Security Target

CubeOne V2.5 SP1

1.3

2020-11-05

eGlobal Systems Co., Ltd

Security Target

eGlobal Systems Co., ltd
4F ilhwan Bldg., 54, Seolleung-ro 93-gil
Gangnam-gu, Seoul, Korea, 135-513
www.eglobalsys.co.kr

Telephone +82-2-6447-6988 •
Fax +82-2-6447-6989 •

Revision

Ver.	Date	Content	writer
1.0	2020.05.12	First Release in English	h.m.yang
1.1	2020.08.20	Incorporation of the review findings	h.m.yang
1.2	2020.09.11	Incorporation of the review findings	h.m.yang
1.3	2020.11.05	Certification Center of the review findings	h.m.yang

Table of Contents

List of Figures

List of Tables

1. ST Introduction

This Document is ST of CubeOne V2.5 SP1 developed by eGlobal Systems Co. for Database Encryption which is aimed for EAL+1 level of CC.

1.1. ST reference

Item	Specification
Title	CubeOne V2.5 SP1 Security Target
Document identification	CubeOne_ST_V2.5.1.2_SP1
Version	V2.5.1.2_SP1
Developer	eGlobal Systems Co., ltd.
Issue date	2020.09.11
Common Criteria	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)
Common Criteria version	CC V3.1 r5
Evaluation Assurance Level	EAL1+(ATE_FUN.1)
Keywords	Database, Encryption

1.2. TOE reference

Item	Specification
TOE name	CubeOne V2.5 SP1
TOE type	Database, Encryption
TOE version	V2.5 SP1

Item		Specification
Detail version		rev.0002
TOE components	CubeOne Manager	- CubeOne_Manager_V2.5.00.01_SP1 : CubeOne_Manager_V2.5.00.01_SP1.exe
	CubeOne Server	[Plug-In] - CubeOne_Server_V2.5.00.01_SP1_A64_7.1_TI6 : CubeOne_Server_V2.5.00.01_SP1_A64_7.1_TI6.tar - CubeOne_Server_V2.5.00.01_SP1_A64_7.1_DB10.5 : CubeOne_Server_V2.5.00.01_SP1_A64_7.1_DB10.5.tar - CubeOne_Server_V2.5.00.01_SP1_S64_5.11_TI6 : CubeOne_Server_V2.5.00.01_SP1_S64_5.11_TI6.tar - CubeOne_Server_V2.5.00.01_SP1_H64_B.11.31_TI6 : CubeOne_Server_V2.5.00.01_SP1_H64_B.11.31_TI6.tar - CubeOne_Server_V2.5.00.01_SP1_L64_3.10_TI6 : CubeOne_Server_V2.5.00.01_SP1_L64_3.10_TI6.tar [API] - CubeOne_Server_V2.5.00.01_SP1_A64_7.1_API : CubeOne_Server_V2.5.00.01_SP1_A64_7.1_API.tar - CubeOne_Server_V2.5.00.01_SP1_S64_5.11_API : CubeOne_Server_V2.5.00.01_SP1_S64_5.11_API.tar - CubeOne_Server_V2.5.00.01_SP1_H64_B.11.31_API : CubeOne_Server_V2.5.00.01_SP1_H64_B.11.31_API.tar
	CubeOne Security Server	CubeOne_SServer_V2.5.00.01_SP1_L64_3.10_MA : CubeOne_SServer_V2.5.00.01_SP1_L64_3.10_MA.tar
	CubeOne Beacon	- CubeOne_Beacon_V2.5.00.01_SP1 : CubeOne_Beacon_V2.5.00.01_SP1.tar.gz
Developer		eGlobal Systems Co., Ltd

1.3. TOE overview

CubeOne V2.5 SP1 (hereinafter referred to as "TOE") is the product of eGlobal Systems Co. for database encryption. TOE performs the function of preventing the unauthorized disclosure of confidential information by encrypting column data in table of database (hereinafter referred to as "DB").

1.3.1. 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 ST can be grouped into the 'plug-in type' and 'API type', depending on the TOE operation type. The TOE can support both types.

TOE has following components.

Item	Specification
CubeOne Manager	Configure and control cryptographic policy like role definition of TOE
CubeOne Server	Perform cryptographic operation of user data for TOE
CubeOne Security Server	Save cryptographic policy and security audit log of TOE.
CubeOne Beacon	Perform latent violation analysis and security alert of TOE.

The cryptographic keys and TSF data used in encryption/decryption process must be created and controlled by CubeOne Manager. And cryptographic keys and TSF data are encrypted using approved algorithm of validated cryptographic module.

1.3.2. TOE usage and major security features

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

1.3.3. TOE operational environment

The TOE operational environment defined in this ST can be classified into two types: plug-in type and API type. The plug-in type, which is installed in the protected DB server, performs encryption/decryption of the user data and API type which is installed in Application server, which is not protected DB server, encrypts/decrypts user data on it. CubeOne Beacon and CubeOne Security Server are installed in the same server. The authorized administrator can connect to CubeOne Manager for security control. The authorized user can connect CubeOne Beacon to check security alert and audit log.

1.3.3.1. Plug-In Type

The authorized administrator can create encryption/decryption key and make cryptographic policy through GUI of CubeOne Manager. CubeOne Manager send/receive TSF data to CubeOne Security Server when login/logout. CubeOne Server encrypts user data marked by CubeOne Manager and delete original table. And authorized user can decrypt data according to cryptographic policy and send it to Application server.

CubeOne Security Server can store user data encryption/decryption keys and policies transmitted from CubeOne Manager, and encryption/decryption execution history generated in CubeOne Server and audit log generated from TOE components.

CubeOne Security Server can store statistic data generated by CubeOne Server and audit log generated by component of TOE and send cryptographic policy by request of CubeOne Server.

The authorized log user can check security alert and audit log through CubeOne Beacon.

