STG.5 depend on FAU_STG.2, but since OE.SECURE_DBMS—a security objective for the TOE operating environment in this Security Target—ensures that audit records containing the audit trail are protected against unauthorized deletion or modification, it is not necessary to satisfy the dependency.

6.3.4. Dependency of the security assurance requirements

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

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

This security target complies with the EAL1 assurance package, but ASE_OBJ.1 includes ASE_SPD.1, which is absent in the EAL1 assurance package due to a dependency.

However, this direct rationale security target includes a security problem definition, and ASE_OBJ.1 provides indirect assurance on the security problem definition, such as requesting an investigation to see if the security objectives for the TOE operating environment are traced to the security problem definition. Therefore, ASE_SPD.1, which is related to the request for a description of the security problem definition, was judged not to be absolutely necessary and was not added to this security target.

ASE_REQ.1 also includes ASE_SPD.1, which is absent in the EAL1 assurance package due to dependency. However, this direct rationale security target includes a security problem definition, and ASE_REQ.1 provides indirect assurance on the security problem definition, such as requesting an investigation to see if the SFR is traced to the security problem definition. Therefore, ASE_SPD.1, which is related to the request for description of security problem definition, was judged not to be absolutely necessary and was not added to this security target.

7. TOE Summary Specification

7.1. Security Audit

The security audit function of the TOE consists of the following security functions: security alarms, audit data generation, audit storage monitoring and security violation response, and audit data review.

7.1.1. Audit Data Generation

The TOE generates audit data as described below. Each audit record includes items such as the date and time of the event, type of event, identity of the subject (if available), and the result of the event (success or failure). These audit records are stored in the DBMS. In the SSO agent, there is no separate audit data repository; instead, audit event information is transmitted to the SSO server and stored in the DBMS of the SSO Server. The audit events that require the SSO agent to directly request the generation of audit data on the SSO Server are those events prefixed with “SP” in the system audit data. If the connection between the SSO agent and the SSO server is lost, the audit event information is temporarily stored in the memory of the SSO agent. Once the connection is re-established, the accumulated information is transmitted to the SSO server and stored in the DBMS. For the stability of the SSO agent, its memory can store up to 1,000 audit event entries; any audit event information generated beyond this limit will not be stored. The detailed contents of each audit item are as follows.

Table 49. Detailed Audit Data Items by Category

Audit Category	Audited Events	Specifics
User Audit Data	• Login • Logout • Reaching threshold of authentication attempts • Unlocking after login lock • Logout due to session timeout • Duplicate login with the same ID • Reuse of authentication information blocked • No authentication token available for linkage	• Timestamp • IP address • ID • Type of event • Success/failure • Audit log details
Administrator Audit Data	• Login • Logout • Reaching threshold of authentication attempts • Unlocking after login lock • Logout due to session timeout • Duplicate login with the same ID • Duplicate login with overlapping authority • Password change • Reuse of authentication information blocked • Access attempt to the administrator web browser from an unauthorized IP address	• Timestamp • IP address • ID • Type of event • Success/failure • Audit log details

System Audit Data	• Start of administrator audit • End of administrator audit • Start of user audit • End of user audit • Start of configuration change audit • End of configuration change audit • Start of IDP system audit • End of IDP system audit • Start of SP system audit • End of SP system audit • Cryptographic key generation failure • Cryptographic operation failure • Successful destruction of authentication token • Database storage full • Server storage full • IDP self-test • Integrity check of IDP libraries and configuration files • Self-test and integrity check of IDP cryptographic module • SP self-test • Integrity check of SP libraries and configuration files • Self-test and integrity check of SP cryptographic module • SP server startup • Mutual authentication • DB password change	• Timestamp • Type of event • Success/failure • Audit log details
Configuration Change Audit Data	• Authentication server management • Integration server management • Alarm settings management • Management server configuration • Server environment settings • Self-test • Default password policy management • Authentication information management by integration server • Authentication session management • My information management • User management	Per Configuration Change Item: • Timestamp • IP address • Administrator ID • Change menu • Success/failure • Details of the change

	• Administrator management • User session management	

Related SFR: FAU_GEN.1, FAU_STG.1

7.1.2. Audit Storage Monitoring and Security Violation Response

The TOE periodically monitors for potential security threats based on the alert cycle configured by the administrator. Specifically, it detects the following events specified in [6.1.1.3. FAU_SAA.1]: self-test failure events, failures in integrity verification audit events, and [administrator reaching the threshold of authentication attempts] as part of the response to user authentication attempt threshold being reached. Upon detection, the TOE sends alert emails to the authorized administrator's email address.
Additionally, when the audit storage (DBMS) reaches the defined disk usage threshold, the TOE also sends an alert email to the authorized administrator.

Related SFR: FAU_ARP.1, FAU_SAA.1, FAU_STG.4, FAU_STG.5

7.1.3. Audit Data Review

The TOE provides a function that allows authorized administrators to review audit data stored in the DBMS via the administrator web browser. Audit information can be retrieved based on the search criteria specified in the "Audit Data Search Criteria," depending on the search conditions entered for the audit records.

