(Document Code : CCV8-STE-1.9)

V-FRONT v8

Security Target

1.9

Revision History

Date	Remark	Author	Version
2024-09-03	Initial version for V-FRONT v8	YS Han	1.0
2025-06-18	Updated changes according to CC 2022 R1	YS Han	1.1
2025-07-10	Reflected Korean National Protection Profile for Single Sign-On v3.1(25.6.27)	YS Han	1.2
2025-08-08	Updated changes according to CC pre-inspection	YS Han	1.3
2025-08-21	Updated for cryptographic module and other modifications	SH Seok	1.4
2025-09-09	Reflection of error corrections	YS Han	1.5
2025-09-15	Reflection of error corrections	YS Han	1.6
2025-10-22	Reflection of error corrections	YS Han	1.7
2025-12-02	Reflection of error corrections	YS Han	1.8
2026-01-05	Reflection of error corrections	YS Han	1.9

Table of Contents

1. ST Introduction

1.1. ST reference

Developers and evaluators can identify this document and verify its assurance for the TOE through the following:

Item	Details
Title	V-FRONT v8 Security Target
Vesion	1.9
Author	R&D Team of AirCUVE Co., LTD.
Date	2026. 1. 5
Document Code	CCV8-STE-1.9
Evaluation Assurance Level	EAL1+ (ATE_FUN.1)
Protection Profile	Korean National Protection Profile for Single Sign-On V3.1(2025.6.27)
Keywords	SSO, Single Sign-On

1.2. TOE reference

The TOE identified through this Security Target includes the following references:

Item		Details
TOE		V-FRONT v8
Version		8.1.1.3
TOE Components	SSO Server	V-FRONT v8 Server 8.1.1.2 (filename: V-FRONTv8_Server_setup-8.1.1.2.x86_64.bin)
	SSO Agent	V-FRONT v8 Agent 8.1.1.2 (filename: V-FRONTv8_Agent_setup-8.1.1.2.x86_64.bin)
Documents		V-FRONT v8 Operational User Guidance 1.9 (filename: CCV8-OPE-1.9_EAL1+.pdf)
Developer		R&D Team of AirCUVE Co., LTD.

1.3. TOE overview

1.3.1. Single Sign On overview

V-FRONT v8 (hereinafter referred to as "TOE") is used to enable the user to access various business systems and use the service through a single user login without additional login action. The TOE performs user identification and authentication, authentication token issuance, and validation functions according to user authentication policies.

The TOE provides the end user login function using an ID/PW-based authentication method in conjunction with an external authentication system, issues an authentication token during end user login, and verifies the issued authentication token when accessing another business system after user login.

For administrators, an authentication function based on ID and password is provided. Additionally, an OTP-based authentication function is provided after ID/PW-based authentication. For end users, an external authentication system provides the authentication function at the initial authentication phase of Single Sign-On, so it is not included in the TOE.

The primary security features provided by the TOE include user identification and authentication, token issue, storage, verification and destruction. During the generation of authentication token and user Single Sign-On based on the authentication token, the TOE uses the MPowerCrypto V3.0, a validated cryptographic module whose security and implementation conformance are validated by the Korea Cryptographic Module Validation Program (KCMVP).

1.3.2. TOE type and scope

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

The TOE components are V-FRONT v8 Server (hereinafter referred to as the "SSO Server") and the V-FRONT v8 Agent (hereinafter referred to as the "SSO Agent"). The TOE is composed of the SSO server that performs functions such as user login processing,

authentication token management, and policy settings, etc., and the SSO agent that is installed in each business system performs functions such as authentication token issuance and verification, etc. In addition, the agent is provided in "process type".

1.3.3. TOE usage and major security features

The TOE performs user identification and authentication to enable the end user to access various business systems and use the service through a single user login without additional login action, and the TOE is supported by user identification and authentication that the external authentication systems provide. The support by the external authentication system, however, is only allowed for the authorized end user.

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

In addition, the token requires confidentiality and integrity protection, and the TOE executable code requires integrity protection.

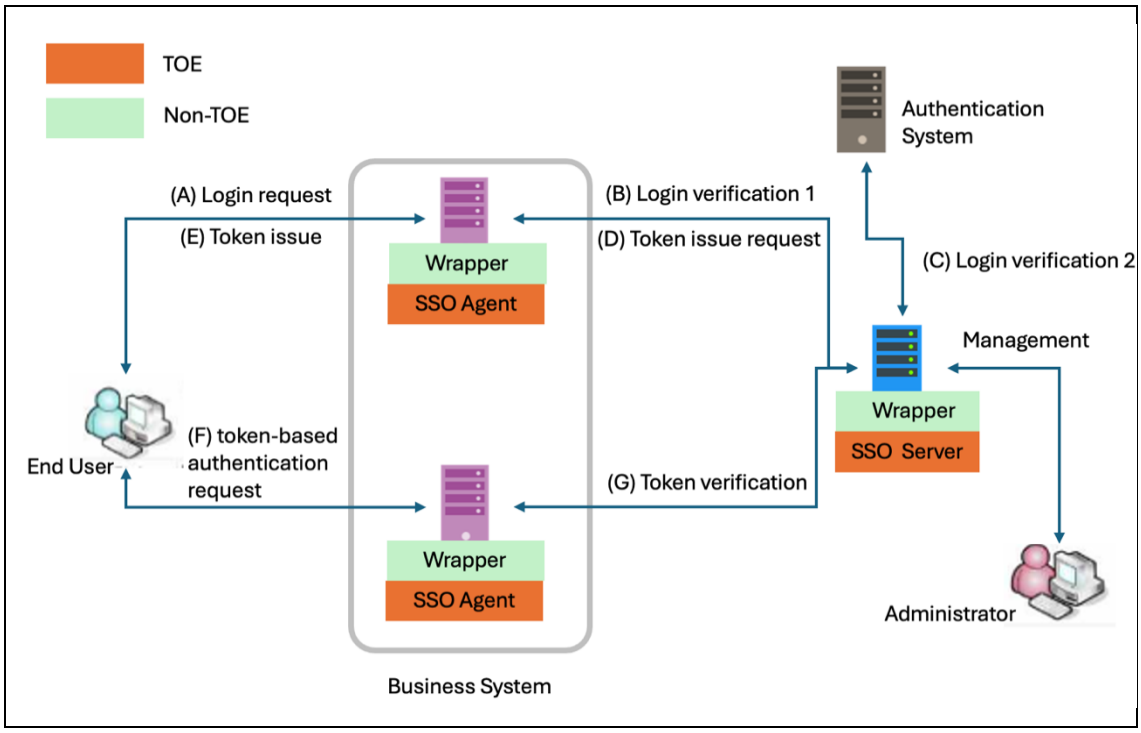
Figure 1-1 shows the user identification and authentication procedure of the TOE.

The user identification and authentication procedure can be grouped into the initial authentication phase using ID/PW, and the token-based authentication phase that accesses the business system using the token issued during the initial authentication procedure.

The initial authentication process begins with a user requesting login using their ID/PW. The SSO agent, upon receiving the login request, sends a login verification request to the

SSO server to verify the user. The SSO server, upon receiving the login verification request, performs login verification through external authentication system.
If the login verification result is valid, the SSO server requests the SSO agent to issue an authentication token. The SSO agent then delivers the issued token to the end user.

The token-based authentication phase is performed only when the token has been normally issued in the initial authentication phase. When the user utilizes business system services, the issued token is transferred to the SSO agent installed in the pertinent business system, and the SSO agent verifies the validity of the token by interfacing with the SSO server upon receiving the token.



[Figure 1-1] User identification and authentication procedure

The user identification and authentication procedure operate as shown in the following Table 1-1, depending on the TOE implementation method.

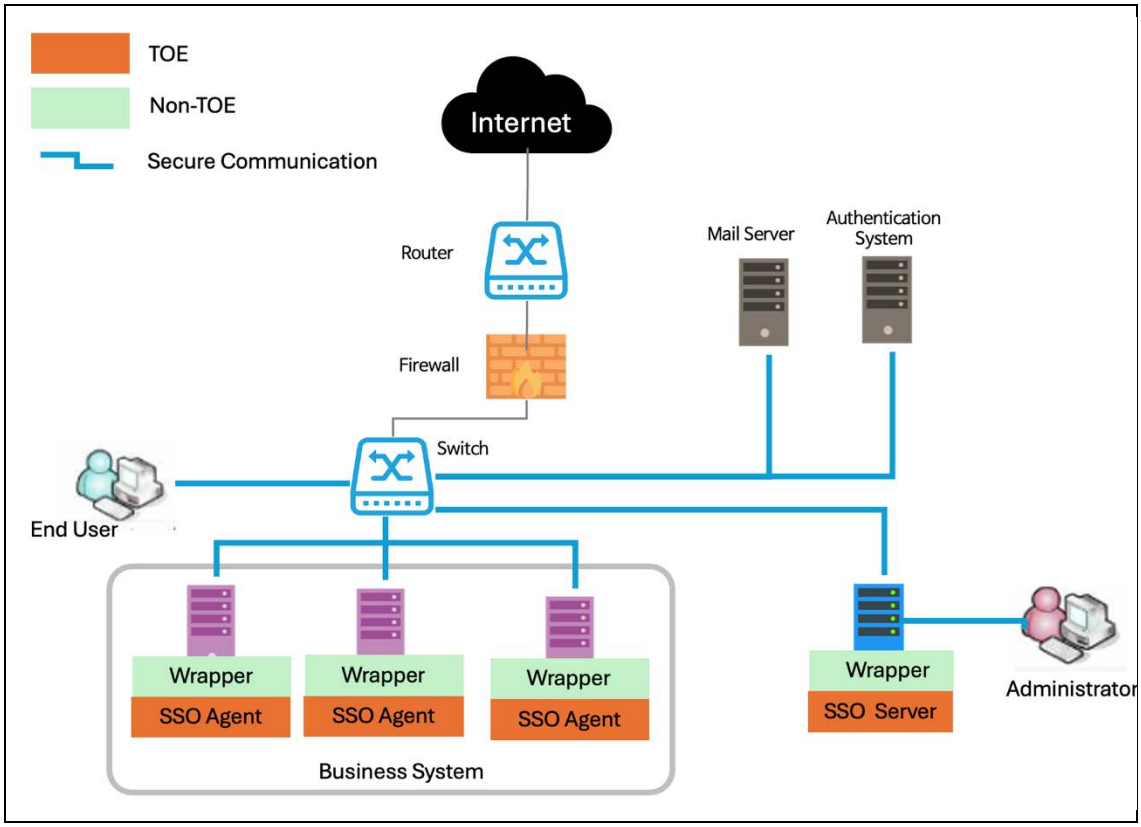
authentication phase	operation procedure
initial authentication	(A) login request 1 ⟶ (B) login verification 1 ⟶ (C) login verification 2 ⟷ (D) token issue request ⟶ (E) token issue
token-based authentication	(F) token-based authentication request ⟶ (G) token verification

[Table 1-1] operation procedure by authentication phase

The following is the subject who issues, stores, and verifies the token.

- Subject who issues the token: SSO Agent
- Token storage location: User PC(Web browser)
- Subject who verifies the token: SSO Agent + SSO Server

1.3.4. Non-TOE and TOE operational environment



[Figure 1-2] TOE operational environment

Figure 1-2 is the TOE operational environments and is composed of the SSO server and SSO agent.

The SSO server provides functions which administrator login verification using the administrator information stored in the DBMS, end user login verification trough the external authentication system (authentication servers inside the organization), the token management, and the policy configuration. The SSO agent is installed in each business

system and requests user login verification to the SSO server and issues the token. In addition, the SSO agent is the ‘process type’ composed of the executable file.

Authorized administrators may perform security management by accessing the SSO server through web browser. Wrappers which may be used to support various types of authentication mechanisms or for compatibility with business systems in the TOE operating environment are out of the TOE scope.

Cryptographic communication is performed in the communication section among TOE components, and between external IT entities and TOE components.

External IT entities necessary for the operation of the TOE are email server to notify the authorized administrator in case of audit data loss and to send OTP number for additional authentication, and the authentication system for the end user identification and authentication.

The email server and authentication system except for the TOE correspond to the TOE operational environment.

For FAU_STG.2, which is the conditionally mandatory security functional requirement, in this function is not implemented in the TOE but provide the function using DBMS and accordingly, the security objectives for the operational environment are added OE.SECURED_DBMS.

For FPT_STM.1, which is the optional security functional requirement, in this function is not implemented in the TOE but provide the function by the operating system and accordingly, the security objectives for the operational environment are added OE.TIMESTAMP.

The TOE provide identification and authentication functions by receiving the authentication results of external authentication system with the TOE, and the authentication information used by external IT entities to perform additional identification and authentication methods is safely managed by external IT entities, so the security objectives for the operating environment are added OE.AUTHENTICATION_SYSTEM_SECURITY accordingly.

When users(authorized administrators) access the SSO server, a secure communication path must be provided between the user's web browser and the web server, so OE.SECURED_ADMIN_ACCESS is added to the security objectives for the operating environment. Since transmitting TSF data between the TOE components(the SSO Agent and the SSO server) which are physically separated, FTT_ITT.1 is applied.

Since TOE procedurally specifies the function that allows administrators to intervene and recover information in the event of information tampering through the operating manual, so the security objectives for the operating environment are added OE.MANUAL_RECOVERY accordingly.

The TOE interacts with external IT entities(mail server, authentication system), so the security objectives for the operating environment are added OE.SECURE_CHANNEL accordingly.

The TOE's operating environment identifies the minimum software and hardware required to operate the TOE. The operating environments of each TOE component, the SSO server and SSO agent, are as shown in [Table 1-2] below.

TOE	Item		Specification
V-FRONT v8 Server	H/W	CPU	Intel CPU : Intel Xeon 3.4Ghz, 4 Core or higher
		RAM	8GB or higher
		HDD	Space required for installation of TOE 100GB or higher
		NIC	10/100/1000Mbps Ethernet Port 1EA or higher
	S/W	OS	Ubuntu 22.04 LTS kernel 5.15.0-164-generic (64 bit)
		DBMS	PostgreSQL 17.7
		WAS	Apache Tomcat 10.1.50
V-FRONT v8 Agent	H/W	CPU	Intel CPU : Intel Xeon 3.4Ghz, 4 Core or higher
		RAM	8GB or higher
		HDD	Space required for installation of TOE 10GB or higher
		NIC	10/100/1000Mbps Ethernet Port 1EA or higher
	S/W	OS	Ubuntu 22.04 LTS kernel 5.15.0-164-generic (64 bit)
		WAS	Apache Tomcat 10.1.50
User PC		WEB Browser	Chrome 143.0

[Table 1-2] V-FRONT v8 operating environment

[Figure 1-2] illustrates the IT environment in which the TOE operates. This IT environment is a network environment equipped with an SSO server and SSO agent. In this IT environment, the physical space (server network) within the firewall is assumed to be physically secure, allowing access only by trusted administrators.