Figure 1. show the general operational environment of the plug-in type

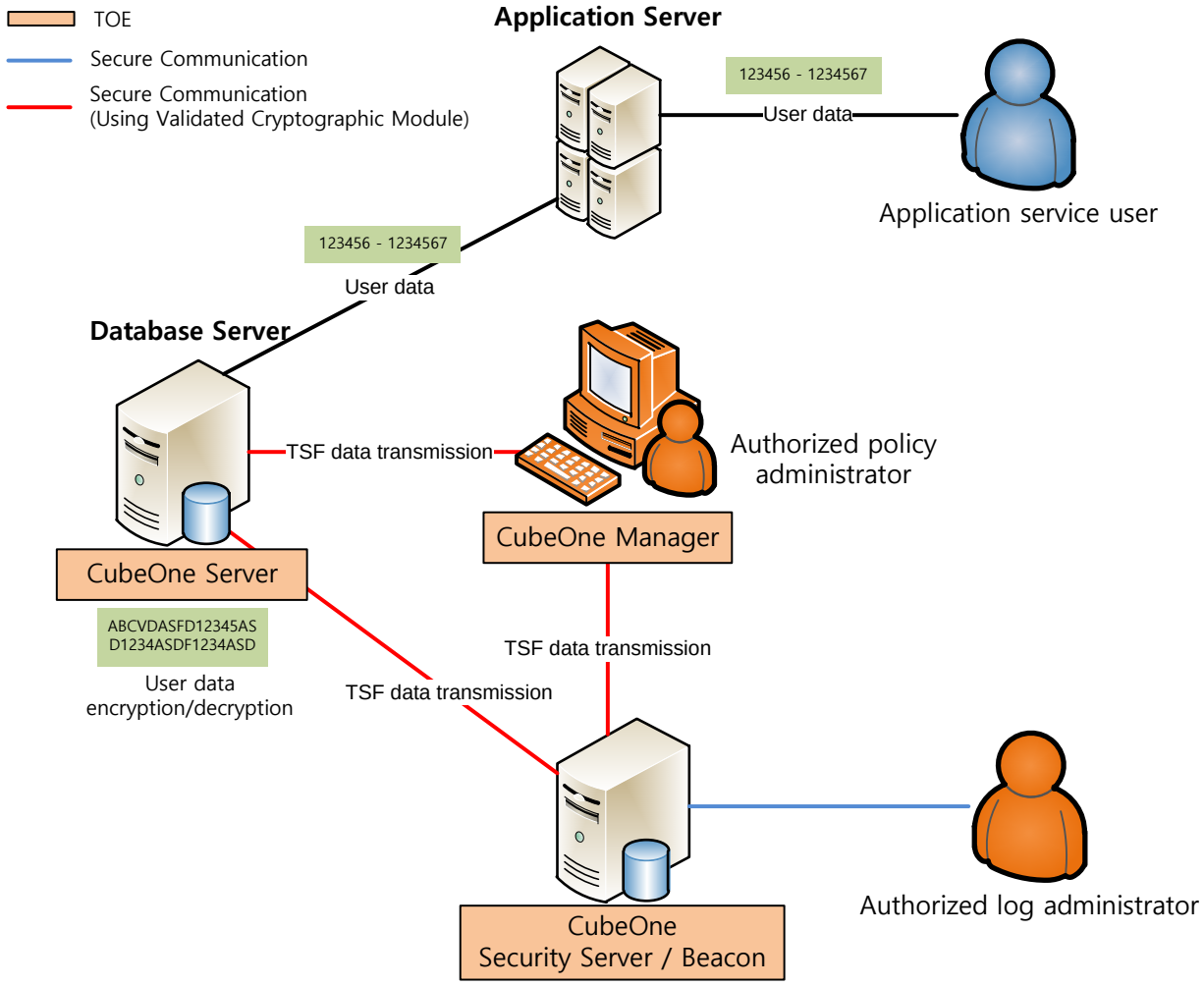


Figure 1. Plug-in type operational environment

1.3.3.2. API Type

The authorized administrator can create encryption/decryption key and make cryptographic policy and select column for encryption, set it to CubeOne Server through GUI of CubeOne Manager. CubeOne Manager send/receive TSF data to/from CubeOne Security Server when login/logout. The application developer can encrypt/decrypt data with API provided by TOE and must delete the original data after encrypt it. The application user must encrypt data before store it at CubeOne Server and it is decrypted while querying it for application user.

CubeOne Security Server can store user data encryption/decryption keys and policies transmitted from CubeOne Manager, and statistic data generated by CubeOne Server and audit log generated by component of TOE and send cryptographic policy by request of CubeOne Server.

The authorized log user can check security alert and audit log through CubeOne Beacon.

Figure 2. show the general operational environment of the API type.