Table 50. Audit Data Search Criteria

Logical Conditions		Selection and Sorting Method
User Audit Logs	① AND: (Duration, User ID, Event type) ② OR: Event success/failure ③ Sorting: Ascending or descending (select one)	AND of the "Logical Conditions" (①, ②, ③)
Administrator Audit Logs	① AND: (Duration, User ID, Event type) ② OR: Event success/failure ③ Sorting: Ascending or descending (select one)	
System Audit Logs	① AND: (Duration, Event type) ② OR: Event success/failure ③ Sorting: Ascending or descending (select one)	
Configuration Change Audit Logs	① AND: (Duration, Event type, Change menu type) ② OR: Event success/failure ③ Sorting: Ascending or descending (select one)	

Related SFR: FAU_SAR.1, FAU_SAR.3

7.2. Cryptographic Support

The TOE performs cryptographic key generation, key distribution, key derivation, key destruction, cryptographic operations, and random bit generation using validated cryptographic modules operating in approved mode.

7.2.1. Specification of Validated Cryptographic Modules

The TOE implements its cryptographic support functions using the validated cryptographic modules listed below:

Table 51. General Information on Validated Cryptographic Modules

Cryptographic Module Name	eXCryptoLib V1.0
Developer (Organization)	Tomato System Co., LTD.
Validation Date	April 16, 2025
Security Level	1
Validation Number	CM-268-2030.4
Expiration Date	April 16, 2030

7.2.2. Cryptographic Key Generation / Distribution / Derivation / Destruction

7.2.2.1. Cryptographic Key Generation

The key generation process specified in FCS_CKM.1 is categorized by usage as shown in the table below:

Table 52. Key Generation Details

Generated Key	Key Length (bit)	Key Generation Entity	Key Generation Algorithm (standard)	Time of Generation
DEK	128	SSO Server	HASH_DRBG (SHA-512) [TTAK.KO-12.0331-Part1](2018), [TTAK.KO-12.0331-Part2](2018)	- At initial startup of the SSO server
TSF data integrity verification key	256	SSO Server	HASH_DRBG (SHA-512) [TTAK.KO-12.0331-Part1](2018), [TTAK.KO-12.0331-Part2](2018)	- Upon SSO server startup
Private key	2048	SSO Server	RSAES ([KS X ISO/IEC 18033-2](2017))	- At initial startup of the SSO server
Public key	2048	SSO Server	RSAES ([KS X ISO/IEC 18033-2](2017))	- At initial startup of the SSO server
Session key for mutual authentication	128	SSO Agent	HASH_DRBG (SHA-512) [TTAK.KO-12.0331-Part1](2018), [TTAK.KO-12.0331-Part2](2018)	- Upon SSO agent startup
Session key for encrypted communication	128	SSO Agent	HASH_DRBG (SHA-512) [TTAK.KO-12.0331-Part1](2018), [TTAK.KO-12.0331-Part2](2018)	- Upon successful mutual authentication between SSO server and SSO agent

The generation of cryptographic keys uses the HASH_DRBG (SHA-512) algorithm as specified in [TTAK.KO-12.0331-Part1] (2018) and [TTAK.KO-12.0331-Part2] (2018), and the RSAES algorithm as specified in [KS X ISO/IEC 18033-2] (2017).

The DEK, TSF data integrity verification key, and private key are decrypted and used when needed, and are destroyed immediately after use.

The session key for mutual authentication is generated at the startup of the SSO agent for the purpose of mutual authentication, and is destroyed immediately after successful mutual authentication with the SSO server. Once

mutual authentication is successful, a session key for encrypted communication is generated, and it is destroyed immediately upon termination of the SSO server or the SSO agent.

Related SFR: FCS_CKM.1, FCS_CKM.6

7.2.2.2. Cryptographic Key Distribution

Upon startup, the SSO server generates a public/private key pair for mutual authentication. The private key is encrypted with the DEK using ARIA/CCM (128-bit) and stored as a file, while the public key must be delivered to the SSO agent in the form of a file named “sso_public_key.”

Related SFR: FCS_CKM.2

7.2.2.3. Cryptographic Key Derivation

When generating the KEK, the TOE derives the encryption key from the password entered by the administrator using the PBKDF algorithm. The KEK is not stored separately; instead, each time the SSO server is restarted, the administrator enters the KEK password through the CLI input interface. The entered KEK password is stored in the memory of the SSO server using a proprietary encoding method. Whenever the DEK needs to be decrypted, the password is derived again to regenerate the same KEK created at startup, and this regenerated KEK is discarded immediately after use. Because the password is entered manually by the administrator, the value may differ across TOEs. The salt used in password derivation is the value stored in the *exsignon.dek* file at the time of the initial startup.

The PBKDF algorithm used is implemented in a secure manner as specified in [TTAK.KO-12.0334-Part1~2] (2018). For the generation of keys other than the KEK, the TOE does not use password-based key derivation methods.

Table 53. Key Generation Details

Generated Key	Cryptographic Algorithm (standard)	Key Length (bit)	Key Derivation Entity	Key Derivation Algorithm	Time of Derivation
KEK	ARIA/CBC ([KS X 3254](2016), [KS X 1213-1](2019))	128	SSO Server	PBKDF	- At initial startup of the SSO server - as needed

Related SFR: FCS_CKM.5

7.2.2.4. Cryptographic Key Destruction

When a cryptographic key is destroyed, it is securely discarded using a triple overwrite method with the value ‘0x00’.
The destruction time of each cryptographic key is as follows.

