KSignSecureDB V3.7
Security Target V1.5

KSign Co., Ltd

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

25, Gwacheon-daero 7na-gil, Gwacheon-si, Gyeonggi-do
TEL : 02-564-0182   FAX : 02-564-1627
http://www.ksign.com
KSign Co., Ltd.

Revision History

Version	Date	Content	Created by	Reviewed by
V1.0	2024.03.21	Initial version	Eunsil Jeong	Sanghak Sim
V1.1	2024.04.15	Unification of TOE Component names (PolicyServer -> Server)	Eunsil Jeong	Sanghak Sim
V1.2	2024.05.20	EOR-01 Update	Eunsil Jeong	Sanghak Sim
V1.3	2024.06.25	Assurance documents update	Eunsil Jeong	Sanghak Sim
V1.4	2024.07.01	Update TOE and 3rd Party version	Eunsil Jeong	Sanghak Sim
V1.5	2024.07.25	Reflect the opinion of the certification body	Eunsil Jeong	Sanghak Sim

Contents

1. ST Introduction

This document is the Security Target (ST) of KSignSecureDB V3.7('TOE') which targets the Common Criteria EAL1+ level.

1.1 ST reference

Title	KSignSecureDB V3.7 Security Target
ST Version	V1.5
Author	KSign Co., Ltd.
Publication Date	2024. 07. 25.
Evaluation Criteria	Common Criteria for Information Technology Security Evaluation
Common Criteria Version	CC V3.1 r5
Evaluation Assurance Level	EAL1+ (ATE_FUN.1)
Protection Profile	Korean National Protection Profile for Database Encryption V1.1
Keywords	Encryption, Decryption, DB, Database, DBMS, Oracle

1.2 TOE reference

Item			**Specification**
TOE			KSignSecureDB V3.7
TOE Version			V3.7.3
TOE Components	**KSignSecureDB Server**		KSignSecureDB Server V3.7.3
	KSignSecureDB DBAgent	KSignSecureDB DBAgent For Oracle_AIX	KSignSecureDB DBAgent For Oracle_AIX V3.7.3
		KSignSecureDB DBAgent For Tibero_AIX	KSignSecureDB DBAgent For Tibero_AIX V3.7.3
	KSignSecureDB APIAgent	KSignSecureDB APIAgent For JAVA_AIX	KSignSecureDB APIAgent For JAVA_AIX V3.7.3

Manual	**Preparation Procedure**	KSignSecureDB V3.7 Preparation Procedure	KSignSecureDB V3.7 Preparation Procedure V1.3
	Operation Guide	KSignSecureDB V3.7 Operation Guide	KSignSecureDB V3.7 Operation Guide V1.3
Developer			KSign Co., Ltd.

1.3 TOE overview

1.3.1 Database Encryption overview

KSignSecureDB V3.7(hereinafter referred to as 'TOE') performs the function of preventing the unauthorized disclosure of confidential information by encrypting the database (hereinafter referred to as 'DB').

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

1.3.2 TOE type and scope

The TOE is provided as software and shall provide the encryption/decryption function for the user data by each column. The TOE type defined in this Security Target can be grouped into the 'plug-in type' and 'API type', depending on the TOE operation type. The TOE supports both types. The TOE developed by the plug-in type is composed of the agent and management server, whereas the TOE developed by the API type is composed of the API module and management server.

The TOE consists of KSignSecureDB Server that performs the security management function of key management, access control policy management, cryptographic key management and administrator management; KSignSecureDB DBAgent as plug-in that installs a cryptographic module inside the user DB server and performs the encryption/decryption; KSignSecureDB APIAgent as API that interlinks with user applications and requests the encryption/decryption of the user data stored in the DB.

1.3.3 TOE usage and major security features

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

The DEK (Data Encryption Key) used to encrypt/decrypt the user data is protected by encryption with the KEK (Key Encryption Key)

Security Function	Main Function
User data protection	- The TOE provides the function of encrypting/decrypting the data stored in the DBMS under the protection by the unit of column by using KSignCASE64 v2.5.2.0, a validated cryptographic module, and generates different ciphertext values for the same plaintexts. - The TOE controls access to the DB to be protected in accordance with the following security policies established by the administrator. (key, encryption, decryption), System, User, IP, Time (period), Day, Date (period)
Cryptographic support	- The TOE offers the function of generation, distribution and destruction of cryptographic keys used for encryption and decryption through the validated cryptographic module. - The TOE performs cryptographic operations (data encryption and decryption) by using the generated cryptographic key.
Security audit	- Audit data are generated, including the date and time of the event, the type of the event, the identity of the subject that caused the event,

	task details and the outcome. - When the TOE generates audit data, it records the date and time of the event by receiving a reliable timestamp from the operating system where the Server has been installed. - The TOE provides the authorized administrator with the function to review the audit data. - The TOE sends a alert mail to the authorized administrator in case a potential security violation is detected. - The TOE sends a alert mail to the authorized administrator and performs the backup of the audit data in case of foreseen audit data loss. An audited event is ignored in case the audit trail is full.
Identification and authentication	- The TOE must perform the identification and authentication process based on the ID and password prior to any behavior of the administrator. The TOE enforces a designated combination rule when administrator ID and password are generated. All administrators can access the TOE through the management tools. If the authentication attempts are unsuccessful for a defined number of times, the TOE postpones the authentication of the administrator for a specified period of time.

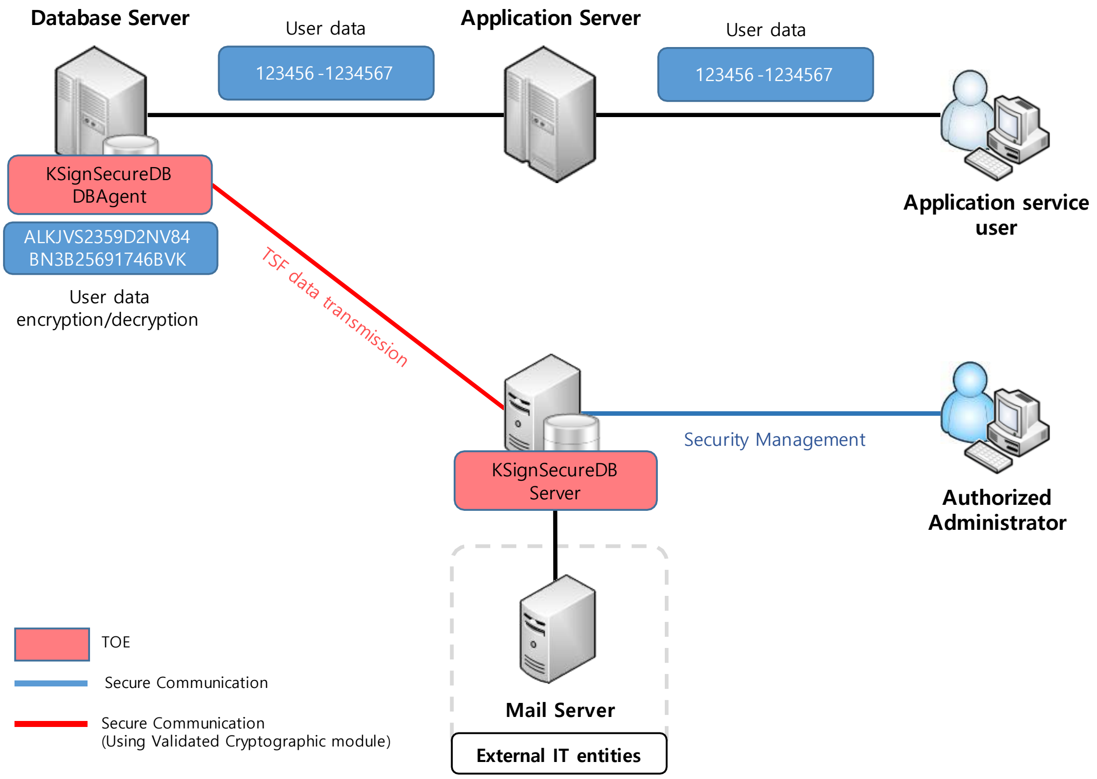
[Table 1-1] Main security properties

1.4 TOE operational environment

1.4.1 Non-TOE and TOE operational environment

The TOE operational environment can be classified into the plug-in type and the API type as follows:

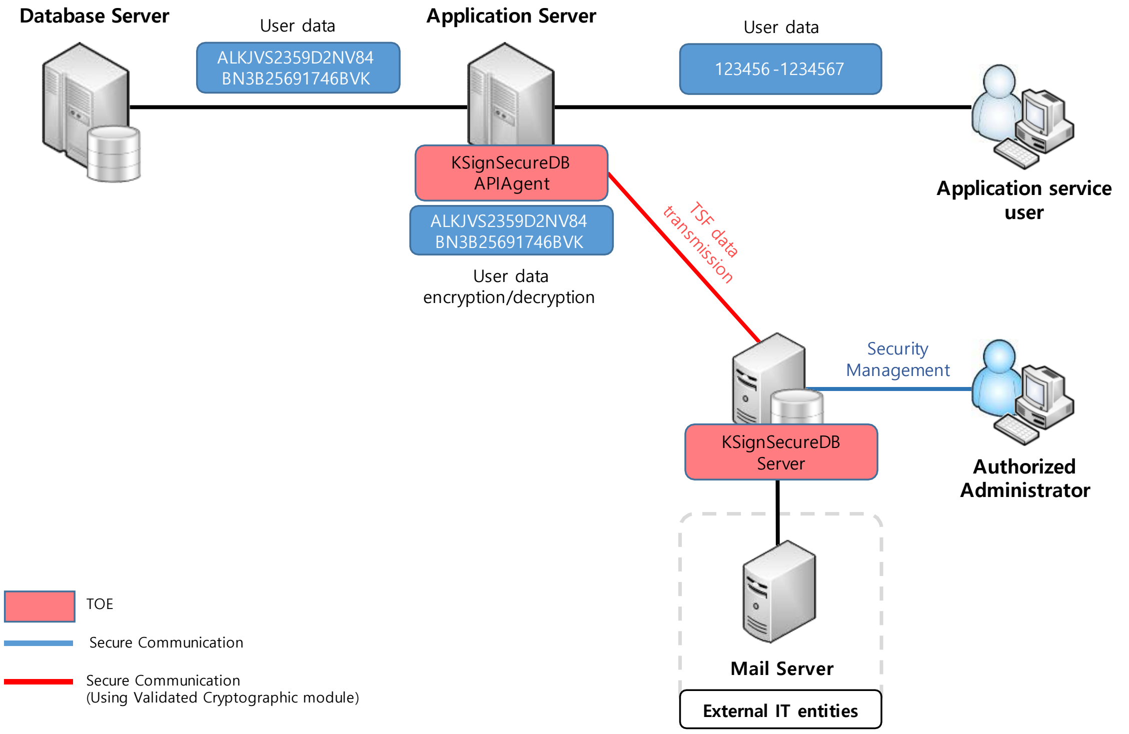
[Figure 1-1] shows an operational environment of the plug-in type. The plug-in operational environment is composed of KSignSecureDB Server and KSignSecureDB DBAgent. First, KSignSecureDB Server manages the information on policies established by the authorized administrator and manages the keys and the audit records. Second, KSignSecureDB DBAgent is installed inside the Database Server where the DB under the protection is located, and encrypts the user data receive from the Application Server before they are stored in the DB according to the policy configured by the authorized administrator. In addition, it decrypts the encrypted user data to be transmitted from the Database Server to the Application Server



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

The application service user requests the encryption or decryption of the user data through the Application Server in accordance with the scope of the encryption as required by the security policy. The requested data are encrypted by KSignSecureDB DBAgent and stored in the DB.

[Figure 1-2] shows the API type operational environment. The API type consists of KSignSecureDB APIAgent and KSignSecureDB Server. KSignSecureDB APIAgent is installed and operated in Application Server, and performs the encryption and decryption of the important data in accordance with the policy established by the administrator. The authorized administrator can access the KSignSecureDB Server and perform the security management.