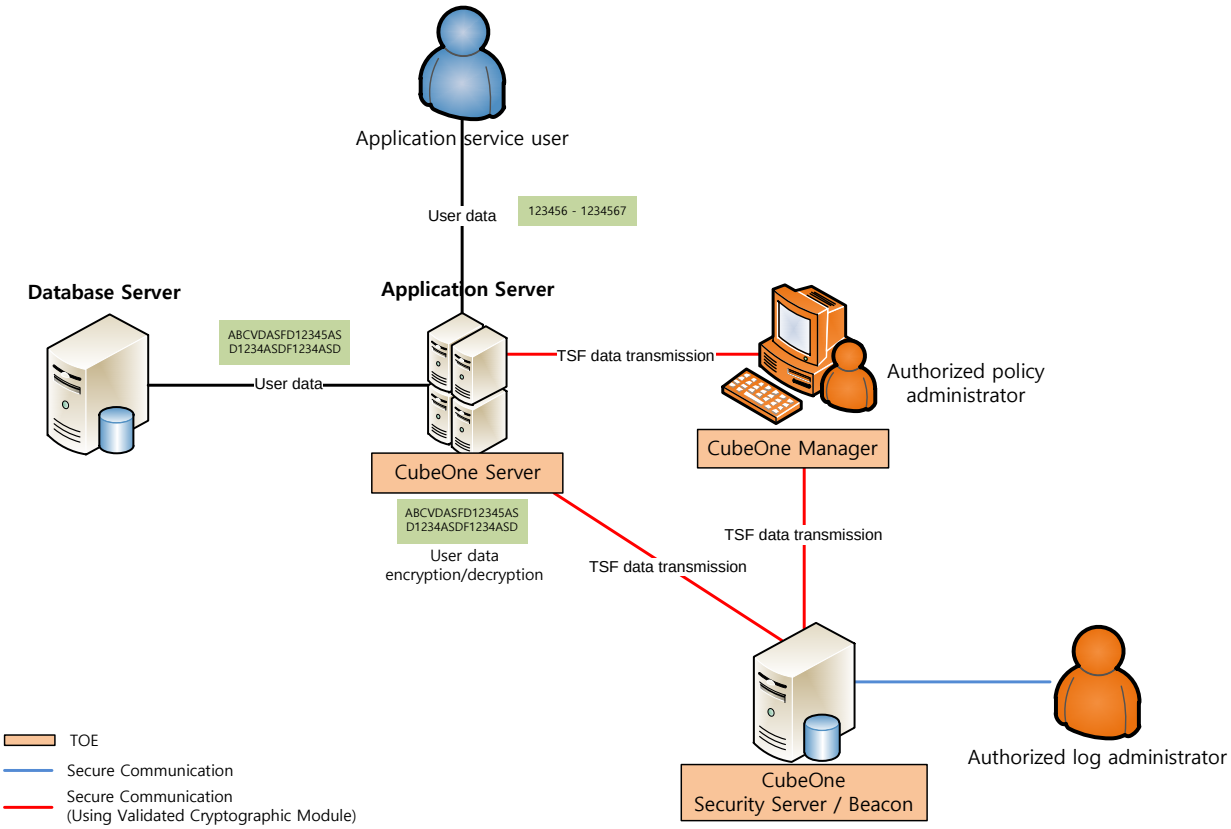


Figure 2. API operational environment

The communication channel between components of TOE shall be encrypted using approved algorithm of validated cryptographic module. And the reliable communication between authorized log administrator and WEB Server shall be guaranteed by using OpenSSL.

The contents of validated cryptographic module used at TOE are as follows.

Item	Specification
Module Name	COLib V1.1.0
Certification Number	CM-150-2024.2
Developer	Eglobal system
Issue Date	2020-05-08
Expiration Date	2024-02-20

Table 1. Validated cryptographic module

1.3.4. Non-TOE Hardware/ Software

The hardware/software lists of non-TOE under TOE operational environment are as follows.

Item	Minimum operation specification				
CubeOne Server (Plug-In)	CPU	POWER5 2.0Ghz above	sparcv9 2848 MHz above	Intel(R) Itanium 2 1.6 GHz above	Intel Dual Core 1.8GHz above
	Memory	4GB above			
	HDD	At least 200MB of space required to install TOE			
	NIC	10/100/1000 X 1Port above			
	OS	AIX 7.1 64bit	SunOS 5.11 64bit	HP-UX 11.31 64bit	CentOS 7.8.2003 64bit (kernel 3.10.0-1127.el7.x86_64)
	DBMS	Tibero 6 DB2 10.5	Tibero 6		
CubeOne Server (API)	CPU	POWER5 2.0Ghz above	sparcv9 2848 MHz above	Intel(R) Itanium 2 1.6 GHz above	
	Memory	4GB above			
	HDD	At least 200MB of space required to install TOE			
	NIC	10/100/1000 X 1Port above			
	OS	AIX 7.1 64bit	SunOS 5.11 64bit	HP-UX 11.31 64bit	
CubeOne Manager	CPU	Intel Core 2 Duo 2.40GHz above			
	Memory	4GB above			
	HDD	At least 200MB of space required to install TOE			
	NIC	10/100/1000 X 1Port above			
	OS	Windows 2012 R2 Standard 64bit			
	essential S/W	- IBM Data Server Client Packages Version 10.5 - Tibero6 ODBC client - MS Visual C++ 2010 Redistributable Package (x86)			
CubeOne Security Server / Beacon	CPU	Intel Core 2 Duo 2.26 GHz above			
	Memory	4GB above			

Item	Minimum operation specification	
	HDD	At least 200MB of space required to install TOE
	NIC	10/100/1000 X 1Port above
	OS	CentOS 7.8.2003 64bit (kernel 3.10.0-1127.el7.x86_64)
	essential S/W	- MariaDB 10.5.5 - Apache tomcat 8.5.57
Authorized log administrator	Web browser	Chrome V 70

Table 2. Minimum operation specification of hardware

The description of essential Software is as follows.

Item	Software	Specification
CubeOne Manager	IBM Data Server Client Packages Version 10.5 , Tibero6 ODBC client	Application program to connect ORACLE DB at CubeOne Manager
	MS Visual C++ 2010 Redistributable Package (x86)	Visual C++ runtime package to use MySQL Connector/ODBC.
CubeOne Security Server	MariaDB 10.5.5	Database used for the audit repository of TOE
CubeOne Beacon	Apache tomcat 8.5.57	WAS server for CubeOne Beacon

1.4. TOE description

According to operational environment of CubeOne Server, the TOE can be classified into two types: plug-in and API type. It means that type is determined by what kinds of subject perform encryption/decryption. If subject is DB, type is plug-in. If subject is Application server, type is API. The TOE provides the functions that the authorized administrator can create policy and distribute it through CubeOne Manager and then CubeOne Server can perform encryption/decryption according to policy. The histories of encryption or decryption and audit log data of TOE are sent to CubeOne Security Server. The authorized log administrator can review TOE through CubeOne Beacon.

1.4.1. Physical Scope

The physical scope of TOE is CubeOne Manager, CubeOne Server, CubeOne Security Server, CubeOne Beacon and those are inside CD. The validated cryptographic module is included in TOE. The physical scope of the TOE also includes 'Operation Manual' and 'Installation Manual' that are distributed to end users in electronic document (CD) form to ensure that they operate the TOE in a safe manner. Required stuffs such as Hardware, OS, DBMS, and Oracle Client, which are needed to operate TOE in the physical scope, are excluded from the physical scope.
The physical scope of TOE is graphically represented as follows.