Table 54. Key Destruction Details

Generated Key	Time of Destruction
KEK	- Upon SSO server shutdown - Immediately after use
DEK	- Immediately after use
TSF data integrity verification key	- Upon SSO server or SSO agent shutdown - Immediately after use
Private key	- Immediately after use
Public key	- N/A
Session key for mutual authentication	- Upon SSO server or SSO agent shutdown - Immediately after use - Upon completion of mutual authentication - Upon termination of encrypted communication

Session key for encrypted communication	- Upon termination of encrypted communication

Related SFR: FCS_CKM.6

7.2.2.5. Cryptographic Operations

The key generation process specified in FCS_COP.1(1), (2), (3), and (4) is categorized by usage as shown in the table below.

Table 55. Usage of Each Cryptographic Operation

Cryptographic Algorithm	Key Used	Usage
ARIA/CBC (128bit)	KEK	Encryption and decryption of DEK, TSF data integrity verification key, and private key for mutual authentication
	DEK	Encryption and decryption of TSF data
ARIA/CCM (128bit)	DEK	Encryption and decryption of authentication tokens
	Session key for mutual authentication	Encryption and decryption of authentication information for mutual authentication
	Session key for encrypted communication	Encryption and decryption of transmitted data
SHA-256	N/A	Hashing of general user and administrator passwords
HMAC-SHA256	TSF data integrity verification key	Extraction and verification of MAC values to verify the integrity of TSF data
RSAES(2048bit)	DEK	Mutual authentication between the SSO server and the SSO agent

The user password used by the TOE for user identification and authentication is stored in the database in the form of SHA-256|(hashed password) using a hash (SHA-256) so that it cannot be decrypted. During the hashing process, a 128-bit salt generated by a random number generator is applied, and the iteration count is repeated 1,000 times to securely store the password.

Related SFR: FCS_COP.1(1),(2),(3),(4)

7.2.3. Random Number Generation

The TOE uses a validated cryptographic module when generating cryptographic keys required for cryptographic operations. The security strength of the random number generator is 384. The cryptographic module collects entropy on its own.

The specifications of the random number generator used in the validated cryptographic module are shown below.

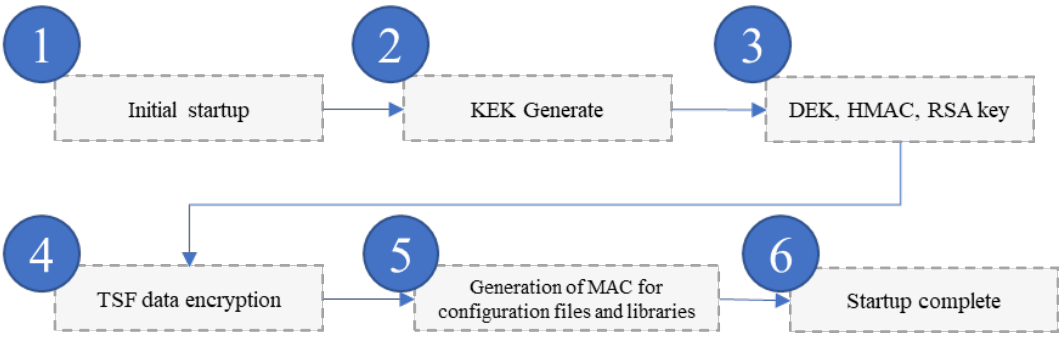
Table 56. Random Number Generator

Random Number Generator	Details	Purpose
Hash_DRBG	SHA-512 Prediction Resistance (PR): Not used Personalization String: Not used Additional Input: Not used Security Strength: 384 Nonce: 256 bits	Key generation Random number generation

Related SFR: FCS_RBG.1

7.2.4. Startup of the SSO server

Figure 4. Startup of the SSO server



① Startup of the SSO server

② The administrator enters a password, and the KEK is generated using the PBKDF algorithm

③ For the DEK, HMAC key, and the IVs of the RSA public/private key pair, the IVs are generated using the HASH_DRBG (SHA-512) algorithm. Each IV together with the KEK is used to encrypt the corresponding key, and the encrypted value is then encoded with a proprietary encoding method and stored in a file along with the generated IV. (Upon server restart, the keys stored in files are decrypted with the KEK and loaded into memory.)

④ The DEK is encrypted using ARIA/CBC (128-bit) and stored. (Database connection information is stored in a file, and the SMTP password is stored in the DBMS.) (Upon restart, the TSF data stored in the file or the DBMS is decrypted and loaded into memory.)

⑤ A MAC value is generated for integrity verification of configuration files and library files, and stored in a file. (Upon restart, the previously generated MAC value is compared with the actual MAC value to verify integrity.)

⑥ Startup completed

Related SFR: FCS_CKM.1, FCS_CKM.5, FCS_COP.1(1),(3),(4), FCS_RBG.1

7.3. Identification and Authentication

Users of the TOE must undergo identification and authentication as either administrators or general users in order to use or manage the services provided by the TOE. For administrators, when logging into the administrator web browser provided by the TOE, the TOE verifies the administrator's identity by checking whether the IP address of the administrator's PC matches an allowed IP address configured by the administrator, and by verifying the ID and password. For general users, if an authentication token is present, the TOE verifies the validity of the token and checks whether a session exists for the user. If no authentication token is present, the TOE only verifies whether the ID and password entered on the login screen match. The authentication token is the encrypted authentication token received by the SSO agent from the SSO server, which is then sent back to the SSO server. The SSO server decrypts the token and verifies whether it matches the authentication token stored in the session.
Additional identification and authentication methods such as FIDO are not supported.