The SSO server provides a management environment that manages TSF data such as user (administrator) information, SSO agent information, and SSO server configuration information.
The SSO agent is installed in the business service environment, generates authentication token (Access Token) information issued to users when general users access, and performs the authentication token verification process in conjunction with the SSO server.

The 3rd party S/W included in the TOE operating environment are PostgreSQL and Apache Tomcat
PostgreSQL is a DBMS installed on the SSO server to store management information.
Apache Tomcat is a WAS (Web Application Server) for running SSO Server and SSO agent applications and provides an HTTPS-based web service execution environment.
Chrome is a web browser running on the user's PC that provides an HTTPS environment for communication with the SSO server or SSO agent.

External IT entities are listed in [Table 1-3] below.

External IT entities	Usages
Authentication System	Perform end user identification and authentication
Mail Server	Send OTP for additional administrator authentication
	Send administrator alert massages

[Table 1-3] External IT entities for V-FRONT v8

1.4. TOE description

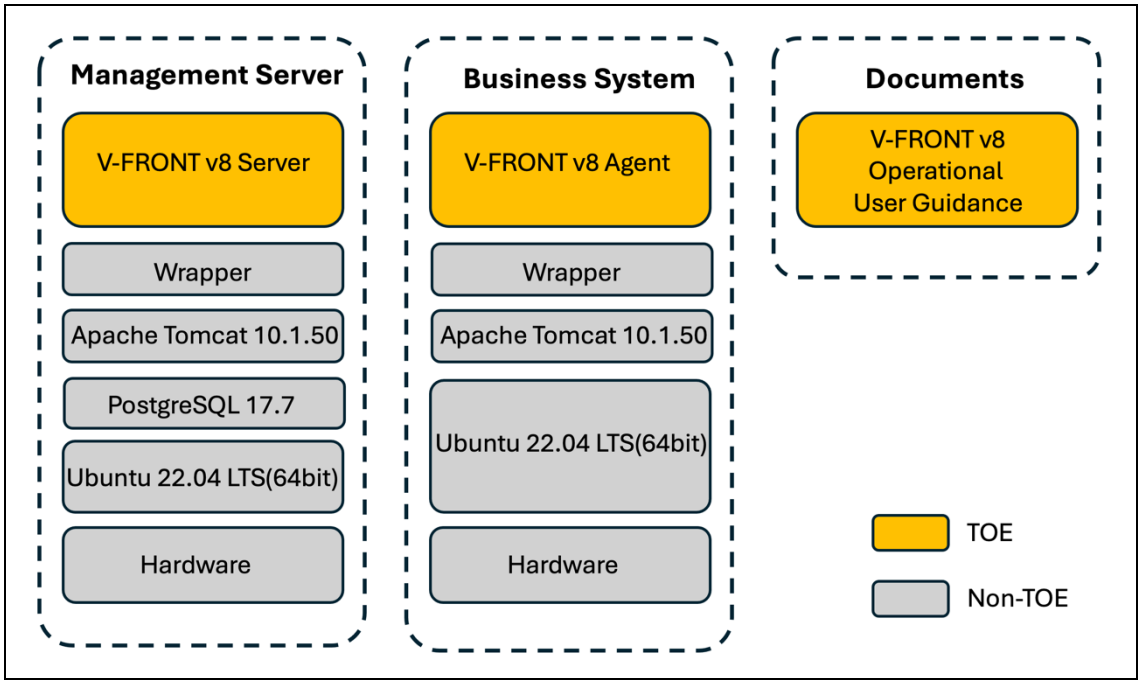
1.4.1 Physical scope of the TOE

The TOE consist of SSO server, SSO agent, Operational User Guidance is installed and operated in software form.

Scope	Distribution Status	Type	Distribute
Name	V-FRONT v8	-	-
Version	8.1.1.3	-	-
TOE Components	V-FRONT v8 Server 8.1.1.2 (V-FRONTv8_Server_setup-8.1.1.2.x86_64.bin) V-FRONT v8 Agent 8.1.1.2 (V-FRONTv8_Agent_setup-8.1.1.2.x86_64.bin)	S/W	CD 1EA/ Personal delivery
Documents	V-FRONT v8 Operational User Guidance 1.9 (CCV8-OPE-1.9_EAL1+.pdf)	file (PDF)	

[Table 1-4] Identification Information of TOE

The physical scope of TOE is as shown in [Figure 1-3].



[Figure 1-3] Physical scope of the TOE

Detailed information about the validated cryptographic module included in the TOE(SSO server, SSO agent) is as follows.

Item	Details
Cryptographic Module Name	MPowerCrypto V3.0
Developer	UBIMINFO. Co., LTD.
Validation Date	2024-06-17
Validation Level	VSL1
Validation Number	CM-249-2029.6
Expiration Date	2029-06-17

[Table 1-5] Detailed information of validated cryptographic module

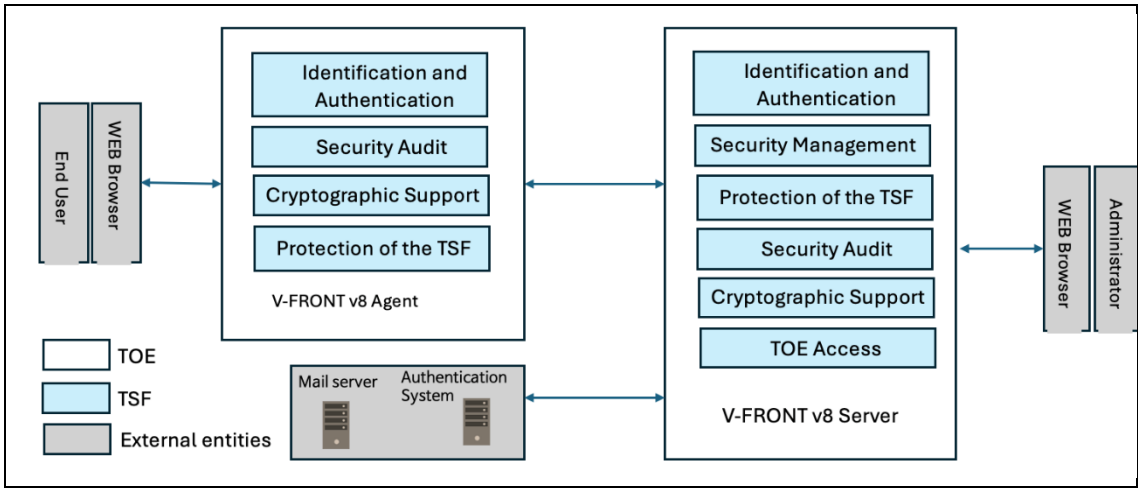
The third-party software required to perform TOE operation and security functions included in TOE is as follows.

TOE	Specification
SSO Server	Amazon Corretto 17 JDK : Used as a JAVA library Spring Boot 3.5.9 : Used as a WEB Application Framework
SSO Agent	Amazon Corretto 17 JDK : Used as a JAVA library Spring Boot 3.5.9 : Used as a WEB Application Framework

[Table 1-6] 3rd Party S/W included in TOE

1.4.2 Logical scope of the TOE

The logical scope of the TOE according to its security functions, each consists of the SSO server (V-FRONT v8 server) and the SSO agent (V-FRONT v8 agent), is as shown in [Figure 1-4] below.



[Figure 1-4] Logical scope of the TOE(V-FRONT v8)

The description of each function included in the TOE area according to the logical scope of V-FRONT v8 is as follows.

• **Identification and authentication**

End users access the business system through a web browser installed on their PCs, and the business system transmits the ID/PW received from the user to the SSO server through the connected SSO agent (V-FRONT v8 Agent), and the SSO server confirms authentication by an external authentication system. At this time, the transmission connection between the SSO server and the SSO agent performs mutual authentication based on custom protocol.

If the end user authentication is successful, the SSO agent generates an authentication token and sends it to the business system, which then stores this token in the user's browser as a cookie. This authentication token contains an authentication session ID and timestamp to prevent reuse.

Afterwards, when the user accesses another business system, the previously issued token is transmitted, and this business system verifies the validity of the token through the connected SSO agent and SSO server, performs Single Sign-On processing.

If token verification is successful, the business system verifies the user based on the token information and allows access to the service according to the business system permissions. If authentication fails, no reason for failure is provided.

The administrator manages the SSO server by accessing the Admin Portal of the SSO server (V-FRONT v8 Server) with WEB browser.

For administrator authentication, identification and authentication are performed based on ID/Password, and additional identification and authentication are performed based on OTP authentication. Authentication data used for authentication will be rejected when reused, and the password information used during authentication is masked (marked with ●) to prevent exposure, and the reason for failure is not provided.

When the number of failed user authentication attempts reaches the number of failed login attempts set by the authorized administrator (default 5, configurable from 1 to 5), the SSO server locks the account for 10 minutes and then automatically unlocks the account.

When using Password for administrator identification and authentication, The SSO Server checks for compliance with password security regulations.

The authentication token generated upon successful authentication of an end user is generated in a form that can ensure sufficient security through a secure algorithm and is destroyed in an irreversible manner (overwriting three times with a value of 0 in the memory area) after transmitting the generated authentication token or verifying the authentication token.

- **Security management**

The SSO server (V-FRONT v8 Server) provides policy settings and permission management for service access by administrators and SSO agents (V-FRONT v8 Agent) by allowing access only to authorized administrators through the login function.
The management function supports the ability to add/modify/delete/query security information for administrator identification and authentication.
Security information is maintained through administrator account information, agent information, agent policy information, logs, and system management (authentication system, mail system connection settings, etc.) functions.
Administrators are classified into read-only and read/write administrators based on their administrative privileges. Read-only administrators can only query data, while read/write administrators can perform add/modify/delete functions in addition to query functions.

The SSO server registers the IP address of the PC for management access and blocks access from unregistered IP addresses.

When installing the SSO server, set the administrator password and the account, and ensure that the password complies with password-related security regulations.

The Service Agent Information Management function manages the settings for trust-based connections with the SSO agent installed on the Single Sign-On target service.

The Policy Information Management manages the security functions to add, query, modify, and delete security policies for administrators of the SSO server.

The Report Management provides administrators with audit logs for security audits.

- **Protection of the TSF**

The SSO server and SSO agent use a shared secret key to ensure confidentiality and integrity of internally transmitting TSF data such as general user authentication request information, agent status information, agent configuration management information, and authentication token verification-related information through mutual connection. Furthermore, TSF data is encrypted using DEK when stored or is securely managed in the DBMS. The TOE maintains a secure state in the event of a failure in the random number generator noise source integrity test of the cryptographic module used.

The SSO server and SSO agent each generate a KEK using the PBKDF2 method using the password entered during installation to protect the encryption key (DEK) for encrypting important stored data such as the administrator password and product settings.

The SSO server performs a self-test to ensure the normal operation of the single sign-on feature and the process status upon startup and at the administrator's request, and the SSO agent performs a self-test upon startup and periodically to check the integrity of the relevant executable file and setting values. If an error occurs, SSO Server and SSO Agent perform response actions. Authorized administrators can review the verification details and results by viewing the logs on the management screen.

When the SSO server and SSO agent are started, normal operation and integrity are checked, and if an abnormality is found, the reason for the failure can be checked on the screen.

- **TOE access**

The SSO server provides a feature to terminate an authorized administrator session after a specified period of inactivity (10 minutes). Furthermore, to prevent duplicate access to the product using the same user account and permissions, it provides a feature to terminate the first session in the event of a duplicate access. Any administrator accessing the SSO server will have their access sessions restricted based on the set allowed IPs.

- **Security audit**

The TOE creates audit records using trusted time information for major audit events and stores them in the DBMS, but does not store information such as authentication passwords or encryption keys.
Provides the ability to detect potential breaches in security-related incidents, and an authorized administrator performs a search function by searching and sorting the

operation logs and user authentication logs through the log management environment of the SSO server according to selected search conditions.
When the audit records size reaches the defined capacity (80%), an administrator is notified, and when the capacity is full (90%), some of the older audit records (10%) are deleted to prevent failure in saving new audit records.

- **Cryptographic support**

The encryption key generated by TOE is derived based on a random number generated by a random bit generation algorithm using the validated cryptographic module, and encryption keys that have been used are safely destroyed.

TOE uses the validated cryptographic module to perform cryptographic key management and cryptographic operations to ensure secure operation and integrated authentication. Additionally, TOE supports encryption related to the storage of important data, issuance, storage, verification, and disposal of authentication tokens, and support encrypted communication (TLSv1.3), and performs encryption key management and encryption operations at this time.

1.5. Terms and definitions

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

- **Admin Portal**

A web-based management environment provided by the SSO server to allow administrators to access and perform management

- **Agent Type1**

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

• **Agent Type2**

Network Access Control products, Patch Management Systems, etc.

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

• **Agent Type3**

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

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

• **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

• **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

- **Client Type**

Virtual Private Network products, Wireless LAN Authentication Products, etc.
- The client is an entity installed on the user's host and serves to request communication with the server on behalf of the user.

- **Component**

Smallest selectable set of elements on which requirements may be based

- **Conditioning**

The process of increasing the entropy rate per bit by removing the bias from collected noise sources

• **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 a PP, ST, functional package or assurance package including a component also includes any other components that are identified as being depended upon or include a rationale as to why they are not

• **Deterministic Random Bit Generator (DRBG)**

It consists of an algorithm that generates a bit string from an initial value called a seed and produces the same bit string when the same seed is input.

• **Element**

Self-contained description of a security need assigned to SAR or SFR

• **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

• **Entropy**

A measure used to evaluate the unpredictability of data.

A numerical representation of the amount of information contained in data.
It represents disorder or randomness, and the closer it is to a random bit, the higher the entropy.

- **Entropy rate**

The entropy of the data divided by the size of the data, expressed as a value between 0 and 1.

- **Entropy source**

A function or device that combines noise sources, health tests, and conditioning algorithms

- **External Entity**

Human technical system or one of its components interacting with the target of evaluation TOE from outside of the TOE boundary

- **Evaluation Assurance Level (EAL)**

Well-formed package of security assurance requirements representing a point on the pre defined assurance scale
Note 1 to entry: EALs are defined in CC Part 5.

- **Family**

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

- **Health test**

Implemented within a random bit generator to monitor noise sources in real time.
The health test is not a process for identifying statistical problems with noise sources; rather, it is a method for detecting cases where the collected noise sources do not operate normally due to equipment aging, etc.
※ For detailed information, refer to the health test defined in Section 5.2 of TTAK.KO-12.0306/R1.