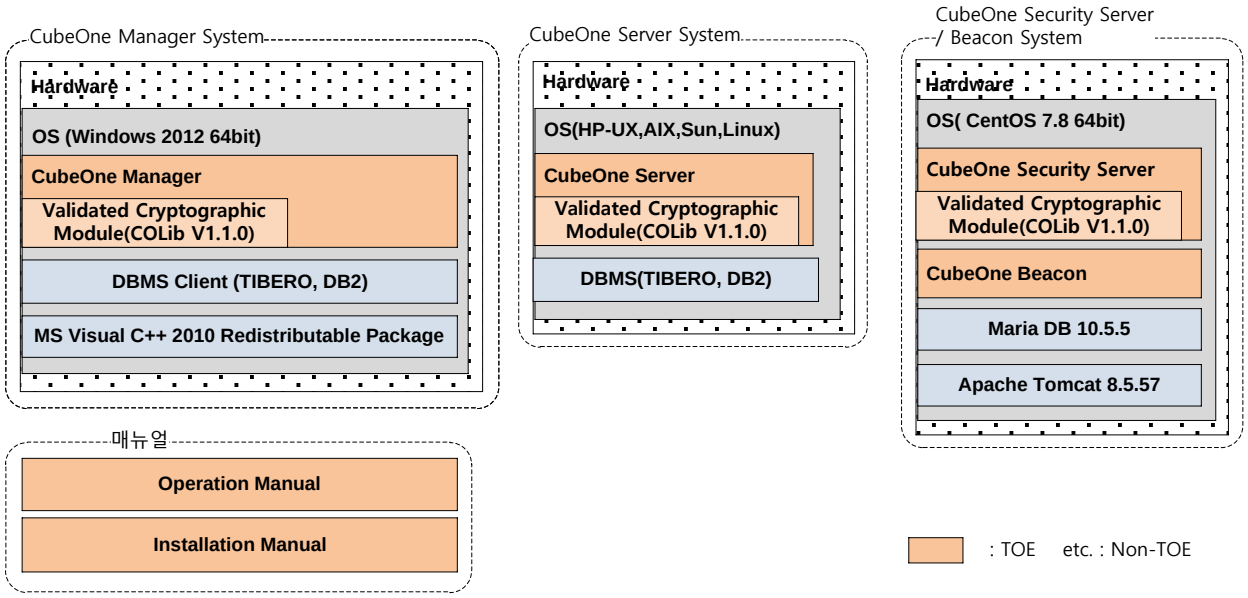


Figure 3. Physical scope of TOE

The product box is comprised of TOE-related materials. The product box is labeled and delivered after packing the product CD case, manuals, and certification into the product box. The components are as follows.

Item		Content	status
TOE Name		CubeOne V2.5 SP1	CD
TOE Components	CubeOne Manager	- CubeOne_Manager_V2.5.00.01_SP1 : CubeOne_Manager_V2.5.00.01_SP1.exe	Included in CD
	CubeOne Server	[Plug-In] - CubeOne_Server_V2.5.00.01_SP1_A64_7.1_TI6 : CubeOne_Server_V2.5.00.01_SP1_A64_7.1_TI6.tar - CubeOne_Server_V2.5.00.01_SP1_A64_7.1_DB10.5 : CubeOne_Server_V2.5.00.01_SP1_A64_7.1_DB10.5.tar - CubeOne_Server_V2.5.00.01_SP1_S64_5.11_TI6 : CubeOne_Server_V2.5.00.01_SP1_S64_5.11_TI6.tar - CubeOne_Server_V2.5.00.01_SP1_H64_B.11.31_TI6 : CubeOne_Server_V2.5.00.01_SP1_H64_B.11.31_TI6.tar - CubeOne_Server_V2.5.00.01_SP1_L64_3.10_TI6 : CubeOne_Server_V2.5.00.01_SP1_L64_3.10_TI6.tar [API] - CubeOne_Server_V2.5.00.01_SP1_A64_7.1_API : CubeOne_Server_V2.5.00.01_SP1_A64_7.1_API.tar - CubeOne_Server_V2.5.00.01_SP1_S64_5.11_API : CubeOne_Server_V2.5.00.01_SP1_S64_5.11_API.tar - CubeOne_Server_V2.5.00.01_SP1_H64_B.11.31_API : CubeOne_Server_V2.5.00.01_SP1_H64_B.11.31_API.tar	Included in CD
	CubeOne Security Server	- CubeOne_SServer_V2.5.00.01_SP1_L64_3.10_MA : CubeOne_SServer_V2.5.00.01_SP1_L64_3.10_MA.tar	Included in CD
	CubeOne Beacon	- CubeOne_Beacon_V2.5.00.01_SP1 : CubeOne_Beacon_V2.5.00.01_SP1.tar.gz	Included in CD
Manuals	Operation Manual	CubeOne_OPE_V2.5.1.1_SP1.pdf	print, Included in CD
	Installation Manual	CubeOne_PRE_V2.5.1.2_SP1.pdf	print, Included in CD
Certificate		Certificate of conformance	print

1.4.2. Logical Scope

Below represent security function of TOE.