Related SFR: FIA_UAU.2, FIA_UID.2

7.3.1. Handling of User Authentication Failures

If the number of authentication failures at the SSO server or SSO agent exceeds the threshold value (1–5 times, default 5 times) defined by the super administrator or general administrator, login attempts will be blocked for the period of time defined by the super administrator or general administrator (5, 10, 30, or 60 minutes, default 5 minutes).

Related SFR: FIA_AFL.1(1),(2)

7.3.2. Protection of Authentication Information

When logging in, the password is displayed as a masked character (*), and no feedback is provided regarding authentication failure when the password is incorrect.
When logging into the administrator web browser or as a regular user, the password is encrypted in the JavaScript environment using the CryptoJS library with AES/CBC (128-bit). A Data Encryption Key (DEK) and a randomly generated IV (Initialization Vector) are used in the encryption process. The encrypted password and IV are concatenated and sent to the server. On the server side, the encrypted password is decrypted using the received IV and the DEK.

Related SFR: FIA_UAU.7 , FIA_AFL.1

7.3.3. Password Policy Verification

Passwords for administrators and general users can be created by the administrator in accordance with the password policy defined within the range of the password security criteria specified in the table below. The verification mechanism provided during password creation and modification also complies with this policy.

Table 57. Password Security Criteria

Category	Details	
Requirements	Allowed Characters	52 English letters (uppercase and lowercase)
		10 numeral digits (0–9)
		Special characters(! @ # $ % ^ & * ( ) - _ = + [{] } | ; : , < . > / ? ` ~)
	Minimum/Maximum Length	Password minimum/maximum length: 9 to 20 characters (configurable from 9 to 20)
		Must include at least one of each: numeral digit, uppercase letter, lowercase letter, and special character (default: 1 of each, configurable from 1 to 2)
Prohibitions	Password must not be the same as the user account (ID)	
	Consecutive repetition of the same character or number is not allowed (default: allowed up to 2 times, configurable from 1 to 2)	
	Sequential input of 4 or more characters or digits in keyboard order is not allowed	
	Reuse of the previously used password is prohibited	

Related SFR: FIA_SOS.1, FMT_PWD.1

7.3.4. Prevention of Authentication Information Reuse

For general users, there are two login methods: one using ID/password and another using the integrated authentication process based on an authentication token.
The ID/password-based login method requires the user to enter their ID and password, upon which an authentication token is generated. The integrated authentication method based on the authentication token uses the token generated during the ID/password-based login. Once an authentication token has been used, it is immediately discarded to prevent reuse.

When logging in to the administrator web browser, verification is performed using a random CSRF token issued one-time. Tokens that have been used or not used within the 5-minute validity period cannot be reused.

Related SFR: FIA_SOS.2, FIA_UAU.4(1),(2)

7.3.5. Direct Data Transmission Between TOE Components

When direct data transmission between TOE components is required, the SSO server and SSO agent perform mutual authentication and encrypted communication using the RSA key pair generated by the SSO server and the session key generated by the SSO agent. If the connection between the SSO server and SSO agent is interrupted, it is automatically re-established depending on the situation, and the mutual authentication process is performed again upon reconnection.

Table 58. Algorithm Used in TOE Internal Mutual Authentication

Type	Item	Description
Mutual authentication	RSA	Encryption and decryption of session key for mutual authentication
	ARIA/CCM	Encryption/decryption and authentication
Session Key	Creation	HASH_DRBG (SHA-512)
Encrypted communication	Cryptographic algorithm	ARIA/CCM

Related SFR: FIA_IMA.1, FCS_CKM.2, FCS_COP.1(1), (4)

7.3.6. Generation, Verification, and Destruction of Authentication Tokens

The components of the authentication token—*type code*, *endpointIndex*, *sourceId*, and *messageHandle*—are concatenated and encrypted using ARIA/CCM (128 bit) with the DEK.
The authentication token is issued by the SSO server upon completion of user login, and the authentication token is stored in the user's browser cookie. During SSO integration using the user's login credentials, the authentication information stored in the browser cookie is transmitted to the SSO server. The SSO server then decrypts the authentication token and verifies whether it matches the decrypted value of the authentication token stored in the session.
The authentication token is stored in encrypted form in the session of the SSO server using ARIA/CCM (128 bit) with the DEK. The cases in which the authentication token is destroyed are: when a new authentication token is generated and stored in the session, upon completion of authentication token verification, at logout, and upon session timeout.
When destroying the authentication token, it is securely destroyed by overwriting it three times with "0x00" as specified in 6.1.3.6. FIA_SOS.3.

Related SFR: FIA_SOS.2, FIA_SOS.3

7.4. Security Management

The SSO server performs security function and TSF data management through the administrator web browser. The security management functions provided by the SSO server are accessible only to authorized administrators. Authorized administrators are classified into super administrator, general administrator, and monitoring administrator according to their roles. The security functions and TSF data management according to each role of administrator meet the specifications described in the table below, and the TOE does not provide a separate update function.