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

The application service user performs the encryption and decryption of the user data through KSignSecureDB APIAgent on the Application Server in accordance with the scope of the encryption as required by the security policy. The encrypted user data is transmitted to the Database Server, and the encrypted user data transmitted from the Database Server is decrypted by APIAgent installed in the Application Server and transmitted to the application service user.

The communication among the TOE components shall be based on the encrypted communication using the approved cryptographic algorithm of the validated cryptographic module. In case the administrator accesses the Management Server through a web browser, a secure path (SSL/TLS V1.2) is generated to carry out the communication.

1.4.2 Requirements for non-TOE software and hardware

The TOE components consist of KSignSecureDB Server, KSignSecureDB DBAgent and KSignSecureDB APIAgent, which are distributed as software.

The minimum requirements and the operating system on which KSignSecureDB Server, KSignSecureDB DBAgent and KSignSecureDB APIAgent are installed and operated are as follows.

TOE	OS	Item	Specification
KSignSecureDB Server	AIX	OS	AIX 7.2 (7200-03) (64bit)
		CPU	PowerPC POWER5 2.1 GHz or higher
		Memory	8 GB or higher
		HDD	Space required for installation of TOE 3GB or higher
		NIC	100/1000 Mbps 1EA or higher
KSignSecureDB DBAgent KSignSecureDB APIAgent	AIX	OS	AIX 7.2 (7200-03) (64bit)
		CPU	PowerPC POWER5 2.1 GHz or higher
		Memory	4 GB or higher
		HDD	Space required for installation of TOE 1GB or higher
		NIC	100/1000 Mbps 1EA or higher

The minimum requirements for administrator systems for security management are as follows.

Item		Specification
S/W	Web Browser	Google Chrome 126.0

The following describes the DBMS information protected by KSignSecureDB DBAgent used in the TOE.

TOE	Protected DBMS
KSignSecureDB DBAgent For Oracle_AIX V3.7.3	Oracle 19c
KSignSecureDB DBAgent For Tibero_AIX V3.7.3	Tibero 7

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

TOE	S/W	Usage
KSignSecureDB Server	Java (JRE) 1.8.0_411	Server start-up and operation, security management and web server start-up based on Java Application
	Apache Tomcat 8.5.100	Encrypted communication between the web browser in the administrator system and the Server Web server to provide the security management screen
	Oracle 19c	DBMS for the TOE management
KSignSecureDB DBAgent/ KSignSecureDB APIAgent	Java (JRE) 1.8.0_411	TOE DBAgent and APIAgent start-up and operation based on Java Application

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

Item	Usage
Mail Server (SMTP Server)	Send the alert mail to an administrator

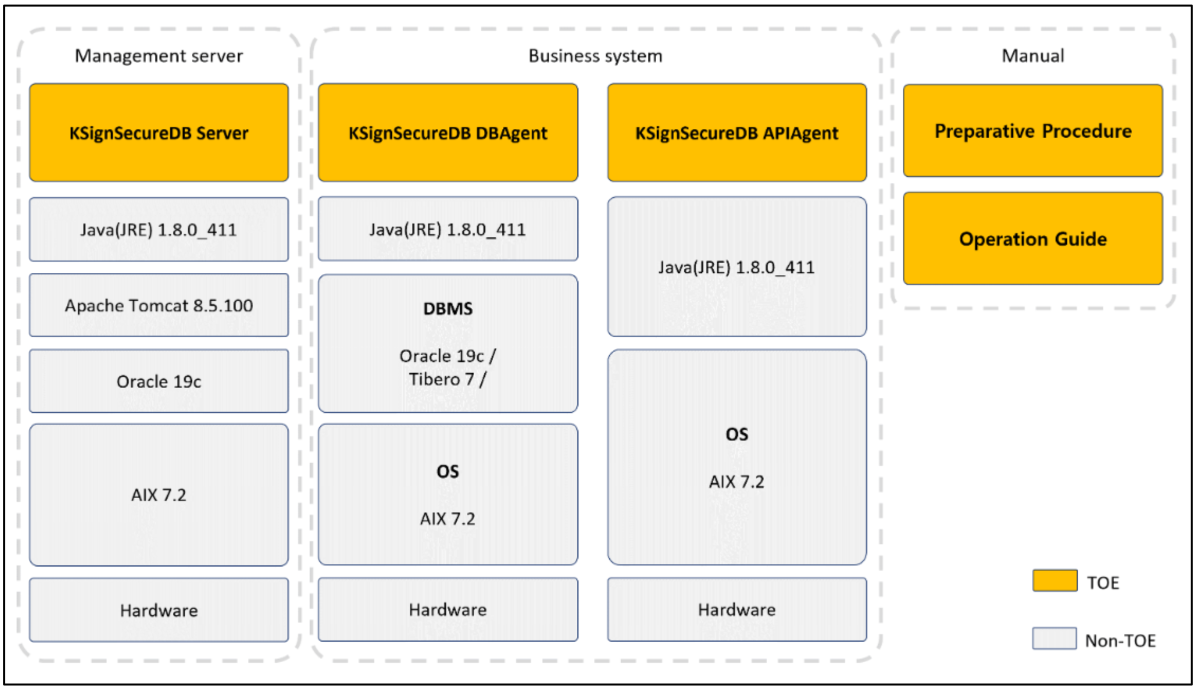
1.5 TOE description

This section describes the physical and logical scope and boundaries of the TOE.

1.5.1 Physical scope of the TOE

The TOE consists of Server, Agent, Preparation procedure and Operation Guide.

Scope	Identification		Type	Distribution Form
TOE	KSignSecureDB V3.7			
TOE Version	V3.7.3			
TOE Components	KSignSecureDB Server	KSignSecureDB Server V3.7.3 (KSDBV37-Server_V3.7.3.tar)	S/W	CD
	KSignSecureDB DBAgent	KSignSecureDB DBAgent For Oracle_AIX V3.7.3 (KSDBV37-DBAgent_For_Oracle_AIX_V3.7.3.tar)		
		KSignSecureDB DBAgent For Tibero_AIX V3.7.3 (KSDBV37-DBAgent_For_Tibero_AIX_V3.7.3.tar		
	KSignSecureDB APIAgent	KSignSecureDB APIAgent For JAVA_AIX V3.7.3 (KSDBV37-APIAgent_For_API_JAVA_AIX_V3.7.3.tar)		
Manual	Preparative procedure	KSignSecureDB V3.7 Preparative procedure V1.3 (KSignSecureDB V3.7 Preparative procedure V1.3.pdf)	File (PDF)	
	Operation Guide	KSignSecureDB V3.7 Operation Guide V1.3 (KSignSecureDB V3.7 Operation Guide V1.3.pdf)		



The TOE uses the following validated cryptographic module.

TOE	S/W	Usage
KSignSecureDB Server	KSignCASE64 v2.5.2.0	Validated cryptographic module for key generation, destruction and distribution, and cryptographic operations. Validated cryptographic module for encrypted communication between TOE components.
KSignSecureDB DBAgent/ KSignSecureDB APIAgent	KSignCASE64 v2.5.2.0	Validated cryptographic module for key generation, destruction and distribution, and cryptographic operations. Validated cryptographic module for encrypted communication between TOE components.

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

Item	Specification

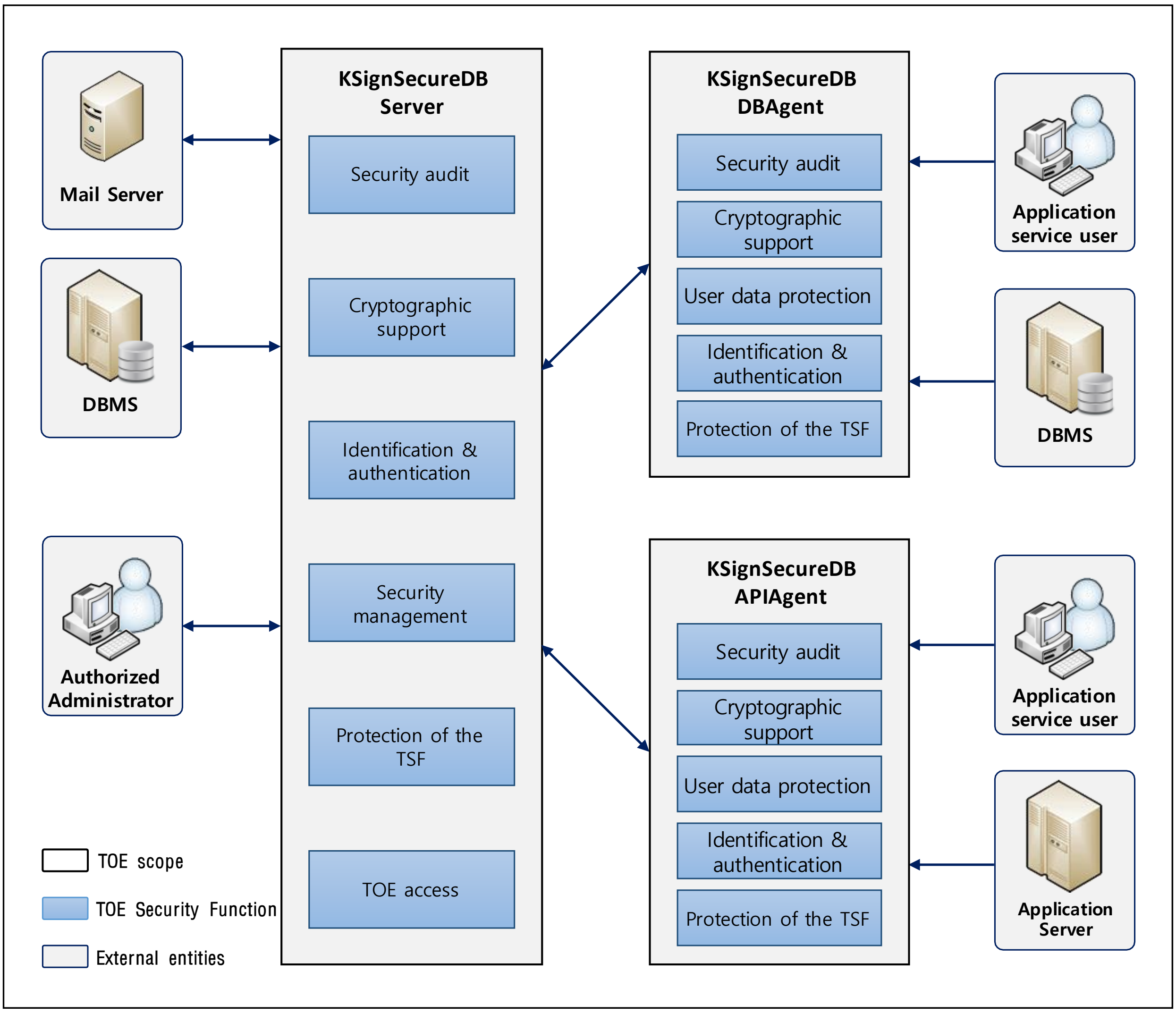
Cryptographic module name	KSignCASE64 v2.5.2.0
Developer	KSign Co., Ltd
Validation date	2023. 10. 16.
Validation level	VSL1
Validation number	CM-237-2028.10
Expiration Date	2028. 10. 16.

The 3rd Party libraries used in TOE is as follows.

TOE	Library	Usage
KSignSecureDB Server	ojdbc8.jar	Library for KSignSecureDB DBAgent For Oracle_AIX to access the protected DBMS
	tibero7-jdbc.jar	Library for KSignSecureDB DBAgent For Tibero_AIX to access the protected DBMS
KSignSecureDB Server	log4j-core-2.23.1.jar log4j-api-2.23.1.jar	Library used for generating temporary audit log files
KSignSecureDB DBAgent/ KSignSecureDB APIAgent	log4j-core-2.23.1.jar log4j-api-2.23.1.jar	Library used for generating temporary audit log files

1.5.2 Logical scope of the TOE

The logical scope of the plug-in type and the API type according to the TOE operation method is as follows.