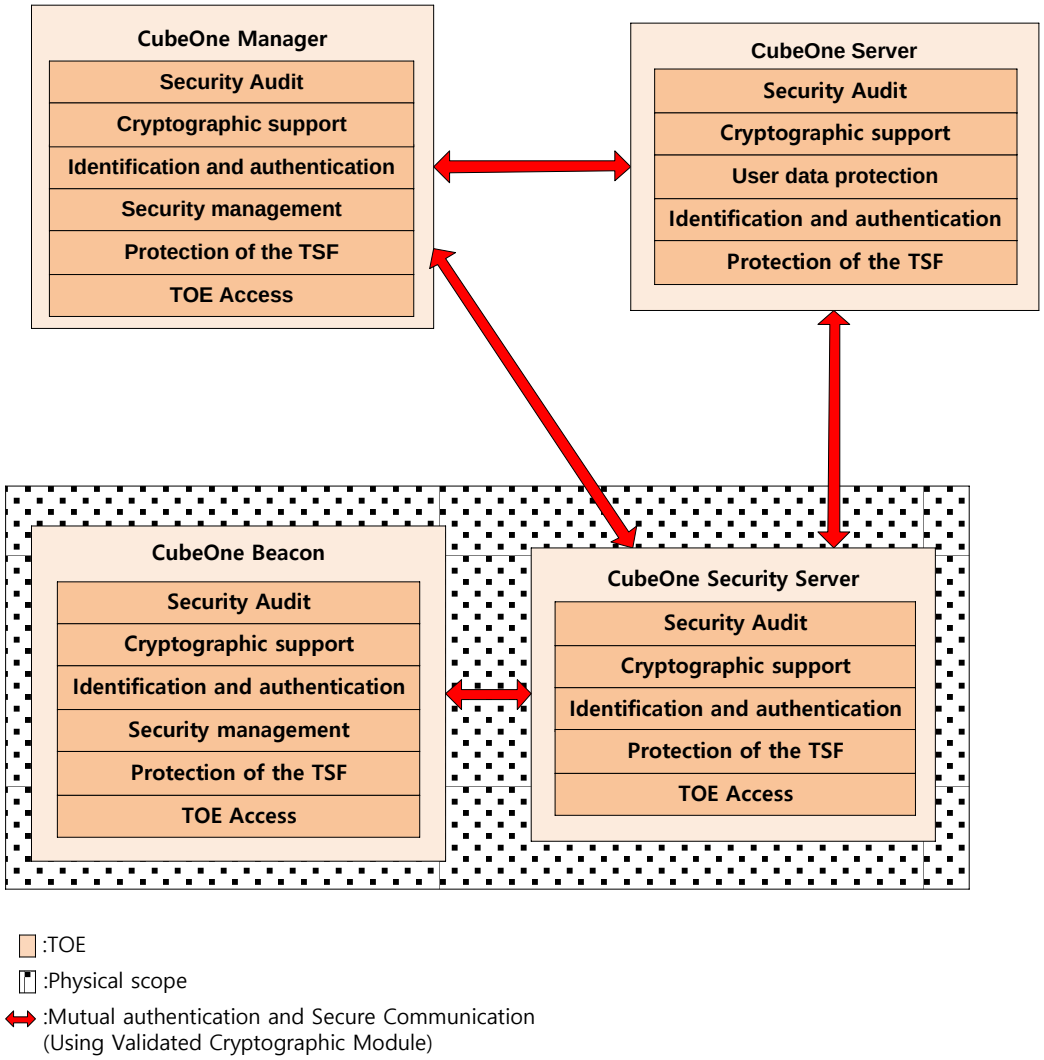


Figure 4. Logical scope of TOE

1.4.2.1. CubeOne Manager

[Security audit]

The TOE generates audit records of the auditable events like cryptographic support, identification and authentication, etc. and the audit record include the date of the event, the type of event, the identity and the outcome of the event. The audit data generated by CubeOne Manager is stored in PC on which CubeOne Manager installed.

The TOE provides a pop-up alarm to the authorized policy administrator when detecting a potential security violation like authentication failure event, integrity violation of auditable events.

An authorized policy administrator can review all audit data from audit records and selectively review audit data according to criteria that has a logical relationship

If the audit trail exceeds 80% of the audit repository capacity, notify the authorized policy administrator by pop-up. The audited event will be ignored if the audit trail is saturated.

[Cryptographic support]

The key for user data encryption and TSF data encryption is generated by random number generator of validated cryptographic module.

The authorized policy administrator generates the user data encryption key through CubeOne Manager and distributes it to CubeOne Server.

The cryptographic operation to encrypt/decrypt TSF data uses ARIA algorithm of validated cryptographic module and its key length is 256 bit. After using the TSF data encryption key, the memory area of key is overwritten by '0'.

[Identification and authentication]

The TOE provides the identification and authentication method based on their ID and password and passwords entered are masked so that they cannot be seen on the screen ("*"). The reason for their failure is not provided. And it provides the method that if five consecutive failed certifications occur, the authentication function is prevented for five minutes.

When creating a password, it must be combined with English letters/special characters/numeric characters, and the password length must be between 9 and 30 characters.

CubeOne Manager performs mutual authentication by using the public key cipher and digital signature method of validated cryptographic module before communication with CubeOne Server, CubeOne Security Server.

[Security Management]

The security function provided by TOE and ability to manage TSF data is performed only for authorized policy administrator.
The ID and password for authentication of CubeOne Manager is registered during installation.
In TOE, users are divided into policy administrator who can set up security policies and log administrators who can review security alerts and audit data.
The policy administrator connects to the CubeOne Manager to perform security management while the log administrator connects to the CubeOne Beacon to perform security management.

[Protection of the TSF]
When sending TSF data between TOE components, TSF data is protected from exposure and change by using hash function and block cipher (ARIA-256) of validated cryptographic module.
The TSF data such as encryption key and TOE setting are stored by encrypting it through DEK and then DEK is stored by encrypting it through KEK.
The TOE provides the self-test and integrity verification functions of TSF execution code and TSF data.

[TOE Access]
CubeOne Manager limits sessions that can be accessed at the same time to a maximum of one.
CubeOne Manger locks the session after 10 minutes of administrator inactivity, and security functions can be performed only after administrator re-authentication.

1.4.2.2. CubeOne Server

[Security audit]
The TOE generates audit records of the auditable events like cryptographic support, identification and authentication, etc. and the audit record consist of the date of the event, the type of event, the identity and the outcome of the event. The audit data generated by CubeOne Server is sent to CubeOne Security Server.

[Cryptographic support]
The TSF data encryption key is generated by random number generator of validated cryptographic module. The cryptographic operation to encrypt/decrypt TSF data uses ARIA algorithm of validated cryptographic module and its key length is 256 bit.
The algorithms for user data encryption use only the block cipher and hash function of validated cryptographic module. The ARIA and SEED algorithm is used for block cipher, SHA-256/384/512 for hash function. And ARIA uses 128/192/256 bit key length, SEED uses only 128 bit key length.

After using the TSF data encryption key and user data encryption key, the memory area of key is overwritten by '0'.