- **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

• **Kerberos**

A centralized authentication scheme, described in RFC 1510, that provides user authentication using symmetric cryptographic technique in a distributed computing environment

• **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

• **Noise Source**

Functions or devices that generate non-deterministic data

• **Object**

Entity in the TOE that contains or receives information, and upon 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 an object)**

〈on an object〉 specific type of action performed by a subject on an object

- **Private Key**

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

- **Protection Profile (PP)**

Implementation-independent statement of security needs for a TOE type

- **Public Key**

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

- **Public Key (asymmetric) cryptographic algorithm**

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

- **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.

※ The cryptographic random bit generator consists of an entropy source used for seed construction and a deterministic random bit generator.

- **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 security component

- **Remote Authentication Dial-In User Services (RADIUS)**

Service to identify and authenticate users by sending information such as user ID, password and IP address to the authentication server when a remote user requests a connection

- **Role**

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

- **Secret Key**

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

- **Secure Sockets Layer (SSL)**

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

- **Security Policy Document**

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

- **Security Target (ST)**

Implementation-dependent statement of security requirements for a TOE based on a security problem definition

- **Security Token (HSM)**

A hardware device implemented to process key generation, electronic signature generation, etc., within the device in order to safely save and store confidential information.

- **Seed**

The secret value used to initialize the random bit generator

- **Selection**

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

- **Shall/must**

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

- **Subject**

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, which is the subject of an evaluation

- **Terminal Access Controller Access Control System (TACACS)**

Authentication protocol that is common for UNIX networks, described in RFC 1492, used by remote access server to send user login passwords to an authentication server

- **Threat Agent**

Entity that has potential to exercise adverse actions on assets protected by the TOE

- **TOE Security Functionality (TSF)**

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

- **Transport Layer Security (TLS)**

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

- **TSF Data**

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

- **User**

As a human technical system or one of its components interacting with the TOE from outside the TOE boundary, the user in the TOE is an authorized administrator and an authorized end user.

※ The types of users related to the SFR are divided into human users and external IT entities. Human users may further be differentiated as local human users, meaning they interact directly with the TOE via TOE devices (e.g. workstations), or remote human users, meaning they interact indirectly with the TOE through another IT product.

- **Validated Cryptographic Module**

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

- **Wrapper**

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

1.6. 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**.

2. Conformance claim

2.1. CC conformance claim

Common Criteria (CC)		Information Technology Security Evaluation Criteria CC:2022 Revision 1 ☐ Part 1: Introduction and General Model, CC:2022 R1 (CCMB-2022-11-001, November 2022) ☐ Part 2: Security Functional Components, CC:2022 R1 (CCMB-2022-11-002, November 2022) ☐ Part 3: Security Assurance Components, CC:2022 R1 (CCMB-2022-11-003, November 2022) ☐ Part 4: Framework for the Specification of Evaluation Methods and Activities, CC:2022 R1 (CCMB-2022-11-004, November 2022) ☐ Part 5: Predefined Packages of Security Requirements, CC:2022 R1 (CCMB-2022-11-005, November 2022) Errata and Interpretation for CC:2022 (Release 1) and CEM:2022 (Release 1), Version 1.1, CCMB-2024-07-002, 2024.07
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)

2.2. Package conformance claim

The Security assurance components package that this Security Target complies with is EAL1, and defines some additional Security assurance components

- ☐ assurance package: EAL1 Augmented (ATE_FUN.1)

2.3. PP conformance claim

This Security Target complies with 'Korean National Protection Profile for Single Sign On V3.1 (2025.6.27)'

- ☐ Conformance type: This security target adheres to the principle of “Strict Compliance to Protection Profile”.
- ☐ PP-Configuration: Compliance with PP-Configuration is not included.

2.4. Conformance claim rationale

Since this Security Target adopts the TOE type, security objectives, and security requirements of the 'Korean National Protection Profile for Single Sign On V3.1(2025.6.27),' its conformance claim is classified as "strict Protection Profile conformance."

Conformance Item	PP	ST
TOE Type	Single Sign-On System	Single Sign-On System
Security problem definition	Assets, Threats, Organizational security policy, Assumptions	conformant, augmented : A.SECURED_ADMIN_ACCESS, A.SECURE_CHANNEL
Security objectives	Security objectives for the operational environment	conformant, augmented : OE.TIMESTAMP, OE.SECURED_DBMS, OE.SECURED_ADMIN_ACCESS, OE.MANUAL_RECOVERY, OE.SECURE_CHANNEL See [Table 4-1]
Security requirements	30 Mandatory SFRs	conformant 30 Mandatory SFRs

	18 Conditionally mandatory SFRs 2 Optional SFRs	6 Conditionally mandatory SFRs 1 Optional SFR

2.4.1. Conformance claim rationale of Security problem definition

Item		PP	ST
Assets		Internal IT resources and services interacting with Single Sign-On	Internal IT resources and services interacting with Single Sign-On
		Important data related to the TOE itself and TOE operation (e.g. TSF data)	Important data related to the TOE itself and TOE operation (e.g. TSF data)
Threats	Unauthorized access	T.SESSION_HIJACK	T.SESSION_HIJACK
		T.RETRY_AUTH_ATTEMPT	T.RETRY_AUTH_ATTEMPT
		T.IMPERSONATION	T.IMPERSONATION
		T.REPLAY	T.REPLAY
		T.WEAK_PASSWORD	T.WEAK_PASSWORD
	Information leak	T.STORED_DATA_LEAKAGE	T.STORED_DATA_LEAKAGE
		T.TRANSMISSION_DATA_DAMAGE	T.TRANSMISSION_DATA_DAMAGE
		T.WEAK_CRYPTO_PROTOCOLS	T.WEAK_CRYPTO_PROTOCOLS
	TOE functionality compromise	T.TSF_COMPROMISE	T.TSF_COMPROMISE
Organizational security policy		P.AUDIT	P.AUDIT
		P.SECURE_OPERATION	P.SECURE_OPERATION
		P.CRYPTO_STRENGTH	P.CRYPTO_STRENGTH
Assumptions		A.PHYSICAL_CONTROL	A.PHYSICAL_CONTROL
		A.TRUSTED_ADMIN	A.TRUSTED_ADMIN
		A.OPERATION_SYSTEM_REINFORCEMENT	A.OPERATION_SYSTEM_REINFORCEMENT
		A.SECURE_DEVELOPMENT	A.SECURE_DEVELOPMENT
		A.AUTHENTICATION_SYSTEM_SECURITY	A.AUTHENTICATION_SYSTEM_SECURITY
		-	A.SECURED_ADMIN_ACCESS
		-	A.SECURE_CHANNEL

2.4.2. Conformance claim rationale of Security objectives

PP	ST
OE.LOG_BACKUP	OE.LOG_BACKUP
OE.PHYSICAL_CONTROL	OE.PHYSICAL_CONTROL
OE.TRUSTED_ADMIN	OE.TRUSTED_ADMIN
OE.OPERATION_SYSTEM_REINFORCEMENT	OE.OPERATION_SYSTEM_REINFORCEMENT
OE.SECURE_DEVELOPMENT	OE.SECURE_DEVELOPMENT
OE.AUTHENTICATION_SYSTEM_SECURITY	OE.AUTHENTICATION_SYSTEM_SECURITY
-	OE.TIMESTAMP
-	OE.SECURED_DBMS
-	OE.SECURED_ADMIN_ACCESS
-	OE.MANUAL_RECOVERY
-	OE.SECURE_CHANNEL

2.4.3. Conformance claim rationale of Security requirements

Class	PP	Remarks	ST	comments
FAU	FAU_ARP.1	Mandatory SFR	FAU_ARP.1	
	FAU_GEN.1	Mandatory SFR	FAU_GEN.1	
	FAU_SAA.1	Mandatory SFR	FAU_SAA.1	
	FAU_SAR.1	Mandatory SFR	FAU_SAR.1	
	FAU_SAR.3	Mandatory SFR	FAU_SAR.3	
	FAU_STG.1	Mandatory SFR	FAU_STG.1	
	FAU_STG.2	Conditionally mandatory SFR	-	OE.SECURED_DBMS
	FAU_STG.4	Conditionally mandatory SFR	FAU_STG.4	
	FAU_STG.5	Conditionally mandatory SFR	FAU_STG.5	
FCS	FCS_CKM.1	Mandatory SFR	FCS_CKM.1	
	FCS_CKM.2	Optional SFR	FCS_CKM.2	

	FCS_CKM.5	Conditionally mandatory SFR	FCS_CKM.5	
	FCS_CKM.6	Mandatory SFR	FCS_CKM.6	
	FCS_COP.1	Mandatory SFR	FCS_COP.1	
	FCS_RBG.1	Mandatory SFR	FCS_RBG.1	Internal seeding – Single source
	FCS_RBG.2	Conditionally mandatory SFR	–	
	FCS_RBG.3	Conditionally mandatory SFR	FCS_RBG.3	
	FCS_RBG.4	Conditionally mandatory SFR	–	
	FCS_RBG.5	Conditionally mandatory SFR	–	
FIA	FIA_AFL.1	Mandatory SFR	FIA_AFL.1	
	FIA_IMA.1(Extended)	Mandatory SFR	FIA_IMA.1(Extended)	
	FIA_SOS.1	Mandatory SFR	FIA_SOS.1	
	FIA_SOS.2	Mandatory SFR	FIA_SOS.2	
	FIA_SOS.3(Extended)	Mandatory SFR	FIA_SOS.3(Extended)	
	FIA_UAU.1	Mandatory SFR	FIA_UAU.2	Hierarchical to
	FIA_UAU.4	Mandatory SFR	FIA_UAU.4	
	FIA_UAU.5	Conditionally mandatory SFR	FIA_UAU.5	Secondary Authentication(OTP)
	FIA_UAU.7	Mandatory SFR	FIA_UAU.7	
	FIA_UID.1	Mandatory SFR	FIA_UID.2	Hierarchical to
FMT	FMT_MOF.1	Mandatory SFR	FMT_MOF.1	
	FMT_MTD.1	Mandatory SFR	FMT_MTD.1	
	FMT_PWD.1(Extended)	Mandatory SFR	FMT_PWD.1(Extended)	
	FMT_SMF.1	Mandatory SFR	FMT_SMF.1	
	FMT_SMR.1	Mandatory SFR	FMT_SMR.1	
FPT	FPT_FLS.1	Mandatory SFR	FPT_FLS.1	
	FPT_ITT.1	Mandatory SFR	FPT_ITT.1	

	FPT_LEE.1(Extended)	Conditionally mandatory SFR	–	OE.AUTHENTICATION_SYSTEM_SECURITY
	FPT_PST.1(Extended)	Mandatory SFR	FPT_PST.1(Extended)	
	FPT_RCV.1	Conditionally mandatory SFR	–	OE.MANUAL_RECOVERY
	FPT_RCV.2	Conditionally mandatory SFR	–	
	FPT_STM.1	Optional SFR	–	OE.TIMESTAMP
	FPT_TST.1	Mandatory SFR	FPT_TST.1	
	FPT_TUD.1(확장)	Conditionally mandatory SFR	–	
FTA	FTA_MCS.2	Mandatory SFR	FTA_MCS.2	
	FTA_SSL.1	Conditionally mandatory SFR	–	
	FTA_SSL.3	Conditionally mandatory SFR	FTA_SSL.3	
	FTA_TSE.1(1)	Mandatory SFR	FTA_TSE.1	
	FTA_TSE.1(2)	Conditionally mandatory SFR	–	
FTP	FTP_ITC.1	Conditionally mandatory SFR	–	OE.SECURE_CHANNEL
	FTP_TRP.1	Conditionally mandatory SFR	–	OE.SECURED_ADMIN_ACCESS

2.5. Reference to evaluation methods/activities

The assurance requirements package that complies with this security target must use the evaluation methods and activities defined in the table below.

The following table contains all packages included in the conforming Protection Profile.

Assurance class	Security assurance components	
Security Target evaluation	ASE_INT.1	ST introduction
	ASE_CCL.1	Conformance claims

Assurance class	Security assurance components	
	ASE_OBJ.1	Security objectives for the operational environment
	ASE_ECD.1	Extended components definition
	ASE_REQ.1	Direct rationale 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

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.

Item	Definition	Description
Unauthorized access	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.
Information leak	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.
TOE functionality compromise	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

Definition	Description

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.

Definition	Description
A.PHYSICAL_CONTROL	The place where SSO agent and SSO server among the TOE components are installed and operated shall be equipped with access control and protection facilities so that only authorized administrator can access.
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.AUTHENTICATION_SYSTEM_SECURITY	If TOE receives the support of the external authentication system (RADIUS, TACACS, Kerberos, or other authentication server within the organization) regarding the initial end user identification and authentication function, the external authentication system shall support the function of storing and managing the authentication information of the authorized end user safely.
A.SECURED_ADMIN_ACCESS	The WEB Server of the TOE operating environment and the WEB Browser of the administrator PC must communicate using a secure path.
A.SECURE_CHANNEL	The TOE must communicate with trusted external IT entities using a secure TLS-based channel.

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

Definition	Description
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 (external log server or separate storage device, etc.) 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 administrator 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 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.
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.AUTHENTICATION_SYSTEM_SECURITY	If TOE receives the support of the external authentication system (RADIUS, TACACS, Kerberos, or other authentication server within the organization) regarding the initial

	end user identification and authentication function, the external authentication system shall support the function of storing and managing the authentication information of the authorized end user safely.
OE.TIMESTAMP	It must be managed to provide a reliable timestamp from the OS used in the TOE.
OE.SECURED_DBMS	Connections to the DB server used in the TOE must be securely managed based on its own defined identification and authentication methods. The DBMS used in the TOE must be securely managed to store audit records and prevent data loss.
OE.SECURED_ADMIN_ACCESS	The WEB Server of the TOE operating environment and the WEB Browser of the administrator PC must communicate using a secure path.
OE.MANUAL_RECOVERY	The TOE must specify in its operating instructions procedurally that information can be restored through administrator intervention in the event of information tampering.
OE.SECURE_CHANNEL	The TOE must perform secure TLS-based communication with trusted external IT entities.

4.2. Security objectives rationale

- Security objectives rationale for operational environment

Operational Environment / Security problem definition	OE.LOG_BACKUP	OE.PHYSICAL_CONTROL	OE.TRUSTED_ADMIN	OE.SECURE_DEVELOPMENT	OE.OPERATION_SYSTEM_REINFORCEMENT	OE.AUTHENTICATION_SYSTEM_SECURITY	OE.TIMESTAMP	OE.SECURED_DBMS	OE.SECURED_ADMIN_ACCESS	OE.MANUAL_RECOVERY	OE.SECURE_CHANNEL
P.AUDIT	●						●	●			
P.SECURE_OPERATION			●							●	
A.PHYSICAL_CONTROL		●									
A.TRUSED_ADMIN	●		●								
A.SECURE_DEVELOPMENT				●							
A.OPERATION_SYSTEM_REINFORCEMENT					●						
A.AUTHENTICATION_SYSTEM_SECURITY						●					
A.SECURED_ADMIN_ACCESS									●		
A.SECURE_CHANNEL											●

[Table 4-1] correspondence with the 'security problem definition' and the 'security objectives for the operational environment'

P.AUDIT is performed by OE.LOG_BACKUP, OE.TIMESTAMP, OE.SECURED_DBMS.
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 are performed to prevent log records from being lost.
OE.TIMESTAMP ensures that log recording time is guaranteed to be reliable by the trusted operating environment.
OE.SECURED_DBMS ensures that DB server of a trusted operating environment is used for storing and retrieving log records.