- Security audit

- KSignSecureDB Server provides a means that enables only the authorized administrator to view the audit information and provides the audit information in an understandable form. If an auditable event occurs, it generates the audit data, detects a potential security violation and sends an alert mail to the authorized administrator. Furthermore, it provides the function of storing all audit data generated by KSignSecureDB Server, KSignSecureDB DBAgent and KSignSecureDB APIAgent in the audit trail storage(DBMS) to manage them securely; preventing the audit data from unauthorized deletion; If the audit trail exceeds 80% of the audit storage capacity, an alert mail is sent to the authorized administrator, and when the audit trail is full, it provides a function to protect the audit trail storage by ignoring audited events.

■ Cryptographic support

- KSignSecureDB Server, KSignSecureDB DBAgent and KSignSecureDB APIAgent generate and destroy all cryptographic keys used for the operation of the product in a secure manner through the validated cryptographic module KSignCASE64 v2.5.2.0 whose safety and suitability for the implementation have been confirmed by the cryptographic module validation scheme, and performs cryptographic operation in accordance with the user data cryptographic policy that defines the cryptographic algorithm, and performs cryptographic operation for confidentiality and integrity of the TSF data. It destroys cryptographic keys from memory after encryption and decryption. In addition, it generates and distribute cryptographic keys using the validated cryptographic module KSignCASE64 v2.5.2.0 for secure communication between KSignSecureDB Server and KSignSecureDB DBAgent/KSignSecureDB APIAgent that are physically separated.

- Cryptographic key generation:
  - HASH_DRBG (SHA256, 256bit): Cryptographic key generation for the encryption/decryption of the TSF data, the encryption/decryption of the user data and the encryption/decryption of the cryptographic key (policy key)
  - RSAES (2048bit): Asymmetric key generation for the encryption/decryption of the master key and KSign-implemented secure communication
  - PBKDF(HMAC(SHA256)): Key generation for the protection of the TSF data encryption key
- Cryptographic key distribution

  - RSAES (2048bit): Encryption/Decryption of the session key to transmit the data between KSignSecureDB Server, KSignSecureDB DBAgent and KSignSecureDB APIAgent in case of the KSign-implemented secure communication

- Cryptographic operation
  - Encryption/Decryption of symmetric key (SEED-CBC, 128bit): Encryption/Decryption of the TSF data and the user data
  - Encryption/Decryption of symmetric key (ARIA-CBC, 128bit/192bit/256bit): Encryption/Decryption of the user data
  - One-way encryption (SHA256): Encryption of the user data, encryption of the TSF data and integrity verification
  - One-way encryption (SHA512): Encryption of the user data
  - Encryption/Decryption of asymmetric key (RSAES, 2048bit): Encryption/Decryption of the master key

- Cryptographic key destruction
  - The cryptographic key information in the memory is deleted after the update with 0x00 if KSignSecureDB DBAgent and KSignSecureDB APIAgent are shut down.
  - The cryptographic key information is deleted by updating the temporarily stored cryptographic key information with 0x00 after sending the cryptographic key from KSignSecureDB Server to KSignSecureDB APIAgent and KSignSecureDB DBAgent.

■ User data protection

- KSignSecureDB DBAgent and KSignSecureDB APIAgent provide the function of encrypting/decrypting the data stored in the DBMS under the protection by the unit of column by using KSignKACE64 v2.5.2.0, a validated cryptographic module, and generates different ciphertext values for the same plaintexts. When performing encryption, the original data is deleted, and when decryption is performed, the encrypted data is deleted, so that the previous information content is not available.

■ Identification and authentication

- KSignSecureDB Server provides the function of performing the identification and authentication of the administrator who intends to use the security management function before the administrator initiates any behavior, masking with (●) when the authentication data are entered and protecting the authentication feedback. Furthermore. it provides the secure identification and authentication function by inactivating the authentication for 5 minutes and sending alert mail to administrator in case of 5 continuous failures in authentication attempts. It also prevents the reuse of authentication data of the administrator who logs on to KSignSecureDB Server.
- KSignSecureDB Server provides the mechanisms to verify that the user password verification meets the following defined metrics.
  - Password length: 10 characters minimum, 30 characters maximum
  - Available characters for password: English alphabets, numbers, special characters (!, @, #, $, %, ^, *)
  - A password must have a combination of three or more among English alphabets, numbers and special characters.
- KSignSecureDB Server and KSignSecureDB DBAgent/KSignSecureDB APIAgent perform the mutual authentication through a KSign-implemented.

■ Security management

- KSignSecureDB Server provides the security management function including the access control policy management, the administrator management and KSignSecureDB Server configuration for the authorized administrator. The authorized administrator carries out the management function through the security management interface.
- The authorized administrator includes supervisor, policy administrator, system administrator, encryption administrator and audit record review administrator. The roles of the authorized administrator provided by KSignSecureDBServer are as follows.
- Supervisor: The top administrator has the privilege of system management, policy management, establishing and performing the table encryption/decryption and viewing audit record, and can create lower-level administrators other than supervisor.
- System administrator: The system administrator has the privilege of system management menu, generation, deletion and modification of the administrator and system configuration.

- Policy administrator: The policy administrator has the privileges of target DBMS management and key(policy) registration.
- Encryption administrator: The encryption administrator has the privilege of establishing and performing the table encryption.
- Audit record review administrator: The audit record review administrator has the privilege of reviewing the audit records.

- It is enforced that the authorized administrator changes the password when the authorized administrator accesses to the security management interface for the first time.

■ Protection of the TSF

- KSignSecureDB Server ensures the confidentiality and the integrity of the TSF data transmitted from/to KSignSecureDB DBAgent and KSignSecureDB APIAgent that are physically separated, through the encryption communication. KSignSecureDB Server runs a suite of self-tests to check the process status during initial start-up and periodically during normal operation to demonstrate that it remains in the safe condition and its security functions are in correct operation. It also verify the integrity of the TSF data and TSF executable codes, which are subject to the integrity verification during initial start-up, periodically during normal operation and when the authorized administrator request.
- KSignSecureDB DBAgent and KSignSecureDB APIAgent loads TSF data for the encryption communication and mutual authentication upon the start-up. After the mutual authentication succeeds, the integrity information is sent to KSignSecureDB Server to verify the integrity against the integrity information registered inside the Server. Integrity verification is performed not only during initial start-up but also on request by authorized administrators. Furthermore, Self-tests is performed to check the process status during initial start-up and periodically during normal operation to ensure that it remains in the safe condition and its security functions are in correct operation.
- KSignSecureDB Server, KSignSecureDB DBAgent and KSignSecureDB APIAgent manage the administrator authentication information, integrity verification information, KSignSecureDB Server and KSignSecureDB DBAgent/KSignSecureDB APIAgent information and so forth by storing them in the DBMS in secure manner or by storing with encryption to protect the TSF data.

■ TOE access

- In case of the management access session by administrator allowed to access perform the security management functions for KSignSecureDB Server, the maximum number of concurrent sessions is limited to one.

- If the supervisor is log-in, a lower-level administrator is not allowed to access. If the supervisor accesses while a lower-level administrator is log-in, the access by a lower-level administrator is cancelled. Furthermore, if an access attempt is made with the account which is the same as the supervisor account, the preceding access is cancelled. In case of log-in with the account or the privilege which is the same as that of a lower-level administrator, the preceding access is cancelled. In addition, the administrator session is terminated after a specified time interval of inactivity (10 minutes).

- In this case, a lower-level administrator refers to the system administrator, the policy administrator, the encryption administrator and the audit record review administrator, except for the supervisor.

- In case of all administrators, access sessions are restricted in accordance with the accessible IP rules, and the management access sessions are allowed only on the terminals (two or less) that have IPs designated as accessible. Audit data are generated regarding the execution result of the session restriction in the security management interface.

1.6 Term and definitions

Terms used in Security Target, which are the same as in the CC, must follow those in the CC.

Session Key

Key generated from the validated cryptographic module and used during secure communication between KSignSecureDB Server and KSignSecureDB DBAgent or APIAgent

Master Key

Key generated from the validated cryptographic module. It is generated on KSignSecureDB Server upon the initial start-up of the product. The generated Master Key is encrypted with the public key, and then stored in the DBMS so that it is managed securely.

Policy Key

Key generated from the validated cryptographic module. It is generated by the authorized administrator in the security management interface to be used for the encryption and decryption of the user data.

Object

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

Attack potential

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

Iteration

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

Security Target(ST)

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

Protection Profile(PP)

Implementation-independent statement of security needs for a TOE type

User

Refer to "External entity", but in the TOE, the user is the authorized administrator and the authorized general user

Selection

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

Identity

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

Element

Indivisible statement of a security need

Role

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

Operation(On a component of the CC)

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

Operation(on a subject)

Specific type of action performed by a subject on an object

External Entity

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

Treat Agent

Entity that can adversely act on assets

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

Authentication Data

Information used to verify the claimed identity of a user

Assets

Entities that the owner of the TOE presumably places value upon

Refinement

Addition of details to a component

Organizational Security Policies

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

Dependency

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

Subject

Active entity in the TOE that performs operations on objects

Augmentation

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

Component

Smallest selectable set of elements on which requirements may be based

Class

Set of CC families that share a common focus

Target of Evaluation(TOE)

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

Evaluation Assurance Level(EAL)

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

Family

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

Assignment

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

TOE Security Functionality(TSF)

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

TSF Data

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

Management access

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

Local access

The direct access through console port by administrator for TOE management

SSL(Secure Sockets Layer)

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

TLS(Transport Layer Security)

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

Shall/must

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

Can/could

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

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

1.7 Conventions

The notation, formatting and conventions used in this Security Target 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 Security Target.

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.**

Security Target (ST) Author

This is used to represent the final decision of attributes being made by the ST author. The ST author's operation is denoted in braces, as in {decided by the ST author}. In addition, operations of SFR not completed in the Protection Profile must be completed by the ST author. “Application notes” is provided to clarify the intent of requirements, provide the information for the optional items in implementation, and define "Pass/Fail" criteria for a requirement. The application notes is provided with corresponding requirements if necessary.

Application notes

This Security Target provides “Application Notes” to clarify the meaning of requirements and provides the information on options to be applied in the process of the implementation. It also defines the “pass/fail” criteria for the requirements. Application notes are provided together with relevant requirements if deemed necessary.

2. Conformance claim

2.1 CC conformance claim

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

2.2 PP conformance claim

The Protection Profile to which this Security Target complies is ‘Korean National Protection Profile for Database Encryption V1.1’

2.3 Package conformance claim

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

2.4 Conformance claim rationale

Since this ST adopts the TOE type, security objectives and security requirements in the same way as the Protection Profile, and the compliance claim of 'National Database Encryption Protection Profile V1.1' is 'strict PP conformance'.

SFRs to which an iteration operation is applied, among SFRs in the "Korean National PP for Database Encryption V1.1"
: FCS_CKM.1, FCS_COP.1

3. Security objectives

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

3.1 Security objectives for the operational environment

OE.PHYSICAL_CONTROL

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

OE.TRUSTED_ADMIN

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

OE.SECURE_DEVELOPMENT

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

OE.LOG_BACKUP

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

OE.OPERATION_SYSTEM_RE-INFORCEMENT

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

OE.TRUSTED_TIMESTAMP

The TOE shall accurately record security-relevant events by using trusted time stamps provided by the TOE operational environment.

OE.SECURE_DBMS

Audit records stored in the audit trail such as the DBMS that interlinks with the TOE shall be protected from unauthorized deletion or modification.

OE.SECURED_ADMIN_ACCESS

The web server, which is the operating environment of the management server, and the web browser of the administrator PC must communicate using a secure path.

4. Extended components definition

4.1 Cryptographic Support

4.1.1 Random Bit generation

Family Behaviour

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

Component leveling

FCS_RBG Random bit generation
1

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

Management FCS_RBG.1

There are no management activities foreseen.

Audit FCS_RBG.1

There are no auditable events foreseen.

4.1.1.1 FCS_RBG.1 Random bit generation

Hierarchical to No other components.

Dependencies No dependencies.

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

4.2 Identification & authentication)

4.2.1 TOE Internal mutual authentication

Family Behaviour

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

Component leveling

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

Management: FIA_IMA.1

There are no management activities foreseen.

Audit: FIA_IMA.1

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

a) Minimal: Success and failure of mutual authentication

b) Minimal: Modification of authentication protocol

4.2.1.1 FIA_IMA.1 TOE Internal mutual authentication

Hierarchical to No other components.