[Protection of user data]

The TOE provides the functions of encryption/decryption by column when encrypt/decrypt user data, and to ensure that the previous information is not available, deletes the original table after encryption and saves only the encryption table.

[Identification and authentication]

CubeOne Server performs mutual authentication by using the public key cipher and digital signature method of validated cryptographic module before communication with CubeOne Manager, CubeOne Security Server.

[Protection of the TSF]

When sending TSF data between TOE components, TSF data is protected from exposure and change by using hash function and block cipher (ARIA-256) of validated cryptographic module.

The TSF data such as encryption key and TOE setting are stored by encrypting it through DEK and then DEK is stored by encrypting it through KEK.

The TOE provides the self-test and integrity verification functions of TSF execution code and TSF data.

1.4.2.3. CubeOne Security Server

[Security audit]

The TOE provide audit records of the auditable events and the audit record consist of the date of the event, the type of event, the identity and the outcome of the event. If the audit trail is saturated, the oldest audit record is overwritten.

The audit record generated by CubeOne Security Server, CubeOne Server, and CubeOne Beacon is stored in DBMS of server which CubeOne Security Server is installed.

[Cryptographic support]

The TSF data encryption key is generated by random number generator of validated cryptographic module. The cryptographic operation to encrypt/decrypt TSF data uses ARIA algorithm of validated cryptographic module and its key length is 256 bit. After using the TSF data encryption key, the memory area of key is overwritten by '0'.

[Identification and authentication]

CubeOne Security Server performs mutual authentication by using the public key cipher and digital signature method of validated cryptographic module before communication with CubeOne Server, CubeOne Manager. In case of an authentication attempt in the CubeOne Manager, the session ID is unique to each session to prevent reuse of the authentication attempts.

.

[Protection of the TSF]

When sending TSF data between TOE components, TSF data is protected from exposure and change by using hash function and block cipher (ARIA-256) of validated cryptographic module.

The TSF data such as encryption key and TOE setting are stored by encrypting it through DEK and then DEK is stored by encrypting it through KEK.

The TOE provides the self-test and integrity verification functions of TSF execution code and TSF data.

.

[TOE Access]

The TOE limits sessions with simultaneous access to the CubeOne Security Server from the CubeOne Manager to a maximum of one. The connection from the CubeOne Manager is blocked based on the connection IP.

1.4.2.4. CubeOne Beacon

[Security audit]

The TOE provides the real-time warning screen to the authorized log administrator when detecting a potential security violation like authentication failure event, integrity violation event.

An authorized log administrator can review all audit data stored in DBMS through CubeOne Beacon and perform the selectable audit review according to logical audit criteria.

If the audit trail exceeds 80% of the audit repository capacity, notify the authorized log administrator by real-time warning screen.

[Cryptographic support]

The TSF data encryption key is generated by random number generator of validated cryptographic module. The cryptographic operation to encrypt/decrypt TSF data uses ARIA algorithm of validated cryptographic module and its key length is 256 bit.

After using the TSF data encryption key, the memory area of key is overwritten by '0'.

[Identification and authentication]

The TOE provides the identification and authentication method based on their ID and password. The passwords entered are masked so that they cannot be seen on the screen ("*"). The reason for their

failure is not provided. And it provides the method that if five consecutive failed certifications occur, the authentication function is prevented for five minutes.
When creating a password, it must be combined with English letters/special characters/numeric characters, and the password length must be between 9 and 30 characters.
CubeOne Beacon performs mutual authentication by using the public key cipher and digital signature method of validated cryptographic module before communication with CubeOne Security Server.

[Security Management]

The ID and password for authentication of CubeOne Beacon is registered during installation. The log administrator connects to the CubeOne Beacon and can perform the security management of IP setting for connection, change of password.

[Protection of the TSF]

When sending TSF data between TOE components, TSF data is protected from exposure and change by using hash function and block cipher (ARIA-256) of validated cryptographic module.
The TSF data such as encryption key and TOE setting are stored by encrypting it through DEK and then DEK is stored by encrypting it through KEK.
The TOE provides the self-test and integrity verification functions of TSF execution code and TSF data.

[TOE Access]

CubeOne Beacon limits sessions that can be accessed at the same time to a maximum of 3.
CubeOne Beacon terminates the session after 10 minutes of administrator inactivity, and security functions can be performed after administrator re-authentication. The connection from the CubeOne Beacon is controlled based on the connection IP.

1.5. Conventions

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

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

Operation	Content
Iteration	Iteration is used when a component is repeated with varying operations. The result of iteration is marked with an iteration number in parenthesis following the component identifier, i.e., denoted as (iteration No.).
Assignment	This is used to assign specific values to unspecified parameters (e.g., password length). The result of assignment is indicated in square brackets like [assignment_value].
Selection	This is used to select one or more options provided by the CC in stating a requirement. The result of selection is shown as *underlined and italicized.*
Refinement	This is used to add details and thus further restrict a requirement. The result of refinement is shown in **bold text**.

1.6. Terms and definitions

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

Terms	Definition
CubeOne	Trademark of cryptographic product made by eGlobal Systems Co. ltd.
CubeOne Manager	Security management part of CubeOne. It provides GUI Interface for authorized administrator.
CubeOne Server	Cryptographic processing part of CubeOne. It is installed at server where need encryption/decryption with access control.
CubeOne Security Server	This takes charge of storing TSF data, audit log, cryptographic policy of CubeOne.
CubeOne Beacon	Security monitoring part of CubeOne. The administrator can monitor TOE through it.
Private Key	A cryptographic key which is used in an asymmetric cryptographic algorithm and is uniquely associated with an entity (the subject using the private key), not to be disclosed
Object	Passive entity in the TOE containing or receiving information and on which subjects perform operations
Approved mode of operation	The mode of cryptographic module using approved cryptographic algorithm
Approved cryptographic algorithm	A cryptographic algorithm selected by Korea Cryptographic Module Validation Authority for block cipher, secure hash algorithm, message authentication code, random bit generation, key agreement, public key cipher, digital signatures cryptographic algorithms considering safety, reliability and interoperability
Attack potential	Measure of the effort to be expended in attacking a TOE expressed as an attacker's expertise, resources and motivation
Public Key	A cryptographic key which is used in an asymmetric cryptographic algorithm and is associated with an unique entity (the subject using the public key), it can be disclosed
Public Key(asymmetric) cryptographic algorithm	A cryptographic algorithm that uses a pair of public and private keys
Management access	The access to the TOE by using the HTTPS, SSH, TLS, etc. to manage the TOE by administrator, remotely