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

OE.MANUAL_RECOVERY ensures the secure operation of the TOE in case the TOE is tampered with.

A.PHYSICAL_CONTROL is supported by OE.PHYSICAL_CONTROL.
OE.PHYSICAL_CONTROL 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 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 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 guidence provided with the TOE so that the security functions of the TOE can be applied safely.

A.OPERATION_SYSTEM_REINFORCEMENT is supported by
OE.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.AUTHENTICATION_SYSTEM_SECURITY is supported by
OE.AUTHENTICATION_SYSTEM_SECURITY.
OE.AUTHENTICATION_SYSTEM_SECURITY guarantees that if End user identification and authentication functions are supported by external authentication systems (e.g., RADIUS, TACACS, Kerberos, and other authentication servers within the organization) in the initial authentication stage, the external authentication system supports the ability to store and manage authentication information of the authorized end user.

A.SECURED_ADMIN_ACCESS is supported by OE.SECURED_ADMIN_ACCESS.
OE.SECURED_ADMIN_ACCESS guaranteed that a secure path is guaranteed when accessing the WEB Browser of the administrator PC through the WEB Server provided in the TOE operating environment.

A.SECURE_CHANNEL is supported by OE.SECURE_CHANNEL.
OE.SECURE_CHANNEL uses TLS-based communication with trusted external IT entities in the TOE's operating environment, thus ensuring the assumption of 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.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

Component relationships	
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. FIA_SOS.3 Destruction of Secrets

Component relationships	
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

Component relationships	
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.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

Component relationships	
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 describe the security functional requirements and assurance requirements that a TOE that accepts this security target specification must satisfy.

6.1. Security functional requirements

The security functional requirements included in this ST are derived from CC Part 2 and Chapter 5 Extended Components Definition. The TOE described in this ST satisfies the essential SFRs listed in [Table 6-1].

Functional class	Security functional components		Remarks
FAU	FAU_ARP.1	Security alarms	Mandatory
	FAU_GEN.1	Audit data generation	Mandatory
	FAU_SAA.1	Potential violation analysis	Mandatory
	FAU_SAR.1	Audit review	Mandatory
	FAU_SAR.3	Selectable audit review	Mandatory
	FAU_STG.1	Audit data storage location	Mandatory
	FAU_STG.4	Action in case of possible audit data loss	Conditionally mandatory
	FAU_STG.5	Prevention of audit data loss	Conditionally mandatory
FCS	FCS_CKM.1	Cryptographic key generation	Mandatory
	FCS_CKM.2	Cryptographic key distribution	Optional
	FCS_CKM.5	Cryptographic key derivation	Conditionally mandatory
	FCS_CKM.6	Timing and event of cryptographic key destruction	Mandatory
	FCS_COP.1	Cryptographic operation	Mandatory
	FCS_RBG.1	Random bit generation (RBG)	Mandatory
	FCS_RBG.3	Random bit generation (Internal seeding – Single source)	Conditionally mandatory
FIA	FIA_AFL.1	Authentication failure handling	Mandatory

Functional class	Security functional components		Remarks
	FIA_IMA.1	TOE Internal mutual authentication (Extended)	Mandatory
	FIA_SOS.1	Verification of secrets	Mandatory
	FIA_SOS.2	TSF generation of secrets	Mandatory
	FIA_SOS.3	Destruction of secrets (Extended)	Mandatory
	FIA_UAU.2	User authentication before any action	Mandatory
	FIA_UAU.4	Single-use authentication mechanisms	Mandatory
	FIA_UAU.5	Multiple authentication mechanisms	Conditionally mandatory
	FIA_UAU.7	Protected authentication feedback	Mandatory
	FIA_UID.2	User identification before any action	Mandatory
FMT	FMT_MOF.1	Management of security functions behaviour	Mandatory
	FMT_MTD.1	Management of TSF data	Mandatory
	FMT_PWD.1	Management of ID and password (Extended)	Mandatory
	FMT_SMF.1	Specification of management functions	Mandatory
	FMT_SMR.1	Security roles	Mandatory
FPT	FPT_FLS.1	Failure with preservation of secure state	Mandatory
	FPT_ITT.1	Basic internal TSF data transfer protection	Mandatory
	FPT_PST.1	Basic protection of stored TSF data (Extended)	Mandatory
	FPT_TST.1	TSF testing	Mandatory
FTA	FTA_MCS.2	Per user attribute limitation on multiple concurrent sessions	Mandatory
	FTA_SSL.3	TSF-initiated termination	Conditionally mandatory
	FTA_TSE.1	TOE session establishment	Mandatory

[Table 6-1] Security functional requirements

6.1.1. Security audit (FAU)

6.1.1.1. FAU_ARP.1 Security alarms

Component relationships

Hierarchical to No other components.

Dependencies FAU_SAA.1 Potential violation analysis.

FAU_ARP.1.1 The TSF shall take [Actions in [Table 6-2] for security violations in [Table 6-2]] upon detection of a potential security violation.

Security violations	Actions
V-FRONT v8 Server Integrity Check Failure during Startup	Process termination
V-FRONT v8 Agent Integrity Check Failure during Startup	Process termination, Send email to authorized administrator
V-FRONT v8 Server Self-Test Failure during Startup	Process termination
V-FRONT v8 Agent Self-Test Failure during Startup	Process termination, Send email to authorized administrator
V-FRONT v8 Server Integrity Check Failure on Administrator Request	Send email to authorized administrator
V-FRONT v8 Server Self-Test Failure when request by administrator	Send email to authorized administrator
V-FRONT v8 Agent Periodic Integrity Check Failure	Send email to authorized administrator
V-FRONT v8 Agent Periodic Self-Test Failure	Send email to authorized administrator

[Table 6-2] Actions for security violations

6.1.1.2. FAU_GEN.1 Audit data generation

Component relationships

Hierarchical to No other components.

Dependencies FPT_STM.1 Reliable time stamps

FAU_GEN.1.1 The TSF shall be able to generate audit data 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;
c) [[Table 6-3], [Table 6-4] Audit events].

FAU_GEN.1.2 The TSF shall record within the audit data at least the following information:
a) Date and time of the auditable event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event;
b) For each audit event type, based on the auditable event definitions of the functional components included in the ST, [other audit relevant information included in the additional audit information on [Table 6-3], [Table 6-4]].

Functional component	Audit events	Additional audit information
FAU_ARP.1	Actions taken in response to a potential security violation	recipient identity
FAU_STG.5	Response to failure to save audit records	recipient identity
FCS_CKM.1	Cryptographic key generation failure	
FCS_COP.1	Cryptographic operation failure (including cryptographic operation type)	
FIA_AFL.1	The reaching of the threshold for the unsuccessful user authentication attempts and the actions taken	detail log
FIA_IMA.1 (Extended)	Success or failure of mutual authentication Change authentication method	
FIA_SOS.2	Reject authentication token on TSF	
FIA_SOS.3 (Extended)	Success or failure of SSO authentication token destruction	Reason for failure
FIA_UAU.2	All execution of authentication mechanism User login and logout	Reason for failure

Functional component	Audit events	Additional audit information
FIA_UAU.4	Authentication failure due to detection of attempted reuse of authentication data	
FIA_UID.2	All execution of identification mechanism	Reason for failure
FMT_MOF.1	All changes of TSF data which the management function items in [Table 6-11] specified in FMT_MOF.1.1	Values of changed TSF data
FMT_MTD.1	Security management actions for the 'TSF data list' specified in FMT_MTD.1.1, All changes to the password values, Agent registration status change	Values of changed TSF data
FMT_PWD.1	Setting the initial administrator ID and password	
FPT_TST.1	Execution of self-test for TSF	Failed security function
	Execution of integrity verification of the TOE itself	Components(filename) with failed integrity verification
FTA_MCS.2	User's session locking or termination Response actions when duplicate login attempts of the same account are detected Denial of new sessions based on the limit on the number of concurrent sessions	detail log
FTA_SSL.3	End of user's session	
FTA_TSE.1	Block accessing IP addresses of management PC's	

[Table 6-3] Audit target events on SSO Server

Functional component	Audit events	Additional audit information
FPT_TST.1	Execution of self-test for TSF	Failed security function
	Execution of integrity verification and its results	Components(filename) with failed integrity verification
FAU_GEN.1	Agent start	

[Table 6-4] Audit target events on SSO Agent

6.1.1.3. FAU_SAA.1 Potential violation analysis

Component relationships

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 [Integrity check of FPT_TST.1, Failure of self-test] known to indicate a potential security violation;
b) [None].

6.1.1.4. FAU_SAR.1 Audit review

Component relationships

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 data.

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

6.1.1.5. FAU_SAR.3 Selectable audit review

Component relationships

Hierarchical to No other components.

Dependencies FAU_SAR.1 Audit review

FAU_SAR.3.1 The TSF shall provide the ability to apply [search conditions such as log type and time slot] of audit data based on [AND operation and the result to sort in ascending/descending order by click on the column].

6.1.1.6. FAU_STG.1 Audit data storage location

Component relationships

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

Component relationships

Hierarchical to No other components

Dependencies FAU_STG.2 Protected audit data storage

FAU_STG.4.1 The TSF shall [Notification to the authorized administrator, [None]] if the audit data storage exceeds [80% of the set disk capacity].

6.1.1.8. FAU_STG.5 Prevention of audit data loss

Component relationships

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

Dependencies FAU_STG.2 Protected audit data storage

FAU_STG.5.1 The TSF shall *overwrite the oldest stored audit records*, [send a warning email to the authorized administrator] if the audit data storage is full.

6.1.2. Cryptographic support (FCS)

6.1.2.1. FCS_CKM.1 Cryptographic key generation

Component relationships

Hierarchical to No other components.

Dependencies [FCS_CKM.2 Cryptographic key distribution, or
FCS_CKM.5 Cryptographic key derivation, or
FCS_COP.1 Cryptographic operation]
FCS_CKM.3 Cryptographic key access
[FCS_RBG.1 Random bit generation, or
FCS_RNG.1 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 [cryptographic key generation algorithm in [Table 6-5]] and specified cryptographic key sizes [cryptographic key sizes in [Table 6-5]] that meet the following: [list of standards in [Table 6-5]].

type	standard	algorithm	key sizes	generator	usage
random bit	[TTAK.KO-12.0332-Part1], [TTAK.KO-12.0332-Part2]	HMAC_DRBG (HMAC-SHA-256)	256bit	SSO Server	SSO Agent secret key
random bit	[TTAK.KO-12.0332-Part1], [TTAK.KO-12.0332-Part2]	HMAC_DRBG (HMAC-SHA-256)	256bit	SSO Server SSO Agent	DEK encrypting TSF data (ARIA)
random bit	[TTAK.KO-12.0332-	HMAC_DRBG	256bit	SSO Agent	Authentication token key (ARIA)

	Part1], [TTAK.KO-12.0332-Part2]	(HMAC-SHA-256)			
public key cryptography	[KS X ISO/IEC 18033-2]	RSA	3072bit	SSO Agent	Key pair of Encryption/Decryption Key of Authentication Token Key
digital signature	[KS X ISO/IEC 14888-2]	RSA	3072bit	SSO Agent	Key pair of Digital Signature Key of Authentication Token

[Table 6-5] list of cryptographic key generation

6.1.2.2. FCS_CKM.2 Cryptographic key distribution

Component relationships

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 [RSAES] that meets the following: [KS X ISO/IEC 18033-2].

[Application notes] This is a security feature required to encrypt authentication tokens when transmitting them and transmit the corresponding encryption key along with the encrypted token.

6.1.2.3. FCS_CKM.5 Cryptographic key derivation

Component relationships

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 [KEK for DEK encryption] from [input password] in accordance with a specified key derivation algorithm [PBKDF(HMAC-SHA-256)]] and specified cryptographic key sizes [256bit] that meet the following: [TTAK.KO-12.0334-Part2].

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

Component relationships

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 [destruction target on [Table 6-6]] when *[destruction timing on [Table 6-6]]*.

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 [overwriting the target of destruction in the memory area with 0 three times or calling the key initialization function provided by the validated cryptographic module three times] that meets the following: [None].

cryptographic key name	destruction timing	destruction target (key & key material)
SSO Server KEK	After encryption or decryption is complete	Key, KEK password
SSO Server DEK	After Key storage is complete, When shutting down the SSO Server	Key
SSO Agent Secret Key	[SSO Server] After Key storage is complete, After Key usage is complete	Key

SSO Agent KEK	After encryption or decryption is complete	Key, KEK password
SSO Agent DEK	After Key storage is complete, After encryption or decryption is complete	Key
SSO Agent Secret Key	[SSO Agent] After Key storage is complete, When shutting down the SSO Agent	Key
Authentication Token Key	After encryption or decryption is complete	Key
Key pair of Encryption/Decryption Key of Authentication Token Key	After encryption or decryption is complete, When shutting down the SSO Agent	Private key, Public key
Key pair of Digital Signature Key of Authentication Token	After signing or verification is complete, When shutting down the SSO Agent	Private key, Public key

[Table 6-6] list of cryptographic key destruction

6.1.2.5. FCS_COP.1 Cryptographic operation

Component relationships

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

Standard List	Cryptographic Operation	Algorithm	Key Length	TOE	Description
[KS X 3275] (2019)	Symmetric Key	ARIA CBC PKCS7Padding	256	V-FRONT v8 Server	- encryption/decryption of SSO Server DEK (Data Encryption Key) using SSO Server KEK (Key Encryption Key) - encryption/decryption of sensitive configuration parameters using SSO Server DEK - encryption/decryption of SSO Agent secret key using SSO Server DEK - Encryption/decryption of data transmitted between SSO Agent and SSO Server
[KS X ISO/IEC 10118-3:2001] (2018)	Hash Function	SHA2-256	-	V-FRONT v8 Server	- Storage of TSF Data(Admin Password) using One-way encryption (salt:64bit, iter:1000) - Generation of integrity verification data for TSF data (configuration files, process modules)
[KS X 3275] (2019)	Symmetric Key	ARIA CBC PKCS7Padding	256	V-FRONT v8 Agent	- encryption/decryption of SSO Agent DEK using SSO Agent KEK - encryption/decryption of [encryption key for Authentication token key], [signature key for Authentication token information] using SSO Agent DEK

					- encryption/decryption of configuration files using SSO Agent DEK - encryption/decryption of Agent Secret Key using SSO Agent DEK - Encryption/decryption of data transmitted between SSO Agent and SSO Server - encryption/decryption of Authentication token using Authentication token key
[KS X ISO/IEC 14888-2] (2011)	Asymmetric Key	RSA-PSS(SHA-256)	3072	V-FRONT v8 Agent	Generate and verify digital signature of Authentication token data
[KS X ISO/IEC 18033-2] (2017)	Asymmetric Key	RSAES(SHA-256)	3072	V-FRONT v8 Agent	Encryption/decryption of Authentication token key
[KS X ISO/IEC 9797-2] (2018) [TTAK.KO-12.0330](2018)	Hash Function	HMAC-SHA2-256	256	V-FRONT v8 Agent V-FRONT v8 Server	Mutual authentication and transmission data integrity verification
[KS X 3275] (2019)	Symmetric Key	ARIA CBC PKCS7Padding	256	V-FRONT v8 Agent	Encryption/decryption of data transmitted between SSO Agent and SSO Server
[KS X ISO/IEC 10118-3:2001]	Hash Function	SHA2-256	-	V-FRONT v8 Agent	Generation of integrity verification data for configuration files, process modules

(2018)					

[Table 6-7] list of cryptographic operation

6.1.2.6. FCS_RBG.1 Random bit generation (RBG)

Component relationships

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 [HMAC_DRBG(SHA-256)] in accordance with [TTAK.KO-12.0332-Part1, TTAK.KO-12.0332-Part2 standard] after initialization.