Table 59 Specification of Security Functions Management

Subclass	Management Actions for Security Functions	Classification of Administrator Authorization
FIA_UAU.2 FIA_UID.2	Register, delete, modify users; assign user privileges	Super Administrator, General Administrator
FMT_PWD.1	Configure password composition and length policy	Super Administrator, General Administrator
FIA_AFL.1	Configure allowed number of user authentication failures	Super Administrator, General Administrator
	Configure time before reactivation after authentication function is deactivated	Super Administrator, General Administrator
FTA_TSE.1	Register, delete, and modify IP addresses of management terminals	Super Administrator, General Administrator
FMT_MTD.1	View agent information – status, version, applied security policy	Super Administrator, General Administrator
	Manage agent security policy – configure policy	Super Administrator, General Administrator
	Configure authentication information for accessing external IT entities	Super Administrator, General Administrator
FPT_TST.1	Request self-test of TOE server	Super Administrator, General Administrator
	Request integrity check of TOE configuration and the TOE itself	Super Administrator, General Administrator
FTA_SSL.3	Configure session timeout for users	Super Administrator, General Administrator
	TOE Version Information Retrieval	Super Administrator, General Administrator, Monitoring Administrator
FTA_MCS.2	Configure number of concurrent user sessions	Super Administrator, General Administrator
FAU_SAR.1	View audit records	Super Administrator, General Administrator, Monitoring Administrator
FAU_STG.4	Configure response settings for audit record loss	Super Administrator, General Administrator
FAU_STG.5		Super Administrator, General Administrator

Table 60 TSF Data Management Actions Subject to FMT_MTD.1

Related SFR	TSF Data Management Actions (TSF data shown in bold)	Classification of Administrator Authorization
FIA_UAU.2 FIA_UID.2	Granting **privileges** to general administrator and monitoring administrator accounts (IDs) Adding, deleting, and modifying general administrator and monitoring administrator **IDs** Modifying **passwords and user information (name, role, email, phone number)** for general	Super Administrator

	administrator and monitoring administrator accounts	
	Adding, deleting, and modifying general user **IDs**	Super Administrator, General Administrator
FMT_PWD.1(Extended)	Configure **password length and combination** policies for administrators and general users	Super Administrator, General Administrator
FMT_PWD.1(Extended) FMT_MTD.1	Modify own **password and personal information (name, role, email, phone number)**	Super Administrator, General Administrator, Monitoring Administrator
	Add or delete **password and personal information (name, email, phone number)** of general users	Super Administrator, General Administrator
FIA_AFL.1(1) FIA_AFL.1(2)	Setting the **allowable number of authentication failures**	Super Administrator, General Administrator
	Configure **the reactivation time after authentication function deactivation**	Super Administrator, General Administrator
FTA_TSE.1	Register, delete, and modify **management terminal IP addresses**	Super Administrator, General Administrator
FMT_MTD.1	Agent inquiry - Required inquiry information: **agent version, security policy applied to the agent, agent operation status** (activated/deactivated), **agent integrity verification result** (success/failure)	Super Administrator, General Administrator
	Managing **agent security policy**	Super Administrator, General Administrator
	Configuring **authentication information for accessing external IT entities (e.g., SMTP)**	Super Administrator, General Administrator
	Change **default passwords used to access internal TOE components (DBMS) or external IT entities**	Super Administrator, General Administrator
	View **identification information of the TOE and TOE components (e.g., server, agent)**	Super Administrator, General Administrator
FAU_STG.4	**Setting audit trail threshold** for notifying administrators of potential audit record loss	Super Administrator, General Administrator
FTA_SSL.3	Configuring **automatic session termination time for users**	Super Administrator, General Administrator
FAU_SAR.1	Viewing **audit records**	Super Administrator, General Administrator, Monitoring Administrator

The super administrator and general administrators perform password management functions as specified in the table below. IDs must be between 5 and 20 characters in length and may include uppercase and lowercase English letters (a–z, A–Z), numbers (0–9), and certain special characters (- _ @ .). Each ID must be unique and cannot be registered more than once. Additionally, all administrators must reset their password after initial authentication using the administrator account before they can access other functions.

Table 61 FMT_PWD.1 Password Management

1. [Password Composition Rules and/or Length]
1) Allowed Characters
o 52 English letters (uppercase or lowercase)
o 10 numeral digits (0–9)
o Special characters(! @ # $ % ^ & * ( ) - _ = + [{] } | ; : , < . > / ? ` ~)
2) Minimum/Maximum Length
o Password minimum/maximum length: 9 to 20 characters (configurable from 9 to 20)
o Must include at least one of each: numeral digit, uppercase letter, lowercase letter, and special character (default: 1 of each, configurable from 1 to 2)

2. [Other Management Settings Such as Special Characters Excluded from Passwords]
o Password must not be the same as the user account (ID)
o Consecutive repetition of the same character or number is not allowed (default: allowed up to 2 times, configurable from 1 to 2)
o Sequential input of 4 or more characters or digits in keyboard order is not allowed
o Reuse of the previously used password is prohibited

The TSF classification values by administrator role refer to the list of security function interfaces of eXSignOn V4.0 provided below.