Dependencies No Dependencies.

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

4.3 User data protection

4.3.1 User data encryption

Family Behaviour

This family provides requirements to ensure confidentiality of user data.

Component leveling

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

Management: FDP_UDE.1

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

a) Management of user data encryption/decryption rules

감사: FDP_UDE.1

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

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

4.3.1.1 FDP_UDE.1 User data encryption

Hierarchical to No other components.

Dependencies FCS_COP.1 Cryptographic operation

FDP_UDE.1.1 TSF shall provide TOE users with the ability to encrypt/decrypt user data according to [assignment: the list of encryption/decryption methods] specified.

4.4 Security Management

4.4.1 ID and password

Family Behaviour

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

Component leveling

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

Management: FMT_PWD.1

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

a) Management of ID and password configuration rules.

Audit: FMT_PWD.1

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

a) Minimal: All changes of the password.

4.4.1.1 FMT_PWD.1 Management of ID and password

Hierarchical to	No other components.
Dependencies	FMT_SMF.1 Specification of management functions FMT_SMR.1 Security roles

FMT_PWD.1.1 The TSF shall restrict the ability to manage the password of [assignment: list of functions] to [assignment: the authorized identified roles].

1. [assignment: password combination rules and/or length
2. [assignment: other management such as management of special characters unusable for password, etc.]

FMT_PWD.1.2 The TSF shall restrict the ability to manage the ID of [assignment: list of functions] to [assignment: the authorized identified roles].

1. [assignment: ID combination rules and/or length]
2. [assignment: other management such as management of special characters unusable for ID, etc.]

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

4.5 Protection of the TSF

4.5.1 Protection of stored TSF data

Family Behaviour

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

Component leveling

FPT_PST Protection of stored TSF data
1

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

Management: FPT_PST.1

There are no management activities foreseen

Audit: FPT_PST.1

There are no auditable events foreseen.

4.5.1.1 FPT_PST.1 Basic protection of stored TSF data

Hierarchical to No other components.

Dependencies No dependencies.

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

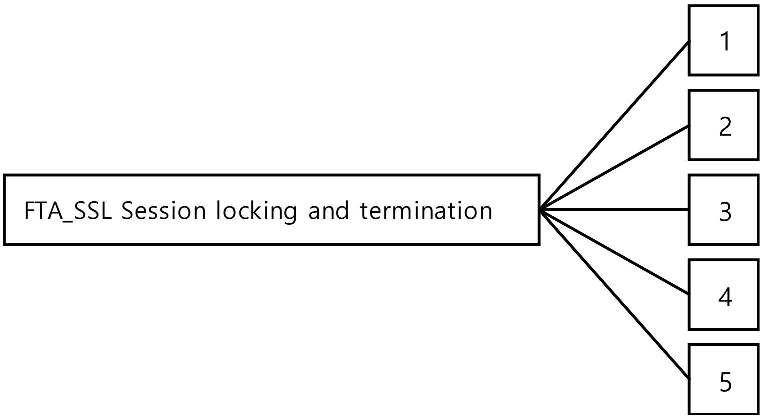
4.6 TOE Access

4.6.1 Session locking and termination

Family Behaviour

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

Component leveling



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

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

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

Management: FTA_SSL.5

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

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

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

Audit: FTA_SSL.5

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

a) Minimal: Locking or termination of interactive session

4.6.1.1 FTA_SSL.5 Management of TSF-initiated sessions

Hierarchical to No other components.

Dependencies [FIA_UAU.1 authentication or No dependencies.]

FTA_SSL.5.1 The TSF shall [selection:

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

5. Security requirements

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

5.1 Security functional requirements

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

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

Security functional class	Security functional component	
FAU	FAU_ARP.1	Security alarms
	FAU_GEN.1	Audit data generation
	FAU_SAA.1	Potential violation analysis
	FAU_SAR.1	Audit review
	FAU_SAR.3	Selectable audit review
	FAU_STG.3	Action in case of possible audit data loss
	FAU_STG.4	Prevention of audit data loss
FCS	FCS_CKM.1	Cryptographic key generation
	FCS_CKM.2	Cryptographic key distribution
	FCS_CKM.4	Cryptographic key destruction
	FCS_COP.1	Cryptographic operation
	FCS_RBG.1(Extended)	Random bit generation
FDP	FDP_UDE.1(Extended)	User data encryption

	FDP_RIP.1	Subset residual information protection
FIA	FIA_AFL.1	Authentication failure handling
	FIA_IMA.1(Extended)	TOE Internal mutual authentication
	FIA_SOS.1	Verification of secrets
	FIA_UAU.2	User authentication before any action
	FIA_UAU.4	Single-use authentication mechanisms
	FIA_UAU.7	Protected authentication feedback
	FIA_UID.2	User identification before any action
FMT	FMT_MOF.1	Management of security functions behaviour
	FMT_MTD.1	Management of TSF data
	FMT_PWD.1(Extended)	Management of ID and password
	FMT_SMF.1	Specification of management functions
	FMT_SMR.1	Security roles
FPT	FPT_ITT.1	Basic internal TSF data transfer protection
	FPT_PST.1(Extended)	Basic protection of stored TSF data
	FPT_TST.1	TSF testing
FTA	FTA_MCS.2	Per user attribute limitation on multiple concurrent sessions
	FTA_SSL.5(Extended)	Management of TSF-initiated sessions
	FTA_TSE.1	TOE session establishment

[Table 5-1] Security functional requirements

5.1.1 Security audit(FAU)

5.1.1.1 FAU_ARP.1 Security alarms

Hierarchical to No other components.

Dependencies FAU_SAA.1 Potential violation analysis

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

5.1.1.2 FAU_GEN.1 Audit data generation

Hierarchical to No other components.

Dependencies FPT_STM.1 Reliable time stamps

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) All auditable events for the *not specified* level of audit; and

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

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

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

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

Security functional	Auditable event	Additional audit record

component		
FAU_ARP.1	Actions taken due to potential security violations	
FAU_SAA.1	Enabling and disabling of any of the analysis mechanisms, Automated responses performed by the tool	
FAU_STG.3	Actions taken due to exceeding of a threshold	
FAU_STG.4	Actions taken due to the audit storage failure	
FCS_CKM.1(1)	Success and failure of the activity	
FCS_CKM.2	Success and failure of the activity (only applying to distribution of key related to user data encryption/decryption)	
FCS_CKM.4	Success and failure of the activity (only applying to destruction of key related to user data encryption/decryption)	
FCS_COP.1(1)	Success and failure of the activity	
FDP_UDE.1	Success and failure of user data encryption/decryption	
FIA_AFL.1	The reaching of the threshold for the unsuccessful authentication attempts and the actions taken, and the subsequent, if appropriate, restoration to the normal state	
FIA_IMA.1	Success and failure of mutual authentication Modify of authentication protocol	
FIA_UAU.2	All use of the authentication mechanism	
FIA_UAU.4	Attempts to reuse authentication data	
FIA_UID.2	All use of the administrator identification mechanism, including the administrator identity provided	
FMT_MOF.1	All modifications in the behaviour of the functions in the TSF	

FMT_MTD.1	All modifications to the values of TSF data	Modified values of TSF data
FMT_PWD.1	All changes of the password	
FMT_SMF.1	Use of the management functions	
FMT_SMR.1	Modifications to the user group of rules divided	
FPT_TST.1	Execution of the TSF self-tests and the results of the tests	Modified TSF data or execution code in case of integrity violation
FTA_MCS.2	Denial of a new session based on the limitation of multiple concurrent sessions	
FTA_SSL.5	Locking or termination of interactive session	
FTA_TSE.1	Denial of a session establishment due to the session establishment mechanism All attempts at establishment of a user session	

[Table 5-2] Audit event

5.1.1.3 FAU_SAA.1 Potential violation analysis

Hierarchical to No other components.

Dependencies FAU_GEN.1 Audit data generation

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

FAU_SAA.1.2 The TSF shall enforce the following rules for monitoring audited events:

a) Accumulation or combination of [

authentication failure audit event among auditable events of FIA_UAU.1, integrity violation audit event and self-test failure event of validated cryptographic module among auditable events of FPT_TST.1,

[Audit Trail Storage Exceeded Threshold and Saturation Event,

License verification failure event]

] known to indicate a potential security violation

b) [None]

5.1.1.4 FAU_SAR.1 Audit review

Hierarchical to No other components.

Dependencies FAU_GEN.1 Audit data generation

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

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

5.1.1.5 FAU_SAR.3 Selectable audit review

Hierarchical to No other components.

Dependencies FAU_SAR.1 Audit review

FAU_SAR.3.1 The TSF shall provide the capability to apply [View Period, Task Target (Agent, Table Owner), Job classification, Job Manager, Job Result and IP in And operation] of audit data based on [Sort by date and time in descending order].

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

Hierarchical to No other components.

Dependencies FAU_STG.1 Protected audit trail storage

FAU_STG.3.1 The TSF shall [Notification to the authorized administrator, [None]] if the audit trail exceeds [Threshold Exceeded Default Tablespace Size 80%].

5.1.1.7 FAU_STG.4 Prevention of audit data loss

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

Dependencies FAU_STG.1 Protected audit trail storage

FAU_STG.4.1 The TSF shall *ignore audited events* and [send alert mail to the authorized administrator] if the audit trail is full.

5.1.2 Cryptographic support (FCS)

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

Hierarchical to No other components.

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

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [cryptographic key generation algorithm for Random bit generation in [Table 5-3]] and specified cryptographic key sizes [cryptographic key sizes for Symmetric key encryption in [Table5-3]] that meet the following: [list of standards for Random bit generation in [Table 5-3]].

List of standards	Cryptographic operation	Cryptographic algorithm	Cryptographic key sizes	usage
KS X 1213-1	Symmetric key encryption	ARIA (CBC)	128	User data encryption/decryption
			192	
			256	
TTAS.KO-12.0004/R1	Symmetric key encryption	SEED (CBC)	128	User data encryption/decryption
ISO/IEC 10118-3	One-way encryption	SHA256	N/A	User data encryption
		SHA512	N/A	
KS X ISO/IEC 18031	Random bit generation	HASH-DRBG-SHA256	N/A	encryption key generation

[Table 5-3] User data encryption algorithm and key size

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

Hierarchical to No other components.

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

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [cryptographic key generation algorithm for Random bit generation, Public key encryption, Key derivation based on password in [Table 5-4]] and specified cryptographic key sizes [cryptographic key sizes for Symmetric key encryption, Key derivation based on password, Public key encryption in [Table 5-4]] that meet the following: [list of standards for Random bit generation, Public key encryption, Key derivation based on password in [Table 5-4]].

List of standards	Cryptographic operation	Cryptographic algorithm	Cryptographic key sizes	TOE module	usage
TTAS.KO-12.0004/R1	Symmetric key encryption	SEED (CBC)	128	KSignSecureDB Server	Encryption for user data encryption key protection (policy key encryption) - master key
TTAS.KO-12.0004/R1	Symmetric key encryption	SEED (CBC)	128	KSignSecureDB Server, KSignSecureDB DBAgent, KSignSecureDB APIAgent	TSF data encryption (used for data encryption transfer between TOE components) - Session key
TTAS.KO-12.0004/R1	Symmetric key encryption	SEED (CBC)	128	KSignSecureDB Server	TSF data encryption (encryption of important information stored in management DB or configuration file) - DEK
TTAS.KO-12.0004/R1	Symmetric key encryption	SEED (CBC)	128	KSignSecureDB Server	Encryption for TSF data encryption key protection (TSF DEK

					encryption and decryption) - KEK
ISO/IEC 10118-3	One-way encryption	SHA256	N/A	KSignSecureDB Server	Encrypt the administrator password
ISO/IEC 10118-3	One-way encryption	SHA256	N/A	KSignSecureDB Server, KSignSecureDB DBAgent, KSignSecureDB APIAgent	TOE component integrity verification Mutual authentication between TOE components Verify Cryptographic Key Integrity data integrity verification for transferring data between TOE components
TTAK.KO-12.0334-Part2	Key derivation based on password	PBKDF2(HMAC-SHA256)	128	KSignSecureDB Server, KSignSecureDB DBAgent, KSignSecureDB APIAgent	Cryptographic key generation for TSF data encryption key protection (KEK generation)
KS X ISO/IEC 18033-2	Public key encryption	RSAES	2048	KSignSecureDB Server	Encryption/Decryption for master key protection
KS X ISO/IEC 18033-2	Public key encryption	RSAES	2048	communication between TOE components	Session key encryption and for data transmission during key distribution between TOE components

KS X ISO/IEC 18031	Random bit generation	HASH-DRBG-SHA256	128	KSignSecureDB Server, KSignSecureDB DBAgent, KSignSecureDB APIAgent	Generate master key, TSF DEK, session key Generate salt for deriving TSF KEK

[Table 5-4] TSF data encryption algorithm and key size

5.1.2.3 FCS_CKM.2 Cryptographic key distribution

Hierarchical to No other components.