FCS_RBG.1.2 The TSF shall use a *TSF entropy source [as specified in [Table 6-8] Entropy Source Name]* for initialization and reseeding.

FCS_RBG.1.3 The TSF shall update the DRBG state by *reseeding* using a *TSF entropy source [as specified in [Table 6-8] Entropy Source Name]* in the following situations:
— on the condition: [Random bit generation request];
in accordance with [TTAK.KO-12.0332-Part1, TTAK.KO-12.0332-Part2 standard].

Operation Environment (JRE)	Entropy Source Name	Description
JRE 17	SecureRandom.nextBytes()	Encrypted Java random numbers
	System.currentTimeMillis()	system current time
	System.nanoTime()	Process execution time difference
	Runtime.getRuntime().freeMemory()	free memory

[Table 6-8] Entropy Source Name

6.1.2.7. FCS_RBG.3 Random bit generation (internal seeding - single source)

Component relationships

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 DRBG using a *TSF software-based noise source* [as specified in [Table 6-8] Entropy Source Name] with [2^536] bits of min-entropy.

6.1.3. Identification and authentication (FIA)

6.1.3.1. FIA_AFL.1 Authentication failure handling

Component relationships

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]* unsuccessful authentication attempts occur related to [user authentication].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been *met*, the TSF shall [lock the account for 10 minutes].

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

Component relationships

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 [custom protocol] that meets the following: [None].

6.1.3.3. FIA_SOS.1 Verification of secrets

Component relationships

Hierarchical to No other components.

Dependencies No dependencies.

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet [regulation as shown in [Table 6-9] below].

Description	Contents
Compliance	Ensure the length is between 9 and 32 characters
	Contains at least one number, uppercase letter(English), lowercase letter(English), and special character each
Prohibition	Do not set the same password as the user account (ID)
	Prohibition of consecutive repeated input of the same letter/number
	Prohibit sequential input of consecutive letters or numbers on the keyboard (qwer, asdf, abcd, 1234 etc)
	Prohibition of reuse of the password used immediately before

[Table 6-9] Password regulation

6.1.3.4. FIA_SOS.2 TSF Generation of secrets

Component relationships

Hierarchical to No other components.

Dependencies No dependencies.

FIA_SOS.2.1 TSF shall provide a mechanism to generate an authentication token that meet [a session ID value generated using a random number with an encrypted timestamp to ensure uniqueness, issuer, user ID, issue time, expiration time].

FIA_SOS.2.1 TSF shall be able to enforce the use of TSF-generated authentication token for [Single Sign-On of general user].

6.1.3.5. FIA_SOS.3 Destruction of secrets (Extended)

Component relationships

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 [overwriting '0' three time] that meets the following: [None].

6.1.3.6. FIA_UAU.2 User authentication before any action

Component relationships

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** to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that **administrator**.

6.1.3.7. FIA_UAU.4 Single-use authentication mechanisms

Component relationships

Hierarchical to No other components.

Dependencies No dependencies.

FIA_UAU.4.1 TSF shall prevent reuse of authentication data related to [[Table 6-10] Authentication items].

Authentication items	Authentication data to prevent reuse	Description of the anti-reuse mechanism
Authentication token-based authentication	timestamp(start time), expires_at(end time), session_id, status(VALID, EXPIRED, REVOKED, INACTIVE, LOGGED_OUT)	The SSO server maintains authentication session information for integrated authentication of successfully authenticated end users. The SSO agent performs session validation (whether to terminate or reuse) by requesting token validation from the SSO server.
ID/PW based authentication (Include OTP)	timestamp(start time), expires_at(end time), session_id,	When the SSO server performs administrator access authentication, it generates an authentication session ID

	status(VALID, EXPIRED, REVOKED, INACTIVE, LOGGED_OUT)	for the successfully authenticated session and sets a validity period. The SSO server changes the status value when the administrator session is terminated, so that it can check whether the session has been terminated or reused when requesting authentication in the future.

[Table 6-10] List of anti-reuse mechanism

6.1.3.8. FIA_UAU.5 Multiple authentication mechanisms

Component relationships

Hierarchical to No other components.

Dependencies No dependencies.

FIA_UAU.5.1 The TSF shall provide [password authentication mechanism, list of additional authentication mechanisms on [Table 6-10]] to support administrator authentication.

FIA_UAU.5.2 The TSF shall authenticate any administrator's claimed identity according to the [process of multiple authentication mechanism on [Table 6-10]].

type	additional authentication mechanism	process of multiple authentication mechanism
OTP based	enter the OTP Number	After ID/PW authentication, OTP is sent using the user's email address.

[Table 6-11] list of additional authentication mechanism

6.1.3.9. FIA_UAU.7 Protected authentication feedback

Component relationships

Hierarchical to No other components.

Dependencies FIA_UID.1 Timing of identification

FIA_UAU.7.1 The TSF shall provide only [masking of password input (●)] to the authorized general user while the authentication is in progress.

6.1.3.10. FIA_UID.2 User identification before any action

Component relationships

Hierarchical to FIA_UID.1 Timing of identification

Dependencies No dependencies.

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

6.1.4. Security management (FMT)

6.1.4.1. FMT_MOF.1 Management of security functions behaviour

Component relationships

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 [as specified list of Security Management Function on [Table 6-11]] to [authorized administrators].

Category	Security Management	Reference(TOE TSF)
Identification and Authentication	Administrator registration, deletion, modification, and authorization	Admin-Administrator Management(Add, Modify, Delete, Search)
	Administrator password regulation and length policy settings	Admin-Setting-Admin Info(Password regulation check)
	Set the number of times administrator authentication failures are allowed	Admin-Setting-Admin login policy(max failure count)

	Setting up authentication information for external IT entities authenticated by TOE	System-Authentication Server
Security Management	IP registration, deletion, and modification for management PC	Admin-Setting-Global Setting(Admin Portal Access Policy-Specific IP Allowed)
	Backup important data, configuration information, audit records, etc.	Admin Portal(/opt/aircuve/vfrontv8)
	Restore important data, configuration information, audit records, etc.	Admin Portal(/opt/aircuve/vfrontv8)
	Agent search - status, version, applied security policy	Agent Management(Agent Add/List)
	Agent security setting -create policy, commit policy	Agent Policy Management(Agent Policy Add/List)
	Authentication Token Session-Session Status, logout	Session Management- Session Status, logout
	Setting up authentication information for accessing external IT entities	System-Config-Email setting(Email server & account Address)
Self Protection	Performing self-test for verification of the TOE security functions by the administrator's request	System-Config-Server Self-Test & Integrity Verification
	Setting response actions when self-test fails	System-Config-Warning Notifier Setting
	Performing an integrity verification of the TOE configuration values and the TOE itself by the administrator's request	System-Config-Server Self-Test & Integrity Verification
	Setting response actions when integrity verification fails	System-Config-Warning Notifier Setting
Update protection	Search TOE version information	System-Config-Server Info(version) Agent Management-Version

Trusted session management	Terminate user session, set session timeout	Admin-Setting- Global Setting(Session Time)
	setting number of session for the same account	Allows only one(1) session per user
Audit records	search audit records	Report-Search
	Settings related to response to audit record loss	Checks every 10 minutes and deletes 10% of log records when the disk threshold (90%) is exceeded.

[Table 6-12] Security Management Function

6.1.4.2. FMT_MTD.1 Management of TSF data

Component relationships

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*** [list of TSF data on [Table 6-13]] to [the authorized identified roles on [Table 6-13]].

TOE Component	TSF data	Actions	Authorized roles
SSO Server	Version	search	read-only administrator read/write administrator
	Agent Management	search	read-only administrator
		search, add, modify, delete	read/write administrator
	Agent Policy Management	search	read-only administrator
		search, add, modify, delete	read/write administrator
	Session Management	search	read-only administrator
		search, Session logout	read/write administrator
	logs	search	read-only administrator read/write administrator
	Admin	search	read-only administrator

TOE Component	TSF data	Actions	Authorized roles
		search, add, modify, delete	read/write administrator
	System Setting	search	read-only administrator
		Search, add, modify, delete	read/write administrator

[Table 6-13] Management of TSF data

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

Component relationships

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 [None] to [the authorized administrator].

1. [None]
2. [None]

FMT_PWD.1.2 The TSF shall restrict the ability to manage the ID of [None] to [the authorized administrator].

1. [None]
2. [None]

FMT_PWD.1.3 The TSF shall provide the capability for *setting ID and password when installing*.

6.1.4.4. FMT_SMF.1 Specification of Management Functions

Component relationships

Hierarchical to No other components.

Dependencies No dependencies.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions: [items specified in FMT_MOF.1 Security Function Management, items specified

in FMT_MTD.1 TSF Data Management, items specified in FMT_PWD.1 ID and Password Management].

6.1.4.5. FMT_SMR.1 Security roles

Component relationships

Hierarchical to No other components.

Dependencies FIA_UID.1 Timing of identification

FMT_SMR.1.1 The TSF shall maintain the roles [read-only administrator, read/write administrator].

FMT_SMR.1.2 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

Component relationships

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: [when the noise source integrity test fails, the state transitions to a critical error]

6.1.5.2. FPT_ITT.1 Basic internal TSF data transfer protection

Component relationships

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)

Component relationships

Hierarchical to No other components.

Dependencies No dependencies.

FPT_PST.1.1 The TSF shall protect [[Table 6-14] List of TSF data] stored in containers controlled by the TSF from the unauthorized *disclosure*.

TOE Component	List of TSF data to be encrypted and stored in a file	List of TSF data stored in DB
SSO Server	Setting values (DB connection information), integrity original hash value	(Encryption) Administrator authentication data, email server settings, agent secret key, and external authentication server setting values
		(Access Control) Audit log, TOE configuration value(Security policy, Config values, etc)
SSO Agent	Setting values (agent secret key, server connection information, token digital signature key pair, token information encryption/decryption key pair), and integrity original hash value, DEK.	

[Table 6-14] List of TSF data

6.1.5.4. FPT_TST.1 TSF testing

Component relationships

Hierarchical to No other components.

Dependencies No dependencies.

FPT_TST.1.1 The TSF shall run a suite of the following self-tests *[Execution method in [Table 6-15]]* to demonstrate the correct operation of *the TSF*]: [Self-test items in [Table 6-15]].

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*.

TOE	Self-test items	Execution method	Testing target
SSO Server	validated cryptographic module self-test	at the start-up, at the administrator's request	/opt/aircuve/vfrontv8/sso/mgmt_data/MPowerCrypto3.0.jar
	process test	at the start-up, at the administrator's request	/opt/aircuve/vfrontv8/sso/mgmt_ui/WEB_UI.war /opt/aircuve/vfrontv8/sso/mgmt_data/WEB_DATA.war /opt/aircuve/vfrontv8/sso/mgmt_data/mods/.checkSum.jar
	integrity check	at the start-up, at the administrator's request	/opt/aircuve/vfrontv8/.key/ /opt/aircuve/vfrontv8/ext/ /opt/aircuve/vfrontv8/sso/mgmt_ui/ /opt/aircuve/vfrontv8/sso/mgmt_data
SSO Agent	validated cryptographic module self-test	at the start-up, periodically	/opt/aircuve/vfrontv8/module/agent/lib/MPowerCrypto3.0.jar
	process test	at the start-up, periodically	/opt/aircuve/vfrontv8/module/agent/agent-0.0.1.jar
	integrity check	at the start-up, periodically	/opt/aircuve/vfrontv8/ext/jdk /opt/aircuve/vfrontv8/module/agent/

[Table 6-15] list of the Self-test

6.1.6. TOE access (FTA)

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

Component relationships

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 belong to the same user according to the rules [limiting the maximum number of concurrent sessions to 1 for users who have the same privilege and the same user, [limiting the maximum number of concurrent sessions to 1 for Same user with read permission]].

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

6.1.6.2. FTA_SSL.3 TSF-initiated termination

Component relationships

Hierarchical to No other components.

Dependencies FMT_SMR.1 Security roles

FTA_SSL.3.1 The TSF shall terminate an interactive session after a [administrator-configurable time(120-600 seconds) interval of user inactivity].

6.1.6.3. FTA_TSE.1 TOE session establishment

Component relationships

Hierarchical to No other components.

Dependencies No dependencies.

FTA_TSE.1.1 The TSF shall be able to deny the administrator's management access session establishment based on [access IP, *[whether an administrative access session of the same account is active, or whether an administrative access session of read/write administrator account with the same privileges is active]*].

6.2. Security assurance requirements

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

Assurance class	Security assurance components	
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	Direct rationale security requirements
	ASE_TSS.1	TOE summary specification
Development	ADV_FSP.1	Basic functional specification
Guidance documents	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
Life-cycle support	ALC_CMC.1	Labelling of the TOE
	ALC_CMS.1	TOE CM coverage
Tests	ATE_FUN.1	Functional testing
	ATE_IND.1	Independent testing - conformance
Vulnerability assessment	AVA_VAN.1	Vulnerability survey

[Table 6-16] Security assurance requirements

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 summarize the usage and major security features of the TOE.