Table 62. Security Function Interfaces of eXSignOn V4.0

Interface Identifier	Interface Name	Role
TSFI_IDP_KEK_GEN	Generate KEK	Super Administrator
TSFI_IDP_DEK_GEN	Generate DEK	Super Administrator
TSFI_ENCRYPT_COMMUNICATION	Establishment of Encrypted Communication Channel	Super Administrator
TSFI_IDP_LICENSE_VERIFICATION	License Verification	Super Administrator
TSFI_ADMIN_AUTH	Administrator Authentication	Super Administrator, General Administrator, Monitoring Administrator
TSFI_ADMIN_LOGOUT	Administrator Logout	Super Administrator, General Administrator, Monitoring Administrator
TSFI_USER_AUTH	User Authentication	General User
TSFI_USER_LOGOUT	User Logout	General User
TSFI_SEVER_STATUS	View Server Status	Super Administrator, General Administrator, Monitoring Administrator
TSFI_IDP_MANAGE	Manage Authentication Server	Super Administrator, General Administrator
TSFI_LIC_MANAGE	Manage License	Super Administrator, General Administrator
TSFI_SP_ADD	Add Integration Server	Super Administrator, General Administrator
TSFI_SP_MANAGE	Manage Integration Server	Super Administrator, General Administrator
TSFI_SP_ACTIVE	Activate Integration Server	Super Administrator, General Administrator
TSFI_SP_DELETE	Delete Integration Server	Super Administrator, General Administrator
TSFI_ALERT_MANAGE	Manage Alerts	Super Administrator, General Administrator

TSFI_ALERT_TEST	Send Test Alert	Super Administrator, General Administrator
TSFI_CC_MANAGE	Manage Control Server	Super Administrator, General Administrator
TSFI_SERVER_MANAGE	Configure Server Environment	Super Administrator, General Administrator
TSFI_SELF_TEST	Self-Test	Super Administrator, General Administrator
TSFI_PWD_MANAGE	Manage Default Password Policy	Super Administrator, General Administrator
TSFI_NAMEID_MANAGE	Manage Authentication Information by Integration Server	Super Administrator, General Administrator
TSFI_AUTH_SESS_MANAGE	Manage Authentication Session	Super Administrator, General Administrator
TSFI_MY_INFO_MANAGE	Manage My Information	Super Administrator, General Administrator, Monitoring Administrator
TSFI_ADM_ADD	Add Administrator	Super Administrator
TSFI_ADM_MANAGE	Edit Administrator	Super Administrator
TSFI_ADM_DELETE	Delete Administrator	Super Administrator
TSFI_USER_ADD	Add General User	Super Administrator, General Administrator
TSFI_USER_MANAGE	Edit General User	Super Administrator, General Administrator
TSFI_USER_DELETE	Delete General User	Super Administrator, General Administrator
TSFI_SESS_MANAGE	Manage User Session	Super Administrator, General Administrator
TSFI_USER_AUDIT_LOG	User Audit Logs	Super Administrator, General Administrator, Monitoring Administrator
TSFI_ADMIN_AUDIT_LOG	Administrator Audit Logs	Super Administrator, General Administrator, Monitoring Administrator
TSFI_SYSTEM_AUDIT_LOG	System Audit Logs	Super Administrator, General Administrator, Monitoring Administrator
TSFI_SETTING_AUDIT_LOG	Configuration Change Audit Logs	Super Administrator, General Administrator, Monitoring Administrator

Related SFR: FMT_MOF.1, FMT_MTD.1, FMT_PWD.1(Extended), FMT_SMF.1, FMT_SMR.1

7.5. TSF Protection

The TSF protects TSF data from unauthorized disclosure and modification when the data is transmitted between separated components of the TOE or stored in TSF-controlled repositories. This includes user account passwords, database and SMTP access credentials, and TOE configuration values.

7.5.1. Maintaining a Secure State in the Event of a Failure

If a failure occurs in the entropy source (e.g., failure of noise source health test), the TOE transitions to a critical error state and halts the operation of the validated cryptographic module whose security and implementation conformance are validated by the Korea Cryptographic Module Validation Program (KCMVP), in order to maintain a secure state.

Each time noise is collected, health tests (Repetition Count Test, hereinafter RCT, and Adaptive Proportion Test, hereinafter APT) and conditioning procedures are performed. The noise source health test of the random number generator is conducted when random numbers are generated using a cryptographic module API such as generateRandom, and also when random numbers are internally generated and used during cryptographic operations such as encryption or digital signatures. When random numbers are generated, noise source generation is required. At this time, the health of the collected noise source is tested, and only noise sources that pass the health test are used to generate random numbers.

The noise source health test consists of the Repetition Count Test and the Adaptive Proportion Test. If either of these tests fails, the TOE transitions to a critical error state, and the cryptographic module must be reinstalled or restarted before it can be used again.

Related SFR: FPT_FLS.1

7.5.2. Basic Protection of Internally Transmitted TSF Data

The TOE ensures confidentiality and integrity by encrypting transmitted TSF data using ARIA/CCM (128 bit). In addition, key distribution is performed through the RSA key pair generated by the SSO server, and encrypted communication is carried out through the mutual authentication protocol specified in “7.3.5. Direct Data Transmission Between TOE Components”.

Upon completion of mutual authentication, secure communication continues using a session key for encrypted communication, which is generated by the SSO agent.

Related SFR: FPT_ITT.1

7.5.3. Protection of Stored TSF Data

The TSF protects the information stored in TSF-controlled repositories from disclosure. However, the HMAC key used by the SSO agent for integrity verification is not stored separately, as it is automatically received from the SSO server at each server startup and used from the key stored in file.