Terms	Definition
Symmetric cryptographic technique	Encryption scheme that uses the same secret key in mode of encryption and decryption, also known as secret key cryptographic technique
Database (or DB)	A set of data that is compiled according to a certain structure in order to receive, save, and provide data in response to the demand of multiple users to support multiple application duties at the same time. The database related to encryption by column, which is required by this ST, refers to the relational database.
Data Encryption Key (DEK)	Key that encrypts and decrypts the data
Iteration	Use of the same component to express two or more distinct requirements
Security Function Policy (SFP)	A Set of rules that describes the specific security action performed by TSF (TOE security functionality) and describe them as SFR (security function requirement)
Security Target (ST)	Implementation-dependent statement of security needs for a specific identified TOE
Security attribute	The characteristics of the subject used to define the SFR, user (including the external IT product), object, information, session and/or resources. These values are used to perform the SFR
Security Token	Hardware device that implements key generation and digital signature generation inside the device to save/store confidential information safely
Protection Profile (PP)	Implementation-independent statement of security needs for a TOE type
Decryption	The act that restoring the cipher text into the plaintext using the decryption key
Secret Key	A cryptographic key which is used in an symmetric cryptographic algorithm and is uniquely associated with one or several entity, not to be disclosed
User	Refer to "External entity"
User Data	Data for the user, that does not affect the operation of the TSF
Selection	Specification of one or more items from a list in a component

Terms	Definition
Identity	Representation uniquely identifying entities (e.g. user, process or disk) within the context of the TOE
Encryption	The act that converts the plaintext into the cipher text using the encryption key
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
Threat 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
Self-test	Pre-operational or conditional test executed by the cryptographic module
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

Terms	Definition
Column	A set of data values of a particular simple type, one for each row of the table in a relational database
Component	Smallest selectable set of elements on which requirements may be based
Class	Set of CC families that share a common focus
Key Encryption Key (KEK)	Key that encrypts and decrypts another cryptographic key
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
Critical Security Parameters (CSP)	Information related to security that can erode the security of the encryption module if exposed or changed (e.g., verification data such as secret key/private key, password, or Personal Identification Number).
Application Server	The application server defined in this ST refers to the server that installs and operates the application, which is developed to provide a certain application service by the organization that operates the TOE. The pertinent application reads the user data from the DB, which is located in the database server, by the request of the application service user, or sends the user data to be stored in the DB to the database server.
Database Server	The database server defined in this ST refer to the server in which the DBMS managing the protected DB is installed in the organization that operates the TOE
DBMS (Database Management System)	A software system composed to configure and apply the database. The DBMS related to encryption by column, which is required by this ST, refers to the database management system based on the relational database model.
SSL	This is a security protocol proposed by Netscape to ensure

Terms	Definition
(Secure Sockets Layer)	confidentiality, integrity and security over a computer network
TOE Security Functionality (TSF)	Set of software, firmware and/or hardware possibly accompanied by guidance
TSF Data	Data for the operation of the TOE upon which the enforcement of the SFR relies
ITEM	It is used in the actual encryption function and contains the contents related to the policy including the encryption key of user data.

1.7. Security Target Contents

Chapter 1 introduces to the Security Target, providing Security Target and TOE references, TOE overview, TOE description and terms and definitions.
Chapter 2 provides the conformance claims to the CC, PP and package; and describes the claim's conformance rationale and PP conformance statement.
Chapter 3 describes the security objectives for the operational environment.
Chapter 4 defines the extended components for the database encryption.
Chapter 5 describes the security functional and assurance requirements. If required, Application notes are provided to clarify the meaning of requirements and provide an explanation of detailed guidelines to the ST author for correct operations.
Chapter 6 describes the security functions and warranty requirements of TOE that satisfy the security requirements in the TOE summary statement.

2. Conformance claim

2.1. CC conformance claim

CC		Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5 - Common Criteria for Information Technology Security Evaluation. Part 1: Introduction and General Model, Version 3.1, Revision 5 (CCMB-2017-04-001, April, 2017) - Common Criteria for Information Technology Security Evaluation. Part 2: Security Functional Components, Version 3.1, Revision 5 (CCMB-2017-04-002, April, 2017) - Common Criteria for Information Technology Security Evaluation. Part 3: Security Assurance Components, Version 3.1, Revision 5 (CCMB-2017-04-003, April, 2017)
Conformance claim	**Part 2 Security functional components**	Extended: FCS_RBG.1, 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 clam

This Protection Profile conform 'Korean National Protection Profile for Database Encryption V.1.0'.

Item	Content
Title	Korean National Protection Profile for Database Encryption
Version	V1.1
Evaluation Assurance Level	EAL1+(ATE_FUN.1)
Issue Date	2019.12.11
Certification Number	KECS-PP-0820a-2017
Conformance status	Strict PP conformance

2.3. Package conformance claim

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

2.4. Conformance claim rationale

This ST comply with 'strict PP conformance' through conformances of TOE type, security objectives for the operational environment, security requirement which is required by 'Korean National Protection Profile for Database Encryption V1.0' - hereinafter referred to as "DBEnc-PP".