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

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

ASE_INT.1.7C For 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 Direct rationale 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 identify the edition of the CC to which the ST and the TOE claim conformance.

ASE_CCL.1.2C The 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 conformance claim shall describe the conformance of the ST as either “CC Part 3 conformant” or “CC Part 3 extended”.

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

ASE_CCL.1.5C The conformance claim shall identify a PP-Configuration, or 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 any conformance of the ST to a 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 PP-Configuration or 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 PP-Configuration1, PPs and any functional packages 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 PP-Configuration2, PPs, and any functional package 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 PP-Configuration, PPs, and any functional packages for which conformance is being claimed.

ASE_CCL.1.12C The conformance claim for PP(s) or a PP-Configuration shall be exact, strict, or demonstrable 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 that shall be used to evaluate the TOE then this set shall include all those that are included in any package, PP, or PP-Module in a PP-Configuration to which the ST claims conformance, and no others.

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 can 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 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 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 controls 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 unique 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 action elements

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

6.2.5. Tests

6.2.5.1. ATE_FUN.1 Functional testing

Dependencies ATE_COV.1 Evidence of coverage

Developer action elements

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

ATE_FUN.1.2D The developer shall provide test documentation.

Content and presentation elements

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

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

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

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

Evaluator action elements

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

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

The rationale for the security functional requirements demonstrates the following:

- ☐ Each threat and organizational security policy is addressed by at least one security functional requirement.
- ☐ Each security functional requirement is traced to at least one threat or organizational security policy.

Security problem definition (SPD) / 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									●			
FAU_GEN.1										●		
FAU_SAA.1									●			
FAU_SAR.1										●		
FAU_SAR.3										●		
FAU_STG.1										●		
FAU_STG.4										●		
FAU_STG.5										●		
FCS_CKM.1						●	●	●				●
FCS_CKM.2						●	●	●				●
FCS_CKM.5						●	●	●				●
FCS_CKM.6						●	●	●				●
FCS_COP.1						●	●	●				●
FCS_RBG.1						●	●	●				●
FCS_RBG.3						●	●	●				●
FIA_AFL.1		●	●						●			
FIA_IMA.1			●									
FIA_SOS.1					●							
FIA_SOS.2			●	●								
FIA_SOS.3	●		●									
FIA_UAU.2			●						●			
FIA_UAU.4			●	●					●			
FIA_UAU.5			●						●			
FIA_UAU.7			●		●				●			
FIA_UID.2			●						●			
FMT_MOF.1							●		●		●	

Security problem definition (SPD) / 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
FMT_MTD.1							●		●		●	
FMT_PWD.1					●				●		●	
FMT_SMF.1									●		●	
FMT_SMR.1									●		●	
FPT_FLS.1						●	●	●				●
FPT_ITT.1							●					
FPT_PST.1						●						
FPT_TST.1						●	●	●	●			●
FTA_MCS.2	●											
FTA_SSL.3	●											
FTA_TSE.1	●											

[Table 6-17] correspondence with the SPD and the SFRs

SPD	SFRs	content
T.SESSION_HIJACK	FIA_SOS.3, FTA_MCS.2, FTA_SSL.3, FTA_TSE.1	FIA_SOS.3 responds to T.SESSION_HIJACK by ensuring safe destruction of the authentication token when the TOE session ends. 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 responds to T.SESSION_HIJACK by ensuring session termination for interactive sessions after a period of inactivity by authorized users. FTA_TSE.1responds to T.SESSION_HIJACK by ensuring that it determines whether to establish an authorized administrator access session based on IP, etc.

T.RETRY_AUTH_ATTEMP	FIA_AFL.1	FIA_AFL.1 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, FIA_IMA.1, FIA_SOS.2, FIA_SOS.3, FIA_UAU.2, FIA_UAU.4, FIA_UAU.5, FIA_UAU.7, FIA_UID.2,	FIA_AFL.1 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 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 and FIA_UAU.5 respond to T.IMPERSONATION by ensuring that users attempting to access the TOE are successfully authenticated. 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 respond to T.IMPERSONATION by ensuring that users attempting to access the TOE are successfully identified.
T.REPLAY	FIA_SOS.2, FIA_UAU.4	FIA_SOS.2 responds to T.REPLAY by ensuring that authentication tokens are not reused when generating authentication tokens. FIA_UAU.4 responds to T.REPLAY by ensuring the ability to prevent reuse of authentication data.
T.WEAK_PASSWORD	FIA_UAU.7, FIA_SOS.1, FMT_PWD.1	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 responds to T.WEAK_PASSWORD by verifying that password complexity rules are satisfied. 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_CKM.6, FCS_COP.1, FCS_RBG.1, FCS_RBG.3, FPT_FLS.1, FPT_PST.1, FPT_TST.1	FCS_CKM.1, FCS_CKM.2, FCS_CKM.5, FCS_RBG.1, FCS_RBG.3, FPT_FLS.1, and 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 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 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 responds to T.STORED_DATA_LEAKAGE by ensuring that the stored TSF data is protected from leakage through encryption and access control.
T.TRANSMISSION_DATA_DAMAGE	FCS_CKM.1, FCS_CKM.2, FCS_CKM.5, FCS_CKM.6, FCS_COP.1, FCS_RBG.1, FCS_RBG.3, FPT_FLS.1, FPT_ITT.1, FPT_TST.1	FCS_CKM.1, FCS_CKM.2, FCS_CKM.5, FCS_RBG.1, FCS_RBG.3, 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 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 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 responds to T.TRANSMISSION_DATA_DAMAGE by ensuring the confidentiality and integrity of transmission data between TOE components.

T.WEAK_CRYPTO_PROTOCOLS	FCS_CKM.1, FCS_CKM.2, FCS_CKM.5, FCS_CKM.6, FCS_COP.1, 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, 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 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 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_SAA.1, FIA_AFL.1, FIA_UAU.2, FIA_UAU.4, FIA_UAU.5, FIA_UAU.7, FIA_UID.2, FMT_MOF.1, FMT_MTD.1, FMT_PWD.1, FMT_SMF.1, FMT_SMR.1, FPT_TST.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 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, FIA_UAU.1(1), FIA_UAU.1(2), FIA_UAU.4, FIA_UAU.5, FIA_UAU.7, FIA_UID.1(1), FIA_UID.1(2) and FPT_LEE.1 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. 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_RCV.1 responds to T.TSF_COMPROMISE by ensuring that the TOE clients can recover modified information when TOE fails self-testing. 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.

P.AUDIT	FAU_GEN.1, FAU_SAR.1, FAU_SAR.3, FAU_STG.1, FAU_STG.4, FAU_STG.5	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 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 satisfies P.AUDIT by providing a selective audit review function based on logical relationship criteria for audit data. FAU_STG.1 satisfies P.AUDIT by providing the ability to store audit data in local DBMS. 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 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_MTD.1, FMT_PWD.1, FMT_SMF.1, FMT_SMR.1	FMT_MOF.1 satisfies P.SECURE_OPERATION by ensuring that only authorized users have the ability to manage security functions. FMT_MTD.1 satisfies P.SECURE_OPERATION by ensuring that only authorized users have the ability to manage the TSF data. 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 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 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, 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 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. Dependency of the security functional requirements

The following table shows dependency of security functional requirements.

No.	SFRs	Dependency	Reference No.
1	FAU_ARP.1	FAU_SAA.1	3
2	FAU_GEN.1	FPT_STM.1	OE.TIMESTAMP * Rationale(1)
3	FAU_SAA.1	FAU_GEN.1	2
4	FAU_SAR.1	FAU_GEN.1	2
5	FAU_SAR.3	FAU_SAR.1	4
6	FAU_STG.1	FAU_GEN.1, FTP_ITC.1	2, OE.SECURED_DBMS * Rationale(2)
7	FAU_STG.4	FAU_STG.2	OE.SECURED_DBMS * Rationale(3)
8	FAU_STG.5	FAU_STG.2	OE.SECURED_DBMS * Rationale(3)
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], * Rationale(4), [14], 12
10	FCS_CKM.2	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5],	[9, 11], * Rationale(4),

No.	SFRs	Dependency	Reference No.
		FCS_CKM.3	
11	FCS_CKM.5	[FCS_CKM.2 or FCS_COP.1], FCS_CKM.6	[13], 12
12	FCS_CKM.6	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5]	[9, 11]
13	FCS_COP.1	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5], FCS_CKM.6	[9, 11], 12
14	FCS_RBG.1	[FCS_RBG.2 or FCS_RBG.3], FPT_FLS.1, FPT_TST.1	[15], 31, 34
15	FCS_RBG.3	FCS_RBG.1	14
16	FIA_AFL.1	FIA_UAU.1	* Rationale(5)
17	FIA_IMA.1	-	-
18	FIA_SOS.1	-	-
19	FIA_SOS.2	-	-
20	FIA_SOS.3	FIA_SOS.2	19
21	FIA_UAU.2	FIA_UID.1	* Rationale(6)
22	FIA_UAU.4	-	-
23	FIA_UAU.5	-	-
24	FIA_UAU.7	FIA_UAU.1	* Rationale(5)
25	FIA_UID.2	-	-
26	FMT_MOF.1	FMT_SMF.1, FMT_SMR.1	29, 30
27	FMT_MTD.1	FMT_SMF.1, FMT_SMR.1	29, 30
28	FMT_PWD.1(	FMT_SMF.1, FMT_SMR.1	29, 30
29	FMT_SMF.1	-	-
30	FMT_SMR.1	FIA_UID.1	* Rationale(6)
31	FPT_FLS.1	-	-
32	FPT_ITT.1	-	-
33	FPT_PST.1	-	-
34	FPT_TST.1	-	-
35	FTA_MCS.2	FIA_UID.1(	* Rationale(6)
36	FTA_SSL.3	FMT_SMR.1	30
37	FTA_TSE.1	-	-

[Table 6-18] Dependency of the SFRs

Rationale(1) : FAU_GEN.1 has a dependency on FPT_STM.1. The TOE uses a reliable timestamp provided by the TOE's operating environment when generating audit records for auditable events, so the requirement is satisfied with OE.TIMESTAMP.

Rationale(2) : FAU_STG.1 has a dependency on FTP_ITC.1, and since audit data is stored in the local storage (DBMS) of the TOE installation environment, it satisfies the requirement of OE.SECURED_DBMS. Therefore, FTP_ITC.1 was not added to the ST.

Rationale(3) : FAU_STG.2, which is dependent on FAU_STG.4 and FAU_STG.5, satisfies the requirement through OE.SECURED_DBMS because audit data is stored through DBMS, which is a trusted operating environment.

Rationale(4): 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.

Rationale(5) : FIA_AFL.1, FIA_UAU.7 have a dependency on FIA_UAU.1, which is satisfied by FIA_UAU.2, which is in a hierarchical relationship with FIA_UAU.1.

Rationale(6) : FIA_UAU.2, FMT_SMR.1 and FTA_MCS.2 have a dependency on FIA_UID.1, which is satisfied by FIA_UID.2, which is in a hierarchical relationship with FIA_UID.1.

6.3.3. 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 ST to allow the developer to independently test whether the TSF has been implemented correctly and whether defects have occurred, etc. and document the results.

6.3.4. 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 ST 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 protection profile 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 ST.
ASE_REQ.1 also includes ASE_SPD.1, which is absent in the EAL1 assurance package due to dependency. However, this direct rationale protection profile 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 ST.

7. TOE Summary specifications

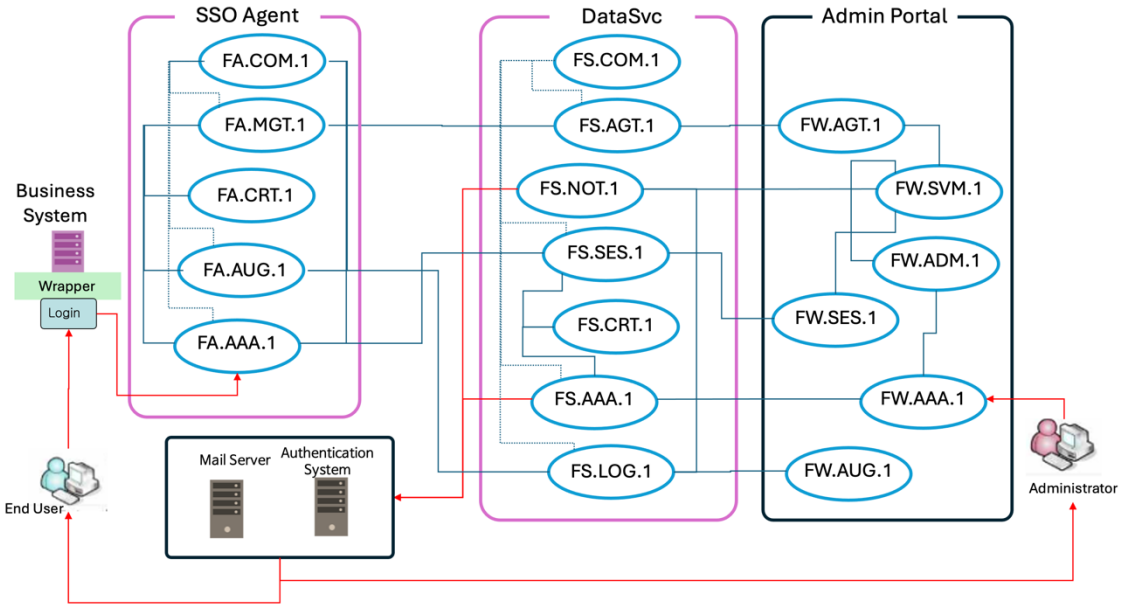
TOE security functions(TSFs) correspond as follows, and the TSFs that satisfy each SFRs are as shown in [Table 7-1].