Related SFR: FPT_PST.1

7.5.3.1. Protection of TSF Data (Sensitive Information) Stored by the TOE Server

“TSF data that must be encrypted when stored” shall be encrypted regardless of its storage location or format.

Table 63. TSF Data Requiring Encryption Upon Save

Category	TSF Data Requiring Encryption Upon Save	Remarks
TSF data stored by TOE server	General user password	User passwords for user identification and authentication are stored using the one-way hash algorithm SHA-256 to prevent decryption. - A randomly generated secret value, called a salt, is applied when hashing passwords. - The salt is generated using a random number generator based on the HASH-DRBG (SHA-512) algorithm, with a size of 128 bits. - The iteration count is 1,000. - As a one-way hash algorithm is used, there is no encryption key, and the encrypted password is stored in the database.
	Administrator password	

	DEK, IV, TSF Data Integrity Verification Key (HMAC)	DEK, IV, and the TSF Data Integrity Verification Key (HMAC) are encrypted using the Key Encryption Key (KEK) with ARIA/CBC (128-bit) and stored securely. The encryption key is not hardcoded into the TOE; instead, it is encrypted with the KEK and stored as a file.
	Integrity verification value	The integrity verification values are generated by hashing each library and configuration file that constitutes the TOE using the TSF Data Integrity Verification Key (HMAC) with HMAC (SHA-256), and are stored in the form of (filename)=(hash value).
	public key	The public key is encoded with a proprietary encoding method and stored.
	private key	The private key is encrypted using the Key Encryption Key (KEK) with ARIA/CBC (128-bit) and stored securely. The encryption key is not hardcoded into the TOE; instead, it is encrypted with the KEK and stored as a file.
	KEK salt	The KEK salt is encoded with a proprietary encoding method and stored.
	KEK Integrity Verification Value	The KEK integrity verification value is generated by encrypting the string “tomato” with ARIA/CBC (128-bit) using the Key Encryption Key (KEK), encoding it with a proprietary encoding method, and stored as a file. It is used to verify whether the KEK password matches the value previously entered.
	Passwords in WAS logs	Passwords are not recorded in WAS logs.
	Encryption keys in WAS logs	Encryption keys do not remain as plain text in WAS logs.
	DBMS access information	Encrypted using DEK with ARIA/CBC (128-bit) and stored as a file.
	SMTP password	Encrypted using DEK with ARIA/CBC (128-bit) and stored securely.
	Authentication tokens	Authentication tokens are automatically destroyed upon successful validation.

Related SFR: FPT_PST.1

7.5.3.2. Protection of TSF Data (Configuration Values, Audit Records) Stored by the TOE Server

The TOE shall provide functionality to ensure that only authorized administrators can access stored TOE configuration values.

Table 64. TSF Data That Must Be Securely Protected via Encryption, Access Control, etc.

Category	TSF Data That Must Be Securely Protected via Encryption, Access Control, etc.	Storage Location	Remarks
TSF data stored by TOE server	TOE configuration values (security policies, environment configuration parameters, etc.)	DB	The TOE does not support backing up configuration values as external files.
	Audit records	DB	Audit records are stored in the local DBMS and protected by the TOE operating environment.

Related SFR: FPT_PST.1

7.5.3.3. Protection for Cryptographic Key Storage

When the TOE stores the following cryptographic keys, it shall securely store them in accordance with the requirements specified in the table below:

Table 65. Protection for Cryptographic Key Storage

Category	TSF Data to Be Securely Protected According to the Conditions Described in the Remarks	Remarks
KEK	Key Encryption Key (KEK)	The KEK (Key Encryption Key) is not stored; instead, a KEK password is entered each time the SSO server starts. This password is stored in memory using a proprietary encoding method. Whenever the KEK is needed, it is newly generated by deriving it from the password stored in memory using a PBKDF (Password-Based Key Derivation Function) algorithm. The salt is stored in a file during the initial startup of the SSO server and is loaded from that file during subsequent startups.

Related SFR: FPT_PST.1

7.5.3.4. Protection of TSF Data (Sensitive Information) Stored by the SSO Agent

Table 66. Protection of TSF Data (Sensitive Information) Stored by the SSO Agent

Category	TSF Data Requiring Encryption Upon Save	Remarks
TSF data stored in files and systems by the SSO agent	TSF data integrity verification key (HMAC)	The SSO agent uses symmetric key encryption when communicating securely with the SSO server.
	Integrity verification value	The integrity verification values are generated by hashing each library and configuration file that constitutes the TOE using the TSF Data Integrity Verification Key (HMAC) with HMAC (SHA-256), and are stored in the form of (filename)=(hash value).

Related SFR : FPT_PST.1

7.5.3.5. Protection of TSF Data (Configuration Values, Audit Data) Stored by the SSO Agent

Table 67. Protection of TSF Data (Configuration Values, Audit Data) Stored by the SSO Agent

Category	TSF Data Requiring Encryption Upon Save	Storage Location	Remarks
TSF data stored in files and systems by the SSO agent	TOE configuration values (security policies, environment configuration parameters, etc.)	N/A	The TOE does not support backing up configuration values as external files.
	Audit records	Memory	Audit records are encrypted using the communication session key with ARIA/CCM (128-bit) and immediately transmitted to the SSO server through a proprietary encrypted communication channel. If transmission is not possible, up to 1,000 records are stored in the memory of the SSO agent to ensure stability.