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

FCS_CKM.2.1 The TSF shall destruct cryptographic keys in accordance with a specified cryptographic key destruction method [cryptographic key distribution method in [Table 5-5]] that meets the following: [list of standards in [Table 5-5]].

List of standards	Distribution target	Distribution method
TTAS.KO-12.0004/R1	User data encryption key of FCS_CKM.1(1)	transmit the user data encryption key securely with SEED(CBC) symmetric key encryption and hash function(SHA256) provided by validated cryptographic module
ISO/IEC 10118-3		
KS X ISO/IEC 18033-2	Session key of FCS_CKM.1(2) for the communication data encryption between TOE components	transmit the session key securely with public key encryption(RSAES) provided by validated cryptographic module

[Table 5-5] Cryptographic key distribution method

5.1.2.4 FCS_CKM.4 Cryptographic key destruction

Hierarchical to No other components.

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

FCS_CKM.4.1 The TSF shall destruct cryptographic keys in accordance with a specified cryptographic key destruction method [cryptographic key destruction method in [Table 5-6]] that meets the following: [list of standards in [Table 5-6]].

List of standards	Cryptographic key storage location	Destruction method	Destruction object	Destruction point
N/A	DB	Overwrite everything with "0x00"	User data encryption key (policy key)	When the administrator deletes the security policy
N/A	Memory	Overwrite everything with "0x00"	encryption keys on memory(public key, private key, policy key, TSF DEK)	When calling Agent process shutdown
N/A	Memory	Overwrite everything with "0x00"	session key	At the end of communication
N/A	Memory	Overwrite everything with "0x00"	policy key, TSF DEK	Immediately after use

[Table 5-6] Stored and used cryptographic key destruction

5.1.2.5 FCS_COP.1(1) Cryptographic operation(User data encryption)(SEED)

Hierarchical to No other components.

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

FCS_COP.1.1 The TSF shall perform [user data encryption and encryption] in accordance with a specified cryptographic algorithm [SEED(CBC) cryptographic algorithm in [Table 5-3]] and cryptographic key sizes [cryptographic key sizes for SEED(CBC) cryptographic algorithm in [Table 5-3]] that meet the following: [list of standards for SEED(CBC) cryptographic algorithm in [Table 5-3]].

5.1.2.6 FCS_COP.1(2) Cryptographic operation(User data encryption)(ARIA)

Hierarchical to No other components.

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

FCS_COP.1.1 The TSF shall perform [user data encryption and encryption] in accordance with a specified cryptographic algorithm [ARIA(CBC) cryptographic algorithm in [Table 5-3]] and cryptographic key sizes [cryptographic key sizes for ARIA(CBC) cryptographic algorithm in [Table 5-3]] that meet the following: [list of standards for ARIA(CBC) cryptographic algorithm in [Table 5-3]]..

5.1.2.7 FCS_COP.1(3) Cryptographic operation(User data encryption)(SHA2)

Hierarchical to No other components.

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

FCS_COP.1.1 The TSF shall perform [user data hash operation] in accordance with a specified cryptographic algorithm [cryptographic algorithm for one-way encryption in [Table 5-3]] and cryptographic key sizes [cryptographic key sizes for one-way encryption in [Table 5-3]] that meet the following: [list of standards for one-way encryption in [Table 5-3]].

5.1.2.8 FCS_COP.1(4) Cryptographic operation(TSF data encryption)(RSAES)

Hierarchical to No other components.

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

FCS_COP.1.1 The TSF shall perform [encryption/decryption for internal transport TSF data encryption key, encryption/decryption for user data encryption key(policy key)] in accordance with a specified cryptographic algorithm [cryptographic algorithm for public key encryption in [Table 5-4]] and cryptographic key sizes [cryptographic key sizes for public key encryption in [Table 5-4]] that meet the following: [list of standards for public key encryption in [Table 5-4]].

5.1.2.9 FCS_COP.1(5) Cryptographic operation(TSF data encryption)(SEED)

Hierarchical to No other components.

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

FCS_COP.1.1 The TSF shall perform [TSF data encryption and decryption] in accordance with a specified cryptographic algorithm [cryptographic algorithm for SEED(CBC) cryptographic algorithm in [Table 5-4]] and cryptographic key sizes [cryptographic key sizes for SEED(CBC) cryptographic algorithm in [Table 5-4]] that meet the following: [list of standards for SEED(CBC) cryptographic algorithm in [Table 5-4]].

5.1.2.10 FCS_COP.1(6) Cryptographic operation(TSF data encryption)(SHA2)

Hierarchical to No other components.

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

FCS_COP.1.1 The TSF shall perform [TSF data hash operation] in accordance with a specified cryptographic algorithm [cryptographic algorithm for one-way encryption in [Table 5-4]] and cryptographic key sizes [cryptographic key sizes for one-way encryption in [Table 5-4]] that meet the following: [list of standards for one-way encryption in [Table 5-4]].

5.1.2.11 FCS_RBG.1 Random bit generation (Extended)

Hierarchical to No other components.

Dependencies No dependencies.

FCS_RBG.1.1 The TSF shall generate random bits required to generate an cryptographic key using the specified random bit generator that meets the following [list of standards in [Table 5-7]].

List of standards	Random bit generator	Base function
KS X ISO/IEC 18031	HASH-DRBG-SHA256	HASH function

[Table 5-7] Random bit generator

5.1.3 User data protection(FDP)

5.1.3.1 FDP_UDE.1 User data encryption (Extended)

Hierarchical to No other components.

Dependencies FCS_COP.1 Cryptographic operation

FDP_UDE.1.1 The TSF shall provide a function that can encrypt/decrypt the user data to the TOE user according to the specified [encryption/decryption method by column, [None]]

5.1.3.2 FDP_RIP.1 Subset residual information protection

Hierarchical to No other components.

Dependencies No dependencies.

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

5.1.4 Identification and authentication

5.1.4.1 FIA_AFL.1 Authentication failure handling

Hierarchical to No other components.

Dependencies FIA_UAU.1 Timing of authentication

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

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met, the TSF shall [send mail to administrator and lock the account for 5 minutes].

5.1.4.2 FIA_IMA.1 TOE Internal mutual authentication (Extended)

Hierarchical to No other components.

Dependencies No Dependencies.

FIA_IMA.1.1 The TSF shall perform mutual authentication using [self-implementation authentication protocol] in accordance with [none] between [KSignSecureDB Server – KSignSecureDB DBAgent, KSignSecureDB Server – KSignSecureDB APIAgent].

5.1.4.3 FIA_SOS.1 Verification of secrets

Hierarchical to No other components.

Dependencies No Dependencies.

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet [acceptance criteria defined below].

- privacy information agreement

  - Password length: From 10 up to 30 characters consisting of a combination of English alphabets, numbers and special characters
  - Uppercase English alphabets: A – Z (26)
  - Lowercase English alphabets: a – z (26)
  - Numbers: 0 – 9 (10)
  - Special characters: !, @, #, $, %, ^, * (7)

- Verifying password rules : Combination of English characters, numbers, and special characters used three or more combinations and lengths must be 10 to 30 characters

5.1.4.4 FIA_UAU.2 User authentication before any action

Hierarchical to FIA_UAU.1 Timing of authentication

Dependencies FIA_UID.1 Timing of identification

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

5.1.4.5 FIA_UAU.4 Single-use authentication mechanisms

Hierarchical to No other components.

Dependencies No Dependencies.

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to [check the nonce value, check the CSRF token].

5.1.4.6 FIA_UAU.7 Protected authentication feedback

Hierarchical to No other components.

Dependencies FIA_UAU.1 Timing of authentication

FIA_UAU.7.1 The TSF shall provide only [masking(●) the entering password, the authentication fail message when authentication fail] to the user while the authentication is in progress.

5.1.4.7 FIA_UID.2 User identification before any action

Hierarchical to FIA_UID.1 Timing of identification

Dependencies No dependencies.

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

5.1.5 Security management

5.1.5.1 FMT_MOF.1 Management of security functions behaviour

Hierarchical to No other components.

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

FMT_MOF.1.1 The TSF shall restrict the ability to ***conduct management actions of*** the functions [[Table 5-8] security function list] to [the authorized administrator]

Security function component	**Security function**	**Management action**				**Authorized administrator**
		determine the behaviour	**disable**	**enable**	**modify the behaviour of**	
FAU_SAR.1	Grant the audit record review administrator role	O	X	X	O	Supervisor, System administrator
FDP_UDE.1	Manage of the encryption policy	O	X	X	X	Supervisor, Policy administrator
	Establish the encryption table	O	X	O	O	Supervisor, Encryption administrator
FMT_SMR.1	Grant the administrator role	O	X	X	O	Supervisor, System administrator

[Table 5-8] security function list

5.1.5.2 FMT_MTD.1 Management of TSF data

Hierarchical to No other components.

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

FMT_MTD.1.1 The TSF shall restrict the ability to ***manage*** [[Table 5-9] TSF data list] to [the authorized administrator].

security function component	TSF data	Manage				Authorized administrator
		change default	query	modify	delete	
FAU_STG.3	Limit of audit records	X	O	X	X	Supervisor, System administrator
FIA_UID.2	Information of administrators	O	O	O	O	Supervisor, System administrator
FIA_UAU.2	Administrator passwords	O	X	O	X	Supervisor, System administrator
FTA_SSL.5	Administrator connection maintenance time	X	O	X	X	Supervisor, System administrator
FTA_TSE.1	Access allowed IP	O	O	O	X	Supervisor, System administrator

[Table 5-9] TSF data list

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

Hierarchical to No other components.

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

FMT_PWD.1.1 The TSF shall restrict the ability to manage the password of [none] to [none].

1. [None]
2. [None]

FMT_PWD.1.2 The TSF shall restrict the ability to manage the ID of [none] to [none]

1. [None]
2. [None]

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

5.1.5.4 FMT_SMF.1 Specification of Management Functions

Hierarchical to No other components.

Dependencies No dependencies.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:
[Specified items in FMT_MOF.1 Management of security functions behaviour,
Specified items in FMT_MTD.1 Management of TSF data,
Specified items in FMT_PWD.1 Management of ID and password]

5.1.5.5 FMT_SMR.1 Security roles

Hierarchical to No other components.

Dependencies FIA_UID.1 Timing of identification

FMT_SMR.1.1 The TSF shall maintain the roles [the following authorized administrators].

- Supervisor
- Policy administrator
- System administrator
- Encryption administrator
- Audit records review administrator

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

5.1.6 Protection of the TSF

5.1.6.1 FPT_ITT.1 Basic internal TSF data transfer protection

Hierarchical to No other components.

Dependencies No dependencies.

FPT_ITT.1.1 The TSF shall protect the TSF data from *disclosure, modification* by **verifying encryption and message integrity** when the TSF data is transmitted among TOE's separated parts.

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

Hierarchical to No other components.

Dependencies No dependencies.

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

5.1.6.3 FPT_TST.1 TSF testing

Hierarchical to No other components.

Dependencies No dependencies.