Classification	SFRs	Implement		TSFs	
		S	A	S(server)	A(Agent)
Security audit	FAU_ARP.1	O		FS.NOT.1	
	FAU_GEN.1	O	O	FS.LOG.1	FA.AUG.1
	FAU_SAA.1	O		FW.AUG.1	
	FAU_SAR.1	O			
	FAU_SAR.3	O			
	FAU_STG.1	O			
	FAU_STG.4	O		FS.NOT.1	
	FAU_STG.5	O			
Cryptographic support	FCS_CKM.1	O	O	FS.CRT.1	FA.CRT.1
	FCS_CKM.2	O	O		
	FCS_CKM.5	O	O		
	FCS_CKM.6	O	O		
	FCS_COP.1	O	O		
	FCS_RBG.1	O	O		
	FCS_RBG.3	O	O		
Identification and authentication	FIA_AFL.1	O		FW.AAA.1 FS.AAA.1	
	FIA_IMA.1	O	O	FS.COM.1	FA.COM.1
	FIA_SOS.1	O		FW.ADM.1	
	FIA_SOS.2		O		FA.AAA.1
	FIA_SOS.3		O		FA.AAA.1
	FIA_UAU.2	O		FW.AAA.1 FS.AAA.1	
	FIA_UAU.4	O	O	FW.AAA.1 FS.AAA.1 FS.SES.1	FA.AAA.1
	FIA_UAU.5	O		FW.AAA.1 FS.AAA.1	

Classification	SFRs	Implement		TSFs	
		S	A	S(server)	A(Agent)
	FIA_UAU.7	O		FW.AAA.1	
	FIA_UID.2	O		FW.AAA.1 FS.AAA.1	
Security management	FMT_MOF.1	O	O	FW.AGT.1 FS.AGT.1 FW.SVM.1 FW.ADM.1 FW.SES.1 FW.AUG.1	FA.MGT.1
	FMT_MTD.1	O	O		
	FMT_PWD.1	O		FW.ADM.1	
	FMT_SMF.1	O			
	FMT_SMR.1	O			
Protection of the TSF	FPT_FLS.1	O	O	FS.CRT.1	FA.CRT.1
	FPT_ITT.1	O	O	FS.COM.1	FA.COM.1
	FPT_PST.1	O		FS.CRT.1	
	FPT_TST.1	O	O	FW.SVM.1	FA.MGT.1
TOE access	FTA_MCS.2	O	O	FW.AAA.1 FS.AAA.1	
				FS.SES.1	FA.AAA.1
	FTA_SSL.3	O	O	FW.AAA.1 FS.AAA.1 FS.SES.1	FA.AAA.1
	FTA_TSE.1	O		FW.AAA.1 FS.AAA.1	

[Table 7-1] Relations of the SFRs and the TSFs

The configuration diagram for TOE components is as shown in [Figure 7-1] below. The SSO Server is composed of subsystems of DataSvc and Admin Portal, and the SSO Agent is composed of subsystems of SSO Agent.



[Figure 7-1] V-FRONT v8 configuration diagram

- **Components**

V-FRONT v8 is an SSO system that identifies and authenticates users requesting access to business systems, granting access rights and authorizations. Access multiple related business systems with the same authority using a token issued through a single authentication process.

This system consists of an SSO server and an SSO agent for linking with the business system, and each component is an application for authenticating and identifying the subject(end user) accessing the object called the business system(typically a web-based business application server).

The business system performs authorization control for the use of each function of the business system based on the identified authentication token(Access Token) information.

- **SSO authentication and authorization token generation, distribution, verification, and destruction for end user**

An SSO server is a security system that allows a user to obtain access rights through a single access authentication process for multiple business systems, and then maintain access rights to multiple business systems for which authentication is integrated without a separate access authentication process (login) within the valid period of access rights.

When a user without an authentication token requests access, the SSO authentication procedure based on ID/PW is performed by the SSO agent installed in the business system, and an authentication token is sent to the terminal of the user who successfully completed the SSO authentication. At this time, the SSO agent generates and encrypts an authentication token and transmits the corresponding access authority information to the end user PC using a response message according to the SSO authentication protocol. At this time, the authentication token key used to encrypt the authentication token is encrypted using an RSAES asymmetric key(authentication token key encryption key) and transmitted together with the authentication token. Once the transmission is complete, the SSO agent securely destroys the authentication token key and authentication token used. The authorization information of the authentication token transmitted to the user browser is included in the access request of the business system and transmitted to the business system, and the SSO agent of the business system connects to the SSO server and performs validation of the authentication token based on the authorization information. When the SSO server receives a request for token verification from the SSO agent, it provides the information(public key) required to verify the token for the corresponding connection session. If token verification is successful through the token verification function of the SSO agent, the business system allows system access according to the user's authorization information. When the session time allowed for the user expires or the user terminates the session or TOE execution, the SSO agent securely destroys the authentication token value loaded in memory by overwriting it three times with the value '0' or calling the key initialization function provided by the validated cryptographic module three times.

- **User**

There are two types of users: the end user and the administrator.
End users access the business system using the WEB browser installed on the user's PC, and the business system performs identification and authentication using the SSO agent. Upon successful authentication, the SSO agent generates an authentication token as a successful authentication response for the authentication session and returns it to the business system. The user is granted access to the business system based on their permissions.
Authorized(administrator identification and authentication) administrator access to the administrator page of the SSO server, which is the Admin Portal, and uses the security

management functions to manage the SSO agent, authentication session, administrator, system operation settings, and security policies.

7.1. Security audit

SFR	TSF		TOE Functional Description
	Server	Agent	
FAU_ARP.1	FS.NOT.1	-	According to FAU_SAA.1, the SSO server (FS.NOT.1) satisfies the required function of FAU_ARP.1 by recording the fact as audit data and sending a security message by e-mail to the administrator or shutdown the service procedure when integrity check failure and self-test failure occurs.
FAU_GEN.1	FS.LOG.1	FA.AUG.1	FS.LOG.1 generates audit data for all administrative actions performed by administrators connected to the SSO server. When administrative actions, including logins, are performed, the results are stored in the DBMS. Audit data satisfies the security function of FAU_GEN.1 by recording the date and time of the audit target incident, task performing user ID and name, task performing content, performance result, and failure reason in [Table 6-3], and FA.AUG.1 is a function that generates audit data on the task performing history of the SSO agent, and transmits the result to the SSO server when performing a task including a login verification request. Audit data satisfies the security function of FAU_GEN.1, including the date and time, task execution user ID or subject identifier, task execution content, execution result, and failure reason for the audit target incident in [Table 6-4].

SFR	TSF		TOE Functional Description
	Server	Agent	
FAU_SAA.1	FW.AUG.1	-	FW.AUG.1 checks the integrity of FW.SVM.1 and FA.MGT.1 when recording management logs, and detects occurrences of self-test failure events. FS.AAA.1 meets the required functionality of FAU_SAA.1 by detecting potential security breaches by determining integrity or process failures and sending alert emails to authorized administrators.
FAU_SAR.1		-	Audit data generated through the audit record creation function (FS.AUG.1, FA.AUG.1) is stored in the local DBMS (PostgreSQL), and authorized administrators can access the Admin Portal and review the audit data stored in the DBMS according to the search conditions through the analyzable search function in the [Report] menu, thereby satisfying the required function of FAU_SAR.1.
FAU_SAR.3		-	Authorized administrators can review audit data based on search conditions (AND operation or click on the result values to sort them in ascending/descending order) through the audit data search function of the Admin Portal, thus satisfying the required function of FAU_SAR.3.
FAU_STG.1		-	The generated audit data is stored in the local DBMS (PostgreSQL).
FAU_STG.4	FS.NOT.1	-	FS.NOT.1 satisfies FAU_STG.4 as it notifies the defined administrator when the audit data storage space (the HDD area where TOE is installed) exceeds 80%.
FAU_STG.5		-	FS.NOT.1 satisfies FAU_STG.5 because it deletes the oldest 10% of audit records and sends them to the administrator by email

SFR	TSF		TOE Functional Description
	Server	Agent	
			when the audit data storage (the HDD area where TOE is installed) is full (90%).

7.2. Cryptographic support

SFR	TSF		TOE Functional Description
	Server	Agent	
FCS_CKM.1	FS.CRT.1	FA.CRT.1	The SSO server (FS.CRT.1) and SSO agent (FA.CRT.1) use the validated cryptographic module to generate encryption keys, thus satisfying FCS_CKM.1. The generated encryption keys and algorisms are shown in [Table 6-5].
FCS_CKM.2			The SSO server (FS.CRT.1) and SSO agent (FA.CRT.1) use the validated cryptographic module to generate encryption keys, it satisfies FCS_CKM.2 because it safely distributes the encryption key (authentication token key) using the RSAES-3072 (asymmetric key) method.
FCS_CKM.5			TOE satisfies FCS_CKM.5 by performing PBKDF2 (HMAC-SHA256) using input password to derive KEK that encrypts the encryption key(DEK) of TSF data.
FCS_CKM.6			TOE satisfies FCS_CKM.6 by destroying the used encryption key by overwriting the memory area where the encryption key is recorded with 0 three times (or calling the key initialization function provided by the validated cryptographic module three times) for the destruction time and destruction target specified in [Table 6-6] for the encryption key specified in [Table 6-6].
FCS_COP.1			FS.CRT.1 and FA.CRT.1 perform cryptographic operations and generate random bits using the following validated cryptographic modules.
FCS_RBG.1			
FCS_RBG.3			

<table>
<tr><th rowspan="2">SFR</th><th colspan="2">TSF</th><th rowspan="2">TOE Functional Description</th></tr>
<tr><th>Server</th><th>Agent</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Item</th><th>Details</th></tr>
<tr><td>Cryptographic Module Name</td><td>MPowerCrypto V3.0</td></tr>
<tr><td>Developer</td><td>UBIMINFO. Co., LTD.</td></tr>
<tr><td>Validation Date</td><td>2024-06-17</td></tr>
<tr><td>Validation Level</td><td>VSL1</td></tr>
<tr><td>Validation Number</td><td>CM-249-2029.6</td></tr>
<tr><td>Expiration Date</td><td>2029-06-17</td></tr>
</table>
TOE performs the cryptographic operations in [Table 6-7] using the above cryptographic module, and thus satisfies FCS_COP.1.
TOE satisfies FCS_RBG.1 by using the Entropy Source Name in [Table 6-8] when performing seeding initialized using the above cryptographic module.
TOE satisfies FCS_RBG.3 by seeding DRBG using Entropy Source Name from software-based entropy source [Table 6-8] with minimum entropy of at least 2^536 bits. TOE generates the entropy input required to perform the derivation function (HMAC-DRBG) defined in the TTAK.KO-12.0331-Part1/R1 standard, which ultimately results in a minimum entropy of at least 2^112 bits. After performing noise source soundness tests (RCT (repetition count test), APT (adaptability ratio test)) from four noise sources which are namely Java random number (64 bytes), system time (1 byte out of 8 bytes), process execution time difference (1 byte out of 8 bytes), and free memory (1 byte out of 8 bytes) the final entropy output (256 bytes) is reseed through a conditioning process (concatenating the 32-byte value generated by the Hash algorithm (LSH512-256) by repeating it 8 times and performing a concatenation operation).
</td></tr>
</table>

7.3. Identification and authentication

<table>
<tr><th rowspan="2">SFR</th><th colspan="2">TSF</th><th rowspan="2">TOE Functional Description</th></tr>
<tr><th>Server</th><th>Agent</th></tr>
<tr><td>FIA_AFL.1</td><td>FW.AAA.1
FS.AAA.1</td><td>-</td><td>The SSO server detects and creates a log of consecutive authentication failures within a range (1 to 5 times) that can be set by the administrator when identifying and authenticating a user attempting to connect, and if the number of consecutive authentication failures is exceeded, the user account is locked for 10 minutes and automatically unlocked, and the administrator is notified by email, thus satisfying FIA_AFL.1.</td></tr>
<tr><td>FIA_IMA.1</td><td>FS.COM.1</td><td>FA.COM.1</td><td>The SSO server (FS.COM.1) maintains a trusted connection session with the SSO agent (FA.COM.1) through mutual authentication according to the following procedure to meet the requirements of FIA_IMA.1.

- Request step from SSO Agent to SSO Server
<table>
<tr><th>Procedure</th><th>SSO Agent</th><th>SSO Server</th></tr>
<tr><td>generate signature</td><td>data_to_sign = {request_body + timestamp + nonce}

perform signature = hmac.new(SecretKey, data_to_sign, hashlib.sha256). hexdigest()</td><td>When receiving a request, recalculate the signature with the same data (request_body, timestamp, nonce) and verify that it matches the value in the header.</td></tr>
<tr><td>Information</td><td>Includes agent_id,</td><td>Verify that the IP</td></tr>
</table>
</td></tr>
</table>

SFR	TSF		TOE Functional Description
	Server	Agent	

and IP verification including request headers	timestamp, nonce, and signature	address of the registered agent with agent_id matches the actual request sending IP address.
encryption method	Use SSO Agent Secret Key, encrypt request body with ARIA/CBC/PKCS7Padding (256bit)	Decryption performed with the same key
Data validation	N/A	Signature validation, decryption success, and data tampering prevention
Prevent Replay	Generate a unique nonce and timestamp for each request	Requests with the same nonce or older than 2 minutes are rejected (nonce/timestamp pairs are kept for 2 minutes)

- Response step from SSO Server to SSO Agent

Procedure	SSO Server	SSO Agent
generate signature	data_to_sign = {response_body + timestamp + nonce} perform signature =	After recalculating the signature in the same way, verify that it matches the

SFR	TSF		TOE Functional Description		
	Server	Agent			
				hmac.new(SecretKey, data_to_sign, hashlib.sha256). hexdigest()	signature in the header.
			Information and IP verification including request headers	Includes timestamp, nonce, and signature	Verify that the response sending IP matches the IP of the registered SSO Server.
			encryption method	Use SSO Agent Secret Key, encrypt response body with ARIA/CBC/PKCS7Padding (256bit)	Decryption performed with the same key
			Data validation	N/A	Signature validation, decryption success, and data tampering prevention
			Prevent Replay	N/A	Requests with the same nonce or older than 2 minutes are rejected (nonce/timestamp pairs are kept for 2 minutes)
			Normal processing conditions	Response sent completed	If verification is successful, normal operation is performed (registration and

<table>
<tr><th rowspan="2">SFR</th><th colspan="2">TSF</th><th rowspan="2">TOE Functional Description</th></tr>
<tr><th>Server</th><th>Agent</th></tr>
<tr><td></td><td></td><td></td><td><table><tr><td></td><td></td><td>other processes are completed)</td></tr></table></td></tr>
<tr><td>FIA_SOS.1</td><td>FW.ADM.1</td><td>-</td><td>When creating or modifying an administrator password, it verifies compliance with the password creation rules in [Table 6-9] and displays them on the screen to ensure compliance, thus satisfying FIA_SOS.1.</td></tr>
<tr><td>FIA_SOS.2</td><td>-</td><td>FA.AAA.1</td><td>The SSO agent (FA.AAA.1) issues an authentication token based on the successful authentication result of a general user (when the SSO server successfully authenticates a user and issues a session_id, the agent generates a token based on the session) and delivers it to the user's terminal's browser. The business system verifies the validity of the authentication token when the user's terminal requests a service, thus satisfying FIA_SOS.2.
When generating an authentication token, the verification-filed encryption module uses MPowerCrypto V3.0, and the authentication token is configured as follows.<table><tr><td>token data structure</td><td>base64url(header) + "." + base64url(payload) + "." + base64url(signature)</td></tr><tr><td>encrypted token data structure</td><td>base64url(header) + "." + base64url(rsaEncrypt(ariaKey)) + "." + base64url(iv) + "." + base64url(ariaEncrypt(payload)) + "." + base64url(ariaEncrypt(signature))
Encrypted token length: 646~899 byte</td></tr><tr><td>token payload structure</td><td>JSON{iss(agent-id), sub(user-id), iat(issue timestamp), exp(experation timestamp), session_id(unique random value)}</td></tr></table></td></tr>
</table>