Related SFR : FPT_PST.1

7.5.3.6. Protection of Stored TSF Data Related to Authentication Tokens (Cryptographic Keys, Critical Security Parameters)

Table 68. Protection of Stored TSF Data Related to Authentication Tokens (Cryptographic Keys, Critical Security Parameters)

Category	TSF Data Related to Authentication Tokens	Remarks
Data Encryption Key	DEK	The Data Encryption Key (DEK) is encrypted using the symmetric key encryption algorithm ARIA/CBC (128-bit) with the KEK and stored as an internal file of TOE.

Related SFR : FPT_PST.1

7.5.4. TSF Self-Test

The SSO server and SSO agent perform TSF self-tests and verify the integrity of configuration files and TSF execution code. Self-tests, library and configuration file integrity checks, cryptographic module self-tests and integrity verification are executed during system startup and subsequently performed periodically. In addition, authorized administrators can manually initiate verification via the administrator web browser. If a self-test fails, the event is logged in the audit log and a notification email is sent to the administrator.

Table 69. Items Subject to TOE Self-Tests

Category	Item	Description (Role)
SSO Server	Self-Test	Verify the operation of the SSO server process
	Library and Configuration File Integrity Check	Verify integrity of library and configuration files (HMAC-SHA256)
	Cryptographic Module Self-Test and Integrity Check	Perform self-test and integrity check of cryptographic module (HMAC-SHA256)
SSO Agent	Self-Test	Verify the operation of the SSO agent process
	Library and Configuration File Integrity Check	Verify integrity of library and configuration files (HMAC-SHA256)
	Cryptographic Module Self-Test and Integrity Check	Perform self-test and integrity check of cryptographic module (HMAC-SHA256)

The self-tests of the SSO server and SSO agent include verifying process operation. In the process operation verification step, the SSO server checks whether its process is running.

Integrity checks are performed on all files within the {SSO server/agent context root}/ directory, except for certain specified files. The integrity verification key is generated using the HASH_DRBG (SHA-512) algorithm, and the integrity verification value for each file is generated using the HMAC-SHA256 algorithm. During verification, the same integrity verification key and the HMAC-SHA256 algorithm are used to calculate the current integrity value, which is then compared with the previously stored integrity value to confirm integrity. The integrity verification method, as well as the list of excluded files and the reasons for their exclusion, are as follows.

Table 70. Integrity Verification Methods

Category	Item	Algorithm
SSO Server	Self-Test	HMAC-SHA256
	Libraries	HMAC-SHA256
	Cryptographic Module	HMAC-SHA256
SSO Agent	Self-Test	HMAC-SHA256
	Libraries	HMAC-SHA256
	Cryptographic Module	HMAC-SHA256

Table 71. Files Excluded from Integrity Verification

Category	Item
SSO Server	File storing the integrity values of the SSO server
	Backup files of the SSO server's configuration
	Configuration files of the SSO server that are deleted after their contents are stored in the DBMS upon initial startup
SSO Agent	File storing the integrity values of the SSO agent

Related SFR: FPT_TST.1

7.6. TOE Access

7.6.1. TOE Access

If an authorized administrator or user fails to log in to the TOE and exceeds the configured number of allowed login failures, login using the same account will be blocked for the specified lockout period.

For super administrators and general administrators who are authorized to change TOE access settings, the number of simultaneous logins is limited to one. However, administrators with monitoring authority only are allowed to log in concurrently. Administrator sessions are accessible on a restricted basis according to the maximum number of allowed IP addresses (1–50, default 2), and at least one allowed IP address must be configured and enforced. Settings that cover the entire network range are not permitted.

Table 72. Unconfigurable IP address

Category	Item	Reason for Exclusion
Unconfigurable IP address	*	A wildcard value covering all IPv4 addresses, making it impossible to identify a specific IP.
	0.0.0.0	Represents “all IPv4 addresses” of the local host or an unbound state, making communication with a specific IP impossible.
	255.255.255.255	An IPv4 broadcast address, used to send packets to all hosts within the network.
	An IP address with the first octet set to 0	Reserved to indicate the current network, and cannot be used as an individual host address.
	An IP address with the last octet set to 0	Reserved as a subnet network address, and cannot be assigned as a host IP.

If multiple logins are attempted to the SSO server using the same ID, the previous session will be terminated, and a duplicate login alert will be sent to the authorized administrator via email.

If a general user or administrator remains inactive beyond the timeout period(60–600 seconds, default 600 seconds) configured by an authorized administrator, the session will be terminated,
.

Related SFR: FTA_MCS.2, FTA_SSL.3, FTA_TSE.1(1)

7.7. Secure Path/Channel (FTP)

The TOE uses SSL to provide a secure communication path/channel that protects data transmitted when interfacing with external IT entities.

One such external IT entity used by the TOE is the SMTP server for sending notification emails. Communication between the SSO server and this external IT entity is secured by establishing a secure channel using the TLS V1.3 protocol with the TLS_AES_256_GCM_SHA384 cipher suite and the ECDHE key exchange algorithm with the curve specified in the Key Share Entry.

Related SFR: FTP_ITC.1