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

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

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

5.1.7 TOE access

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

Hierarchical to FTA_MCS.1 Basic limitation on multiple concurrent sessions

Dependencies FIA_UID.1 Timing of identification

FTA_MCS.2.1 The TSF shall restrict the maximum number of concurrent sessions [belonging to the same **administrator** according to the rules for the list of management functions defined in FMT_SMF1.1]

a) limit the maximum number of concurrent sessions to 1 for management access by the same administrator who has the right to perform FMT_MOF.1.1 “Management actions” and FMT_MTD.1.1 “Management.”

b) limit the maximum number of concurrent sessions to { 1 } for management access by the same administrator who doesn’t have the right to perform FMT_MOF.1.1 “Management actions” but has the right to perform a query in FMT_MTD.1.1 “Management” only

c) [None]

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

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

Hierarchical to No other components.

Dependencies FIA_UAU.1 authentication or No dependencies.

FTA_SSL.5.1 The TSF shall *terminate* the administrator's interactive session after a [10 minutes of the administrator inactivity].

5.1.7.3 FTA_TSE.1 TOE session establishment

Hierarchical to No other components.

Dependencies No dependencies.

FTA_TSE.1.1 The TSF shall be able to refuse the **management access session of the administrator**, based on [Access IP, *the status of activating the management access session of the administrator having the same rights, [terminate the existing lower-level administrator when accessing the supervisor, unable to access lower-level administrator during supervisor connection]*].

5.2 Security assurance requirements

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

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

5.2.1 Security Target evaluation

5.2.1.1 ASE_INT.1 introduction

Dependencies No dependencies.

Developer action elements

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

Content and presentation elements

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

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

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

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

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

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

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

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

Evaluator action elements

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

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

5.2.1.2 ASE_CCL.1 Conformance claims

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

Developer action elements

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

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

Content and presentation elements

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

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

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

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

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

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

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

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

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

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

Evaluator action elements

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

5.2.1.3 ASE_OBJ.1 Security objectives for the operational environment

Dependencies No dependencies.

Developer action elements

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

Content and presentation elements

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

Evaluator action elements

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

5.2.1.4 ASE_ECD.1 Extended components definition

Dependencies No dependencies.

Developer action elements

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

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

Content and presentation elements

ASE_ECD.1.1C The statement of security requirements shall identify all extended security requirements. ASE_ECD.1.2C The extended components definition shall define an extended component for each extended security requirement.

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

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

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

Evaluator action elements

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

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

5.2.1.5 ASE_REQ.1 Stated security requirements

Dependencies ASE_ECD.1 Extended components definition

Developer action elements

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

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

Content and presentation elements

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

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

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

ASE_REQ.1.4C All operations shall be performed correctly.

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

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

Evaluator action elements

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

5.2.1.6 ASE_TSS.1 TOE summary specification

Dependencies ASE_INT.1 ST introduction

ASE_REQ.1 Stated security requirements

ADV_FSP.1 Basic functional specification

Developer action elements

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

Content and presentation elements

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

Evaluator action elements

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

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

5.2.2 Development

5.2.2.1 ADV_FSP.1 Basic functional specification

Dependencies No dependencies.

Developer action elements

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

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

Content and presentation elements

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

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

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

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

Evaluator action elements

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

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

5.2.3 Guidance documents

5.2.3.1 AGD_OPE.1 Operational user guidance

Dependencies ADV_FSP.1 Basic functional specification

Developer action elements

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

Content and presentation elements

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

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

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

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

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

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

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

Evaluator action elements

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

5.2.3.2 AGD_PRE.1 Preparative procedures

Dependencies No dependencies.

Developer action elements

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

Content and presentation elements

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

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

Evaluator action elements

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

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

5.2.4 Life-cycle support

5.2.4.1 ALC_CMC.1 TOE Lavelling of the TOE

Dependencies ALC_CMS.1 TOE CM coverage

Developer action elements

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

Content and presentation elements

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

Evaluator action elements

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

5.2.4.2 ALC_CMS.1 TOE CM coverage

Dependencies No dependencies.

Developer action elements

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

Content and presentation elements

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

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

Evaluator action elements

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

5.2.5 Tests

5.2.5.1 ATE_FUN.1 Functional testing

Dependencies ATE_COV.1 Evidence of coverage

Developer action elements

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

ATE_FUN.1.2D The developer shall provide test documentation.

Content and presentation elements

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

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

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

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

Evaluator action elements

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

5.2.5.2 ATE_IND.1 Independent testing - conformance

Dependencies ADV_FSP.1 Basic functional specification

AGD_OPE.1 Operational user guidance

AGD_PRE.1 Preparative procedures

Developer action elements

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

Content and presentation elements

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

Evaluator action elements

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

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

5.2.6 Vulnerability assessment

5.2.6.1 AVA_VAN.1 Vulnerability survey

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

Developer action elements

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

Content and presentation elements

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

Evaluator action elements

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

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

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

5.3 Security requirements rationale

5.3.1 Dependency rationale of security functional requirements

The following table shows dependency of security functional requirements.

No.	Security functional requirements	Dependency	Reference No.
1	FAU_ARP.1	FAU_SAA.1	3
2	FAU_GEN.1	FPT_STM.1	OE.TRUSTED_TIME STAMP
3	FAU_SAA.1	FAU_GEN.1	2
4	FAU_SAR.1	FAU_GEN.1	2
5	FAU_SAR.3	FAU_SAR.1	4
6	FAU_STG.3	FAU_STG.1	OE.SECURE_DBMS
7	FAU_STG.4	FAU_STG.1	OE.SECURE_DBMS
8	FCS_CKM.1(1)	[FCS_CKM.2 or FCS_COP.1]	10 or 12, 13, 14
		FCS_CKM.4	11
9	FCS_CKM.1(2)	[FCS_CKM.2 or FCS_COP.1]	10 or 15, 16, 17
		FCS_CKM.4	11
10	FCS_CKM.2	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	8, 9
		FCS_CKM.4	11
11	FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 orFCS_CKM.1]	8, 9
12	FCS_COP.1(1)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	8
		FCS_CKM.4	11
13	FCS_COP.1(2)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	8
		FCS_CKM.4	11
14	FCS_COP.1(3)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	-
		FCS_CKM.4	-

15	FCS_COP.1(4)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9
		FCS_CKM.4	11
16	FCS_COP.1(5)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9
		FCS_CKM.4	11
17	FCS_COP.1(6)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	-
		FCS_CKM.4	-
18	FCS_RGB.1	-	-
19	FDP_UDE.1	FCS_COP.1	12, 13, 14
20	FDP_RIP.1	-	-
21	FIA_AFL.1	FIA_UAU.1	24
22	FIA_IMA.1	-	-
23	FIA_SOS.1	-	-
24	FIA_UAU.2	FIA_UID.1	27
25	FIA_UAU.4	-	-
26	FIA_UAU.7	FIA_UAU.1	24
27	FIA_UID.2	-	-
28	FMT_MOF.1	FMT_SMF.1	31
		FMT_SMR.1	32
29	FMT_MTD.1	FMT_SMF.1	31
		FMT_SMR.1	32
30	FMT_PWD.1	FMT_SMF.1	31
		FMT_SMR.1	32
31	FMT_SMF.1	-	-
32	FMT_SMR.1	FIA_UID.1	27
33	FPT_ITT.1	-	-
34	FPT_PST.1	-	-
35	FPT_TST.1	-	-
36	FTA_MCS.2	FIA_UID.1	27
37	FTA_SSL.5	FIA_UAU.1	24
38	FTA_TSE.1	-	-

-

- FAU_GEN.1 has the dependency on FPT_STM.1, which is satisfied by the security objective OE. TRUSTED_TIMESTAMP for the operating environment. Because It records security related tests using reliable time stamps provided by the TOE operating environment.
- FAU_STG.3 and FAU_STG.4 have the dependency on FAU_STG.1 that is satisfied by the security objective OE.SECURE_DBMS for the operational environment. Because It protects the DBMS interacts with TOE that store audit data against unauthorized changes or deletions.
- FCS_COP.1(3) and FCS_COP.1(6) have the dependency on FDP_ITC.1, FDP_ITC.2, or FCS_CKM and FCS_CKM.4 that is satisfied because the Hash algorithm does not use the encryption key.
- FIA_AFL.1 and FIA_UAU.7 have the dependency on FIA_UAU.1, which is satisfied by FIA_UAU.2 in hierarchical relationship with FIA_UAU.1.
- FIA_UAU.2, FMT_SMR.1 and FTA_MCS.2 have the dependency on FIA_UID.1, which is satisfied by FIA_UID.2 in hierarchical relationship with FIA_UID.1.
- FTA_SSL.5 has the dependency on FIA_UAU.1, which is satisfied by FIA_UAU.2 in hierarchical relationship with FIA_UAU.1.

5.3.2 Dependency rationale of security assurance requirements

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

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

6. TOE Summary Specification

This chapter describes the SFRs of the TOE; security functions that satisfy the security assurance components; and the assurance methods

6.1 Security Alert

The TOE adopts the SFRs and provides the function of audit data generation, audit record review, audit data loss prevention and sending the alert mail. In addition, to protect the audit data, It prevents the unauthorized access to the DBMS in which the audit data stored and manages it.

6.1.1 Audit data generation

The TOE generates the audit data for the auditable events that occur during the operation. The generated audit data are stored in the storage (DBMS). The TOE uses a reliable time stamp (the time in the OS where the Server is installed) provided by the TOE operational environment to ensure that the audit data are generated sequentially.

Auditable events are generated and stored, based on the review period, task target, table owner, task type, task administrator, IP, task outcome (success/failure of the event).

The generated auditable events are as follows

Security functional component	Auditable event	Additional audit record
FAU_ARP.1	Actions taken due to potential security violations	
FAU_SAA.1	Enabling and disabling of any of the analysis mechanisms, Automated responses performed by the tool	
FAU_STG.3	Actions taken due to exceeding of a threshold	
FAU_STG.4	Actions taken due to the audit storage failure	
FCS_CKM.1(1)	Success and failure of the activity	

FCS_CKM.2	Success and failure of the activity (only applying to distribution of key related to user data encryption/decryption)	
FCS_CKM.4	Success and failure of the activity (only applying to destruction of key related to user data encryption/decryption)	
FCS_COP.1(1)	Success and failure of the activity	
FDP_UDE.1	Success and failure of user data encryption/decryption	
FIA_AFL.1	The reaching of the threshold for the unsuccessful authentication attempts and the actions taken, and the subsequent, if appropriate, restoration to the normal state	
FIA_IMA.1	Success and failure of mutual authentication Modify of authentication protocol	
FIA_UAU.2	All use of the authentication mechanism	
FIA_UAU.4	Attempts to reuse authentication data	
FIA_UID.2	All use of the administrator identification mechanism, including the administrator identity provided	
FMT_MOF.1	All modifications in the behaviour of the functions in the TSF	
FMT_MTD.1	All modifications to the values of TSF data	Modified values of TSF data
FMT_PWD.1	All changes of the password	
FMT_SMF.1	Use of the management functions	
FMT_SMR.1	Modifications to the user group of rules divided	
FPT_TST.1	Execution of the TSF self-tests and the results of the tests	Modified TSF data or execution code in case of integrity violation
FTA_MCS.2	Denial of a new session based on the limitation of	

	multiple concurrent sessions	
FTA_SSL.5	Locking or termination of interactive session	
FTA_TSE.1	Denial of a session establishment due to the session establishment mechanism All attempts at establishment of a user session	

6.1.2 Audit data review

The TOE stores the audit data in the audit trail storage (DBMS) and provides the function for the authorized administrator to review all audit data so that the administrator can appropriately interpret the information from the audit records. It also allows the audit data review based on AND conditions with the review period, task target (agent, table owner), task type, task administrator, IP and task outcome.

The authorized administrator (supervisor and audit record review administrator) can review and search the audit data by using the security management interface in KSignSecureDB Server.

6.1.3 Audit data loss prevention

The audit records generated by the TOE are stored in the storage (DBMS) provided by the TOE operational environment. The audit records stored in the storage (DBMS) are protected with access control against unauthorized deletion and modification.

The TOE checks the space in the audit record storage on a periodic basis; generates audit records on an event that exceeds the storage if it exceeds the threshold of the remaining space in the storage; and sends an alert (alert mail) to the authorized administrator. If the audit trail is full, the TOE ignores the audit detail to protect the audit records and send an alert (alert mail) to the authorized administrator.