<table>
<tr><th rowspan="2">SFR</th><th>TSF</th><th rowspan="2">TOE Functional Description</th></tr>
<tr><th>Server</th><th>Agent</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td>agent-id : 64 byte or less
user-id: 64 byte or less
iat , exp: 10byte, or less
session_id: 24byte</td></tr>
<tr><td>Encryption method</td><td>Use of the ARIA-256-CBC algorithm provided by the validated cryptographic module</td></tr>
<tr><td>Encryption Key Encryption method</td><td>Use of the RSAES-3072(SHA-256) algorithm provided by the validated cryptographic module</td></tr>
</table>
</td></tr>
<tr><td>FIA_SOS.3</td><td>-</td><td>FA.AAA.1</td><td>The SSO agent (FA.AAA.1) satisfies FIA_SOS.3 by safely destroying the issued authentication token (repeatedly overwriting with 0 three times) after transmitting it to the user terminal's browser.
* Caution: Authentication tokens stored in the application of the user PC must be securely destroyed by the application of the PC.</td></tr>
<tr><td>FIA_UAU.2</td><td>FW.AAA.1
FS.AAA.1</td><td>-</td><td>Administrator login (FW.AAA.1, FS.AAA.1) is performed by entering the administrator ID and password and then entering the OTP value (see FIA_UAU.5 generation algorithm below) sent to the his/hers email address. At this time, a request is made to the DBMS where the administrator information is stored to verify that the ID and password match and that the administrator has the authority to authenticate. Since no other function can be performed other than entering the OTP value sent to the email address, it satisfies FIA_UAU.2.
This TOE follows the following procedures to ensure unpredictable and even distribution when generating one-time password(OTP) for Email</td></tr>
</table>

SFR	TSF		TOE Functional Description
	Server	Agent	
			transmission.
FIA_UAU.4	FW.AAA.1 FS.AAA.1 FS.SES.1	FA.AAA.1	The SSO server (FW.AAA.1, FS.AAA.1) generates a unique session ID with a time value for the access of an administrator who has successfully authenticated, thereby preventing the reuse of authentication data (authentication session). The SSO server (FS.SES.1) maintains a session ID generated based on the access authentication of a general user, and the SSO agent (FA.AAA.1) encrypts the authentication token and ensures uniqueness (timestamp, unique session ID based on random number, session expiration time setting) to prevent reuse of authentication data related to the authentication token, thereby satisfying FIA_UAU.4.
FIA_UAU.5	FW.AAA.1 FS.AAA.1	-	When performing administrator login authentication (FW.AAA.1, FS.AAA.1), additional authentication (OTP) is performed by sending an OTP to the administrator's email address after ID/PW authentication according to the authentication policy, thus satisfying FIA_UAU.5. This TOE follows the following procedures to ensure unpredictable and even distribution when generating one-time password(OTP) for Email transmission.

SFR	TSF		TOE Functional Description	
	Server	Agent		
			Process of the OTP generation	1. initialize the random number generator - OTP is generated using Java's standard cryptographic random number generator, SecureRandom. - SecureRandom is based on a random number source provided by the operating system, making it more unpredictable than general random number generators (Random) and providing cryptographic security that meets CC certification requirements. 2. Set OTP range (6 digits) - OTP has a range of 6 digits (000000 to 999999). - For this, TOE use the remainder of dividing the random value by 1,000,000. - If only simple modulo operations are used, the maximum value range may not be exactly divisible, which can cause distribution bias problems where some values appear more frequently. 3. Bias removal (rejection sampling method) - To avoid distribution bias, only the largest multiple (2,147,000,000) divisible by 1,000,000 (Integer.MAX_VALUE(2,147,483,647) is used as the random number range. - If a random value exceeds this range, it is discarded and a new random number is generated (rejection sampling method). - This ensures that all OTP candidate values (0 to 999,999) are generated with equal probability, eliminating distribution bias. 4. Remove negative numbers - SecureRandom.nextInt() can also return negative numbers, so if a negative number occurs, it is converted to an absolute value and processed. - Finally, only integers greater than or equal to 0 and less than 2,147,000,000 are used to generate OTP. 5. Final OTP output - The final OTP is generated by taking the remainder of the random number that passed the above process divided by 1,000,000. - This value is always an integer greater than or equal to 0 and less than or equal to 999,999. 6. Convert 6-digit fixed string - When converting the generated OTP integer to a string, use String.format("%06d", otp) to always ensure 6 digits. - Examples: 42 → "000042", 754 → "000754", 456789 → "456789"

SFR	TSF		TOE Functional Description
	Server	Agent	
			Security Basis: 1. Unpredictability: Use SecureRandom to obtain statistically and cryptographically secure random numbers. 2. Uniform distribution: Eliminate distribution bias by rejecting samples → OTP candidate values are generated with equal probability. 3. Fixed digits: Always expressed as 6 digits to avoid user confusion. 4. Prevent reuse: Issued OTPs are registered in server-side storage and immediately discarded after verification.
FIA_UAU.7	FW.AAA.1	-	When entering administrator login information, the password is processed with ● (dot masking) to prevent exposure, and the reason for failure is not provided in case of login failure, thereby satisfying FIA_UAU.7, a security function requirement for identification and authentication.
FIA_UID.2	FW.AAA.1 FS.AAA.1	-	The administrator login (FW.AAA.1, FS.AAA.1) function is performed by entering an ID value that uniquely identifies the administrator, and at this time, a request is made to the DBMS where the administrator information is stored to verify the ID. Since only the OTP information transmitted to the administrator is entered prior to login, it satisfies FIA_UID.2.

7.4. Security Management

SFR	TSF		TOE Functional Description
	Server	Agent	
FMT_MOF.1	FW.AGT.1 FS.AGT.1 FW.SVM.1 FW.ADM.1 FW.SES.1 FW.AUG.1	FA.MGT.1	Since TOE provides only authorized administrators with the ability to manage (add, modify, delete, search) agent registration and policy setting/enforcement (FW.AGT.1, FA.MGT.1/FS.AGT.1), server settings (FW.SVM.1), administrator account

SFR	TSF		TOE Functional Description
	Server	Agent	
			and permission settings (FW.ADM.1), authentication token sessions (FW.SES.1), and audit logs (FW.AUG.1), FMT_MOF.1 is satisfied. Detailed functions are as in '6.1.4.1. FMT_MOF.1 Management of security functions behaviour ' [Table 6-12].
FMT_MTD.1			This satisfies FMT_MTD.1 because it provides only authorized administrators with the ability to view and change TSF data such as agent registration and policy settings/enforcement (FW.AGT.1, FA.MGT.1/FS.AGT.1), server settings (FW.SVM.1), administrator account and permission settings (FW.ADM.1), authentication token sessions (FW.SES.1), and audit logs (FW.AUG.1). Detailed functions are as in '6.1.4.2. FMT_MTD.1 Management of TSF data ' [Table 6-13].
FMT_PWD.1	FW.ADM.1	-	When setting up an administrator account and password during the TOE installation process, the password security rules are managed to create it according to the password combination rules and length policy in [Table 6-9], thus satisfying the required function of FMT_PWD.1.
FMT_SMF.1		-	The SSO server administrator manages administrator account and permission policy information through the settings management menu, thus satisfying the FMT_SMF.1 requirement.
FMT_SMR.1		-	The server satisfies the requirements of FMT_SMR.1 by providing administrator role

SFR	TSF		TOE Functional Description
	Server	Agent	
			according to the administrator security policy.

7.5. TSF Protection

SFR	TSF		TOE Functional Description
	Server	Agent	
FPT_FLS.1	FS.CRT.1	FA.CRT.1	The cryptographic module of FS.CRT.1 and FA.CRT.1 maintains a secure state in case of self-test failure against a noise source, thus satisfying FPT_FLS.1. If the noise source intergrity test fails, it may be a temporary error, so the random number generator is kept in a safe state by retrying, but if the number of retries exceeds a certain number of times, it will transition to a critical error state. The list of failure types is as follows. *- Repetition Count Test(RCT) is higher than or equal to cutoff value(5)* *- Adaptive proportion Test(APT) is higher than or equal to cutoff value(9)*
FPT_ITT.1	FS.COM.1	FA.COM.1	The data transmitted and received through the SSO Server and SSO Agent's transmission and reception connection management function (FS.COM.1, FA.COM.1) is protected from exposure and modification threats using HTTPS, thus satisfying FPT_ITT.1. The communication encryption method uses TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 , TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 according to TLSv1.2 and

<table>
<tr><th rowspan="2">SFR</th><th colspan="2">TSF</th><th rowspan="2">TOE Functional Description</th></tr>
<tr><th>Server</th><th>Agent</th></tr>
<tr><td></td><td></td><td></td><td>uses TLS_AES_256_GCM_SHA384, TLS_AES_128_GCM_SHA256, TLS_CHACHA20_POLY1305_SHA256 according to TLSv1.3</td></tr>
<tr><td>FPT_PST.1</td><td>FS.CRT.1</td><td>-</td><td>SSO Server satisfies FPT_PST.1 by encrypting TSF data (administrator password, email server settings, agent encryption key, external authentication server settings) stored in the repository with DEK to protect them from unauthorized exposure.
<table>
<tr><th>TSF data</th><th>data protection method</th></tr>
<tr><td>administrator password</td><td>SHA-256 (salt:64bit, iter:1000)</td></tr>
<tr><td>SSO Agent secret key</td><td>ARIA-CBC PKCS7Pading</td></tr>
<tr><td>Alert mail address</td><td>ARIA-CBC PKCS7Pading</td></tr>
<tr><td>E-mail setting value</td><td>ARIA-CBC PKCS7Pading</td></tr>
<tr><td>Authentication server setting value</td><td>ARIA-CBC PKCS7Pading</td></tr>
<tr><td>DBMS connection value</td><td>ARIA-CBC PKCS7Pading</td></tr>
</table></td></tr>
<tr><td>FPT_TST.1</td><td>FW.SVM.1</td><td>FA.MGT.1</td><td>The server (V-FRONT v8 Server) performs self-tests upon startup and upon administrator request, and the agent (V-FRONT v8 Agent) performs self-tests and periodically upon startup, thereby satisfying FPT_TST.1. FW.SVM.1 provides an environment for administrators to request self-tests, and FA.MGT.1 provides an environment for setting agent test cycles.
The self-test items are as follows:
<table>
<tr><th>TOE</th><th>test interval</th><th>self-test items</th></tr>
<tr><td rowspan="2">Server</td><td rowspan="2">startup, administrator request</td><td>Self-test of the validated cryptographic module</td></tr>
<tr><td>process test</td></tr>
</table></td></tr>
</table>

<table>
<tr><th rowspan="2">SFR</th><th colspan="2">TSF</th><th rowspan="2">TOE Functional Description</th></tr>
<tr><th>Server</th><th>Agent</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>integrity test</td></tr>
<tr><td>Agent</td><td>startup, periodically (Agent Policy Setting(1~1440 min)</td><td>Self-test of the validated cryptographic module</td></tr>
<tr><td></td><td></td><td>process test</td></tr>
<tr><td></td><td></td><td>integrity test</td></tr>
</table>

The response to test failure is as in '[Table 6-2] Actions for security violations'.

The testing mechanism is as follows:
<table>
<tr><th>TOE</th><th>self-test items</th><th>test mechanism</th></tr>
<tr><td>Server</td><td>Self-test of the validated cryptographic module</td><td>Calling the crypto module self-test API</td></tr>
<tr><td></td><td>process test</td><td>Calling the mgmt_ui process test API
Calling the checkSum.jar process test API
Calling the mgmt_data process test API</td></tr>
<tr><td></td><td>integrity test</td><td>After generate checksum file, hashing all files contained in the /opt/aircuve/vfrontv8/ subdirectory and encrypt the file with DEK.

When it tests, this be compared with actual hash result those files the hash value</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th rowspan="2">SFR</th><th colspan="2">TSF</th><th rowspan="2">TOE Functional Description</th></tr>
<tr><th>Server</th><th>Agent</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Agent</td><td>Self-test of the validated cryptographic module</td><td>Calling the crypto module self-test API</td></tr>
<tr><td></td><td>process test</td><td>Calling the test API for Authentication token issuance and verification</td></tr>
<tr><td></td><td>integrity test</td><td>6.1.2.4After generate CHECKSUM file, hashing all files contained in the /opt/aircuve/vfrontv8/ subdirectory and encrypt the file with DEK.

When it tests, this be compared with actual hash result those files the hash value</td></tr>
</table>

The target to integrity check are as follows:
<table>
<tr><th>TOE</th><th>integrity check items</th></tr>
<tr><td>Server</td><td>All files contained in the /opt/aircuve/vfrontv8/ subdirectory</td></tr>
<tr><td>Agent</td><td>All files contained in the /opt/aircuve/vfrontv8/ subdirectory</td></tr>
</table>
</td></tr>
</table>

7.6. TOE Access

SFR	TSF		TOE Functional Description
	Server	Agent	
FTA_MCS.2	FW.AAA.1 FS.AAA.1 FS.SES.1	FA.AAA.1	The SSO server (FW.AAA.1, FS.AAA.1, FS.SES.1) satisfies the security function requirement FTA_MCS.2 by limiting the number of concurrent sessions to 1 by verifying whether a previous login session of the same authority administrator is active when a login request is made for a read/write administrator and terminating the previous session, according to the administrator's information settings. In addition, the number of concurrent sessions is limited to 1 by verifying whether a previous login session of the same user is active when a login request is made for a read administrator and terminating the previous session.
FTA_SSL.3	FW.AAA.1 FS.AAA.1 FS.SES.1	FA.AAA.1	The SSO server (FW.AAA.1, FS.AAA.1, FS.SES.1) verifies the administrator's inactivity period according to the administrator's information settings and terminates the administrator's management access session when the period exceeds the period, thus satisfying FTA_SSL.3. The SSO server maintains the final processing time of the data transmission/reception request from the administrator's PC after the administrator logs in as session information, and compares the final processing time with the current time. When the inactivity period, which can be set by the administrator (120 to 600 seconds), has elapsed, the interactive session is terminated.
FTA_TSE.1	FW.AAA.1 FS.AAA.1	-	The SSO server (FW.AAA.1, FS.AAA.1) verifies whether the IP address is authorized for access when an administrator requests access and work to be performed based on the management terminal IP settings, and satisfies FTA_TSE.1 because access failure occurs when accessing the TOE

SFR	TSF		TOE Functional Description
	Server	Agent	
			security management screen from an unregistered address.