- If the audit data reaches the default threshold of 80% of the total audit record storage capacity (based on the tablespace), an alert(alert mail) is sent to the authorized administrator. It is not allowed to change the default value.

- If the audit trail fills up the default threshold of 90% of the total audit record storage capacity (based on the tablespace), it ignores events audited at the time when the audit trail is full and sends an alert (alert mail) to the authorized administrator. It is not allowed to change the default value.

6.1.4 Security Alert

The TOE applies a combination of rules that indicate potential security violations in the audit data, and performs security alarm by sending an alert email to the administrator defined in case of a violation. Potential security violations are as follows:

- When the administrator authentication has failed;
- When the threshold of the defined number of unsuccessful authentication attempts has been reached;
- When a user access control policy has been violated;
- When the integrity verification of the TOE configuration files has failed;
- When the license verification has failed;
- When the self test of the validated cryptographic module has failed;

6.1.5 SFR Mapping

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

6.2 Cryptographic Function Support

The TOE supports cryptography using the validated cryptographic module KSignCASE64 v2.5.2.0 in the policy transmission interval for the cryptographic support between the TOE components. Details of the validated cryptographic module included in the TOE are as follows.

Item	Specification
Cryptographic module name	KSignCASE64 v2.5.2.0
Developer	KSign Co., Ltd
Validation date	2023. 10. 16.
Validation level	VSL1
Validation number	CM-237-2028.10
Expiration Date	2028. 10. 16.

6.2.1 Cryptographic Support

The object that communicates for the protection of the TSF data transmitted inside the TOE generates the certificate (private key and public key) with RSAES(2048bit); distributes the TSF data encryption key using the certificate; encrypts the TSF data with SEED-CBC algorithm; sends the data and verifies the integrity of the transmitted data through the one-way algorithm (SHA-256); and decrypts the encrypted data by using the TSF data encryption key distributed.

The user data encryption key which is distributed KSignSecureDB DBAgent and KSignSecureDB APIAgent is encrypted and sent securely with SEED(CBC) algorithm and hash function (SHA-256) provided by the validated cryptographic module

The user data and TSF data are encrypted by the symmetric key cryptographic operation. For this purpose, the 128/192/256-bit cryptographic key is generated through HASH_DRBG(SHA-256) conforming to KS X ISO/IEC 18031(2018). The master key used to encrypt the user data encryption key is also generated with 128-bit through HASH_DRBG(SHA256) conforming to KS X ISO/IEC 18031(2018). The key for encryption TSF encryption key is generated with 128-bit through PBKDF2-HMAC(SHA256) conforming to TTAK.KO-12.0334-Part2.

When a cryptographic key necessary for the asymmetric key cryptographic operations generated, 2048-bit cryptographic key is generated through RSAES algorithm that complies with KS X ISO/IEC 18033-2(2017) standard.

The user data encryption key(policy key) managed by KSignSecureDB Server is encrypted with SEED(CBC) algorithm, stored and managed in the DBMS, the TSF data encryption key is encoded with KSign-implemented encoding method on memory at start-up, and used to encrypt and decrypt the TSF data with decoding. the decoded TSF data encryption key is deleted on memory after use.

The master key used for the encryption of the cryptographic key is encrypted through RSAES (2048bit) and managed in the DBMS.

The validated cryptographic module is used for the supported cryptographic algorithm, and the information on algorithms by use is as follows.

Item		Algorithm	Key Length	List of Standard
Mutual communication, (KSignSecureDB Server ↔ KSignSecureDB DBAgent) (KSignSecureDB Server ↔ KSignSecureDB APIAgent)	transmitted Data Encryption	SEED (CBC)	128bit	TTAS.KO-12.0004/R1
	Key Distribution	RSAES	2048bit	KS X ISO/IEC 18033-2
	Integrity	SHA256	N/A	ISO/IEC 10118-3
Master key (KEK) encryption		RSAES	2048bit	KS X ISO/IEC 18033-2
Policy key (DEK) encryption		SEED (CBC)	128bit	TTAS.KO-12.0004/R1
TSF data encryption key (DEK) encryption				
User data encryption		SEED (CBC)	128bit	TTAS.KO-12.0004/R1
		ARIA (CBC)	128/192/256bit	KS X 1213-1
		SHA256/512	N/A	ISO/IEC 10118-3
TSF data encryption		SEED (CBC)	128bit	TTAS.KO-12.0004/R1
Store administrator password		SHA256	N/A	ISO/IEC 10118-3
TOE module integrity		SHA256	N/A	

6.2.2 Cryptographic key destruction

If a cryptographic key loaded on the memory upon key generation, distribution and operation expires, its random bits are all overwritten with 0x00 to destroy the cryptographic key.

- The cryptographic key-related information is deleted:

List of standards	Cryptographic key storage location	Destruction method	Destruction object	Destruction point
N/A	DB	Overwrite everything with "0x00"	User data encryption key (policy key)	When the administrator deletes the security policy
N/A	Memory	Overwrite everything with "0x00"	encryption keys on memory (public key, private key, policy key, TSF DEK)	When calling Agent process shutdown
N/A	Memory	Overwrite everything with "0x00"	session key	At the end of communication
N/A	Memory	Overwrite everything with "0x00"	policy key, TSF DEK	Immediately after use

6.2.3 Random bit generation

The TOE uses HASH_DRBG (SHA-256) algorithm through the random number generator of the validated cryptographic module KSignCASE64 v2.5.2.0 whose safety and suitability for the implementation have been confirmed by the cryptographic module validation scheme, and generates random numbers necessary for generating cryptographic keys.

6.2.4 SFR Mapping

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

6.3 User data protection

6.3.1 User data protection

The TOE provides the function of encrypting/decrypting the data stored in the DBMS under the protection by the unit of column by using KSignCASE64 v2.5.2.0, a validated cryptographic module, and generates different ciphertext values for the same plaintexts.

Furthermore, it ensures that any previous information content of a resource is made unavailable by deleting the original data after encryption and deleting the encrypted data after decryption in the DBMS under the protection.

6.3.2 SFR Mapping

SFR to be satisfied: FDP_UDE.1, FDP_RIP.1

6.4 Identification and Authentication

6.4.1 Identification and Authentication

The authorized administrator shall be identified through the administrator authentication (ID, password) to be allowed to perform the security management and cannot use any security management function without undergoing such authentication process. The administrator authentication information is transmitted through a web-based browser, and the authentication information is securely transmitted through the HTTPS communication between the web browser and KSignSecureDB Server.

If the administrator login attempts are unsuccessful for five times, the TOE locks the account for 5 minutes and sends alert mail to administrator. If the identification and authentication succeed normally after 5 minutes, the account is unlocked.

6.4.2 Protection of authentication data

The TOE provides the following to protect the feedback when the administrator password is entered:

- The password used for the authentication is masked with "●" to prevent them from being disclosed.
- It does not provide a reason for authentication failure in case of an unsuccessful authentication attempt.

The TOE provides the following to prevent the reuse of the administrator authentication information:

- Prevention of the reuse of the administrator authentication information: To avoid a CSRF (Cross Side Request Forgery) attack, the TOE allocates a nonce and CSRF token to each page prior to the administrator authentication. and restrict access if the transmitted the nonce and CSRF token is do not matched the allocated.

6.4.3 Password policy validation

The validity of password values is verified in accordance with the defined password combination rules when the administrator password is generated or modified.

The TOE provides the following verification mechanisms for password.

- Password length: From 10 up to 30 characters consisting of a combination of English alphabets, numbers and special characters
- Uppercase English alphabets: A – Z (26)
- Lowercase English alphabets: a – z (26)
- Numbers: 0 – 9 (10)
- Special characters: !, @, #, $, %, ^, * (7)
- Verifying password rules: Combination of English characters, numbers, and special characters used three or more combinations and lengths must be 10 to 30 characters

6.4.4 Mutual authentication between components

The TOE performs mutual authentication through mutual authentication protocol between KSignSecureDB Server, KSignSecureDB DBAgent and KSignSecureDB APIAgent at start-up of KSignSecureDB DBAgent and APIAgent, and the detailed mechanism is as follows.

1. KSignSecureDB DBAgent or APIAgent sends Agent Name and the hash value of Agent Name, IP, Port, OS type to KSignSecureDB Server at start-up
2. KSignSecureDB Server looks up the agent information registered through the security management interface with Agent Name which is received.
3. KSignSecureDB Server verifies the IP that requested the connection matches Agent IP is found.
4. KSignSecureDB Server makes the SHA256 hash value of the found Agent Name, IP, Port, OS Type and compares it with the hash value received, and sends the Agent authentication result
5. If the Agent authentication is successful, KSignSecureDB Server sends the SHA256 hash value of its IP and Port to KSignSecureDB DBAgent or APIAgent
6. KSignSecureDB DBAgent or APIAgent makes the SHA256 hash value of the Server IP and Port recorded in the configuration file and compares it with the hash value received, and sends the Server authentication result

6.4.5 SFR Mapping

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

6.5 Security Management

6.5.1 Management of security functions

The TOE calls the function of the security management access control only if the self-enforced identification and authentication are successfully carried out. Only an administrator permitted by the authorized administrator (supervisor) is allowed to access the security management interface through a secure channel (SSL).

The roles of the authorized administrator provided by the TOE are as follows:

- Supervisor: The top administrator has the privilege of system management, policy management, establishing and performing the table encryption/decryption and viewing audit record, and can create lower-level administrators other than supervisor.
- Policy administrator: The policy administrator has the privileges of the protected DBMS management and key(policy) registration.
- System administrator: The system administrator has the privilege of system management menu, generation, deletion and modification of the administrator and system configuration.
- Encryption administrator: The encryption administrator has the privilege of establishing and performing the table encryption.
- Audit records review administrator: The audit record review administrator has the privilege of reviewing the audit records.

the TOE provides management actions for security functions as follows.

Security function component	Security function	Management action				Authorized administrator
		determine the behaviour	disable	enable	modify the behaviour of	
FAU_SAR.1	Grant the audit record review administrator role	O	X	X	O	Supervisor, System administrator
FDP_UDE.1	Manage of the encryption policy	O	X	X	X	Supervisor, Policy administrator
	Establish the encryption table	O	X	O	O	Supervisor, Encryption administrator
FMT_SMR.1	Grant the administrator role	O	X	X	O	Supervisor, System administrator

The TOE provides the functions to management for TSF data as follows.

security function component	TSF data	Manage				Authorized administrator
		change default	query	modify	delete	
FAU_STG.3	Limit of audit records	X	O	X	X	Supervisor, System administrator
FIA_UID.2	Information of administrators	O	O	O	O	Supervisor, System administrator
FIA_UAU.2	Administrator passwords	O	X	O	X	Supervisor, System administrator
FTA_SSL.5	Administrator connection maintenance time	X	O	X	X	Supervisor, System administrator
FTA_TSE.1	Access allowed IP	O	O	O	X	Supervisor, System administrator

6.5.2 ID and password management

It is enforced that the authorized administrator changes the password upon the initial access to the security management interface. The authorized administrator (supervisor, system administrator) can create an administrator and change the administrator password through the security management interface.

The validity of password values is verified in accordance with the defined password combination rules when the administrator password is generated or modified. The TOE provides the following verification mechanisms for passwords:

- Password length: From 10 up to 30 characters consisting of a combination of English alphabets, numbers and special characters

- Uppercase English alphabets: A – Z (26)
- Lowercase English alphabets: a – z (26)
- Numbers: 0 – 9 (10)
- Special characters: !, @, #, $, %, ^, * (7)
- Verifying password rules: Combination of English characters, numbers, and special characters used three or more combinations and lengths must be 10 to 30 characters

6.5.3 SFR Mapping

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

6.6 Protection of the TSF

6.6.1 Internal TSF data transfer protection

The TOE performs KSign-implemented encrypted communication for policy transfer with the aim of the internal TSF data transfer protection, and protects the communication by using the validated cryptographic module KSignCASE64 v2.5.2.0 as follows:

1. KSignSecureDB DBAgent or APIAgent transfers the Agent Flag value with Helo message to KSignSecureDB Server
2. Agent Flag is the data that encrypted nonce value, current time, unique code value and checksum (SHA256) with the certificate in order to prevent MITM(man-in-the-middle) or reply attack
3. KSignSecureDB Server receives the Agent Flag value, it decrypts the value and verifies the Flag value
4. The Flag value is verified by decrypting the value received from the Agent with the private key to check the value from the Agent; checking if the time value that indicates when it was encrypted and sent matches the time value that indicates when it was sent as plaintexts; and checking the SHA256 checksum value for the transmitted data and the unique code value.

5. The communication is terminated if the Flag value verification fails. A Hello response (ack + received nonce value) message is sent to KSignSecureDB DBAgent or APIAgent if the Flag value verification succeeds.

6. KSignSecureDB DBAgent or APIAgent that received the response message decrypts the nonce value with the private key to ensure it is the value that it sent. If not, it terminates the communication. Otherwise, the Session Key (SEED-CBC, 128 bits) is generated by the random number generator in the validated cryptographic module.

7. Encrypts the generated Session Key with the certificate(RSAES, 2048 bits).

8. KSignSecureDB DBAgent or APIAgent generates the hash value (SHA256) of the Session Key encrypted with the certificate and transfers the encrypted Session Key and the hash value to KSignSecureDB Server.

9. KSignSecureDB Server generates the hash value (SHA256) of the Session Key received from KSignSecureDB DBAgent or APIAgent, and compares it against the received hash value to verify whether they match or not.

10. KSignSecureDB Server compares the hash values. If they do not match, the communication is terminated. If they match, it decrypts (RSAES, 2048 bits) the Session Key encrypted with its private key.

11. KSignSecureDB Server and KSignSecureDB DBAgent or APIAgent send and receive the data by encrypting them with the Session Key shared between the two parties.

12. When sending or receiving the data encrypted with the Session Key between KSignSecureDB Server and KSignSecureDB DBAgent or APIAgent, Verify the hash values for the encrypted data is generated (SHA256) to make sure that they match.

13. If the hash values match, decrypt (SEED-CBC, 128 bits) the data with the shared Session Key to obtain the plaintexts. If not, the communication is terminated.

14. The Session Key generated in each stage is destroyed by initializing the memory variables with 0x00 after the use. In case the communication is cut off in KSignSecureDB Server and KSignSecureDB DBAgent or APIAgent, the memory variable of the corresponding Session Key is initialized with 0x00 so that the cryptographic key is normally destroyed.

6.6.2 Protection of stored TSF data

The TOE stores and manages the TSF data to be protected by encrypting them to protect the stored TSF data from unauthorized disclosure or modification.

Information required to be encrypted includes administrator passwords, TOE set value information (DB storage information and configuration file information) and so on. An administrator password is encrypted with SHA256, and the TOE set value information is encrypted with SEED-CBC 128 bits.

The TOE set values are included in and exist inside KSignSecureDB Server, KSignSecureDB DBAgent and KSignSecureDB APIAgent, which are the TOE components

Information required to be encrypted, from configuration file information in KSignSecureDB Server, includes web server SSL certificate storage path, certificate password, DB URL, DB account ID and DB account password. Information managed in the Server policy database and required to be encrypted includes Agent IP, Agent port, Agent installation path, policy DB port, policy DB service name, security administrator account, security administrator password, JDBC URL, administrator password, mail server IP, mail server port, mail server username and mail server password.

Information located inside the configuration file of KSignSecureDB DBAgent and required to be encrypted includes domain name, basic agent path, agent IP address, agent port, server IP address, server port, shared memory ID.

Information located inside the configuration file of KSignSecureDB APIAgent and required to be encrypted include basic agent path, agent IP address, agent port, server IP address, server port, shared memory ID and certificate password.

The data encryption key (DEK) for the protection of the TSF data encrypts the TOE set values with SEED-CBC 128 bits.

The data encryption key (DEK) for the protection of the TSF data is securely encrypted and protected with the SEED-CBC 128-bit key encryption key (KEK).

The TSF data encryption key (DEK) and the key encryption key (KEK) generated by method of password based key derivation are generated through KSignCASE64 v2.5.2.0, which is a secure validated module.

The TSF data encryption key (DEK) for encrypting the TOE set values is generated by the validated module. The key encryption key (KEK) for securely protecting the data encryption key (DEK) is also generated by the validated module and password based key derivation and encrypts the data encryption key (DEK) and stores it in a file.

Upon the operation of the product, the key file that contains the encryption key encrypted with the key encryption key (KEK) generated through the key derivation based on the password entered by the administrator is decrypted to obtain the TSF data encryption key (DEK), which is encoded with KSign-implemented encoding method and stored in the policy DB in case of KSignSecureDB Server, and encoded with KSign-implemented encoding method and stored in the shared memory area in the same way as the Server in case of KSignSecureDB DBAgent and KSignSecureDB APIAgent.

After loading the TSF data encryption key(DEK) on memory and shared memory, destroy the key encryption key(KEK) generated by key derivation based on password entered by the administrator at start-up.

For the TSF data encryption, KSignSecureDB Server takes the KSign-implemented encoded TSF data encryption key from memory and decode it, encrypts the TSF data and save it the managed DB or the configuration file.

For the TSF data encryption, KSignSecureDB DBAgent and KSignSecureDB APIAgent takes the KSign-implemented encoded TSF data encryption key from memory and decode it, encrypts TOE set values and save it the configuration file.

The TSF data list to be protected and used algorithms applied are as follows.

TSF data	Algorithm and Data
Administrator password	SHA256(password)
TOE set value	SEED-CBC(data)
TSF data encryption key User data encryption key(policy key) Certificate private key	SEED-CBC(key)

6.6.3 Integrity verification

The integrity verification of the TOE is the function of determining the corruption of the TSF execution data. The authorized administrator performs the integrity verification through KSignSecureDB Server. Hash values of the files tested for the integrity verification are generated upon the initial start-up, and SHA-256 is used as the hash algorithm. In case of the integrity verification upon the initial start-up, periodically during normal operation and by authorized administrator, the operation stops, generates audit data on the integrity verification and sends alert mail to administrator if the corruption is detected.

The conditions for the occurrence of integrity verification are as follows.

TOE component	Condition of occurrence
KSignSecureDB Server	during start-up, periodically during normal operation, by authorized administrator
KSignSecureDB DBAgent	during start-up, by authorized administrator
KSignSecureDB APIAgent	during start-up, by authorized administrator

- Integrity verification files of KSignSecureDB Server are as follows.

Target		Description
Directory	File	
agent	All files in directory	DBAgent and APIAgent certificate
bin	libKCASECRYPTO.so	Validated cryptographic module
	libKCASECRYPTO_jni.so	Interface of Validated cryptographic module
	KSignSecureDBServer	Core execution script
	startServer.sh	Server startup script
	stopServer.sh	Server shutdown script
	InitSecureDB.dat	TSF data encryption key file
	InitSecureDB.ath	Hash value of TSF data encryption key file
cert	signCert.der	Server certificate
	signPri.key	Server encrypted private key file
conf	KSDB_Integrity_info.ini	Hash value for all of integrity verification file of Server
	KSDB_PSVR.properties	Server configuration file

jdbcDriver	All files in directory	JDBC driver library for Protected DBMS
license	license.dat	License file
ssl	securedb_server.keystore	Certificate file for SSL communication
WebContent	All files in directory	Server execution file (Java library, content of web page, etc.)

- Integrity verification files of KSignSecureDB DBAgent are as follows.

Target		Description
Directory	File	
bin	libKCASECRYPTO.so	Validated cryptographic module
	libKCASECRYPTO_jni.so	Interface of Validated cryptographic module
	KSignSecureDBAgent	Core execution script
	startKeyAgent.sh	DBAgent startup script
	stopKeyAgent.sh	DBAgent shutdown script
	InitSecureDB.dat	TSF data encryption key file
	InitSecureDB.ath	Hash value of TSF data encryption key file
cert	signCert.der	DBAgent certificate
	signPri.key	DBAgent encrypted private key file
	serverCert.der	Server certificate
conf	KSDB_KAGT.properties	DBAgent configuration file
lib	KSignLicenseVerify-2.7.3.jar	License validation library
	KSDB_KAGT.jar	DBAgent core library
	KSDB_SSL.jar	KSign-implemented encrypted communication library
	libKSDB_SHM.a	Shared memory library
license	license.dat	License file
DB_LIB/lib	KSDB_JFT.jar	DBAgent cryptographic library
DB_LIB/conf	KSDB_JFT.properties	DBAgent cryptographic configuration file

Integrity verification files of KSignSecureDB APIAgent are as follows.

Target	Description

Directory	File	
bin	libKCASECRYPTO.so	Validated cryptographic module
	libKCASECRYPTO_jni.so	Interface of Validated cryptographic module
	KSignSecureDBAgent	Core execution script
	startKeyAgent.sh	APIAgent startup script
	stopKeyAgent.sh	APIAgent shutdown script
	InitSecureDB.dat	TSF data encryption key file
	InitSecureDB.ath	Hash value of TSF data encryption key file
cert	signCert.der	APIAgent certificate
	signPri.key	APIAgent encrypted private key file
	serverCert.der	Server certificate
conf	KSDB_KAGT.properties	APIAgent configuration file
lib	KSignLicenseVerify-2.7.3.jar	License validation library
	KSDB_KAGT.jar	DBAgent core library
	KSDB_SSL.jar	KSign-implemented encrypted communication library
	libKSDB_SHM.a	Shared memory library
license	license.dat	License file
API_LIB/lib	KSDB_JAP.jar	APIAgent cryptographic library
API_LIB/conf	KSDB_JAP.properties	APIAgent cryptographic configuration file

6.6.4 TSF Self Tests

The TOE performs the self tests on KSignSecureDB Server, KSignSecureDB DBAgent and KSignSecureDB APIAgent during start-up and periodically during normal operation based on the KSignSecureDB Server's startup date to demonstrate that the TSF is operated correctly.

The self tests includes process verification and validated cryptographic module self-test. Process verification compares the process ID at startup with the process ID currently running to verify that the process is not modulated. After process verification and validated cryptographic module self-test, The TOE records the result of them as audit data. If the result is fail, send alert mail to administrator.

6.6.5 SFR Mapping

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

6.7 TOE Access

6.7.1 Administrator Session Restrictions

The TOE limits the maximum number of concurrent sessions that belong to the same administrator to 1 in accordance with the rule for re-access requests (accessible IP) by the authorized administrator with the same account or the same privilege after the administrator access is made. In addition, the administrator access sessions are permitted in accordance with the allowable IP for the administrator access (up to 2 IP addresses by default) registered through the security management interface, and access sessions by non-permitted IPs are restricted.

If the supervisor is log-in, a lower-level administrator is not allowed to access. If supervisor accesses while a lower-level administrator is log-in, the access by a lower-level administrator is cancelled. Furthermore, if an access attempt is made with the account which is the same as supervisor account, the preceding access is cancelled. In case of login with the account or the privilege which is the same as that of a lower-lever administrator, the preceding access is cancelled. In addition, the administrator session is terminated after a specified time interval of inactivity. In this case, a lower-level administrator refers to the system administrator, the policy administrator, the encryption administrator and the audit record review administrator, except for the supervisor.

6.7.2 Terminate the Session in the Security Management Interface

The authorized administrator accesses the TOE security management interface through a web browser on the administrator PC once the TOE is distributed/installed normally. The TOE allows access to the security management interface (HTTPS) only if the administrator trying to make explicitly permitted access completes the identification and authentication process successfully.

After the authorized administrator successfully logs on to the security management interface (web UI) of the TOE and remains inactive for a specified allowable interval, the TOE terminates a session that interacts with the authorized administrator. The default value of the allowable interval of inactivity is set as 10 minutes and cannot be modified. During the session termination, the TOE disables all activities from the existing sessions and terminates the session. If the authorized administrator whose activities have been disabled tries to use the security management interface again, the TOE allows the access to the security management interface by creating a new session only if the re-authentication of the administrator (administrator identification and authentication) is successfully completed. The TOE generates the audit data on the result of such events, that is, the execution result of session termination in the security management interface.

6.7.3 SFR Mapping

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