Item	ST	PP	Rationale
TOE type	DB encryption product	The same as DBEnc-PP	The same as DBEnc-PP
Security objectives for the operational environment	OE.PHYSICAL_CONTROL	The same as DBEnc-PP	The same as DBEnc-PP
	OE.TRUSTED_ADMIN		
	OE.SECURE_DEVELOPMENT		
	OE.LOG_BACKUP		
	OE.OPERATION_SYSTEM_REINFORCEMENT		
	OE.SECURE_DBMS	Add	The same as DBEnc-PP - added according to the Application notes of FAU_STG.1 which is the optional SFR
	OE.TIMESTAMP	Add	The same as DBEnc-PP - added according to the Application notes of FAU_STM.1 which is the optional SFR
	OE.SECURE_CHANNEL	Add	The same as DBEnc-PP - added according to the Application notes of FAU_TRP.1 which is the optional SFR
Security requirement	FAU_ARP.1	FAU_ARP.1	The same as DBEnc-PP.
	FAU_GEN.1	FAU_GEN.1	The same as DBEnc-PP
	FAU_SAA.1	FAU_SAA.1	The same as DBEnc-PP

Item	ST	PP	Rationale
	FAU_SAR.1	FAU_SAR.1	The same as DBEnc-PP
	FAU_SAR.3	FAU_SAR.3	The same as DBEnc-PP
	FAU_STG.3	FAU_STG.3	The same as DBEnc-PP
	FAU_STG.4(1)	FAU_STG.4	The same as DBEnc-PP
	FAU_STG.4(2)	FAU_STG.4	The same as DBEnc-PP
	FCS_CKM.1(1)	FCS_CKM.1(1)	The same as DBEnc-PP
	FCS_CKM.1(2)	FCS_CKM.1(2)	The same as DBEnc-PP
	FCS_CKM.2	FCS_CKM.2	The same as DBEnc-PP
	FCS_CKM.4	FCS_CKM.4	The same as DBEnc-PP
	FCS_COP.1(1)	FCS_COP.1(1)	The same as DBEnc-PP
	FCS_COP.1(2)	FCS_COP.1(2)	The same as DBEnc-PP
	FCS_RBG.1(Extended)	FCS_RBG.1(Extended)	The same as DBEnc-PP
	FDP_UDE.1(Extended)	FDP_UDE.1(Extended)	The same as DBEnc-PP
	FDP_RIP.1	FDP_RIP.1	The same as DBEnc-PP
	FIA_AFL.1	FIA_AFL.1	The same as DBEnc-PP
	FIA_IMA.1(Extended)	FIA_IMA.1(Extended)	The same as DBEnc-PP
	FIA_SOS.1	FIA_SOS.1	The same as DBEnc-PP
	FIA_UAU.1	FIA_UAU.1	The same as DBEnc-PP
	FIA_UAU.2	FIA_UAU.1	The same as DBEnc-PP - Use FIA_UAU.2 in hierarchical relationships according to Application notes of FIA_UAU.1
	FIA_UAU.4	FIA_UAU.4	The same as DBEnc-PP
	FIA_UAU.7	FIA_UAU.7	The same as DBEnc-PP
	FIA_UID.2	FIA_UID.1	The same as DBEnc-PP - Use FIA_UID.2 in hierarchical relationships according to Application notes of FIA_UID.1

Item	ST	PP	Rationale
	FMT_MOF.1	FMT_MOF.1	The same as DBEnc-PP
	FMT_MTD.1	FMT_MTD.1	The same as DBEnc-PP
	FMT_PWD.1(Extended)	FMT_PWD.1(Extended)	The same as DBEnc-PP
	FMT_SMF.1	FMT_SMF.1	The same as DBEnc-PP
	FMT_SMR.1	FMT_SMR.1	The same as DBEnc-PP
	FPT_TST.1	FPT_TST.1	The same as DBEnc-PP
	FPT_ITT.1(Extended)	FPT_ITT.1(Extended)	The same as DBEnc-PP
	FPT_PST.1	FPT_PST.1	The same as DBEnc-PP
	FTA_MCS.2	FTA_MCS.2	The same as DBEnc-PP
	FTA_SSL.5(Extended)	FTA_SSL.5(Extended)	The same as DBEnc-PP
	FTA_TSE.1	FTA_TSE.1	The same as DBEnc-PP

3. Security objectives

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

3.1. Security objectives for the operational environment

Item	Content
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.TIMESTAMP	The TOE accurately records incidents related to security by receiving reliable time stamps provided by the TOE operating environment.
OE.SECURE_DBMS	DBMS that saves the TSF data and audit data is operated in a physically safe environment.
OE.SECURE_CHANNEL	All information that is sent when an authorized log administrator connect to the Web server through the Web browser shall be protected through a secure channel.

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

4.2. Identification and authentication

4.2.1. TOE Internal mutual authentication

Family Behaviour	This family defines requirements for providing mutual authentication between TOE components in the process of user identification and authentication.	
Component leveling	FIA_IMA TOE Internal mutual authentication — 1	
	FIA_IMA.1	TOE Internal mutual authentication requires that the TSF provides mutual authentication function between TOE components in the process of user identification and authentication.
Management	FIA_IMA.1	There are no management activities foreseen.
Audit	FIA_IMA.1	The following actions are recommended to record if FAU_GEN Security audit data generation family is included in the PP/ST: a) Minimal: Success and failure of mutual authentication 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 User data encryption — 1	
	FDP_UDE.1	User data encryption requires confidentiality of user data.
Management	FDP_UDE.1	The following actions could be considered for the management functions in FMT: a) Management of user data encryption/decryption rules
Audit	FDP_UDE.1	The following actions are recommended to record if FAU_GEN Security audit data generation is included in the PP/ST: a) Minimal : Success and failure of user data encryption/decryption

4.3.1.1. FDP_UDE.1 User data encryption

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

4.4. Security Management

4.4.1. ID and password

Family Behaviour	This family defines the capability that is required to control ID and password management used in the TOE and set or modifies ID and/or password by authorized users.	
Component leveling	FMT_PWD ID and password — 1	
	FMT_PWD.1	ID and password management, requires that the TSF provides the management function of ID and password.
Management	FMT_PWD.1	The following actions could be considered for the management functions in FMT: a) Management of ID and password configuration rules.
Audit	FMT_PWD.1	The following actions are recommended to record if FAU_GEN Security audit data generation is included in the PP/ST: a) Minimal: All changes of the password.

4.4.1.1. FMT_PWD.1 Management of ID and password

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

	password when the authorized administrator accesses for the first time, changing the password when the authorized administrator accesses for the first time].

4.5. Protection of the TSF

4.5.1. Protection of stored TSF data

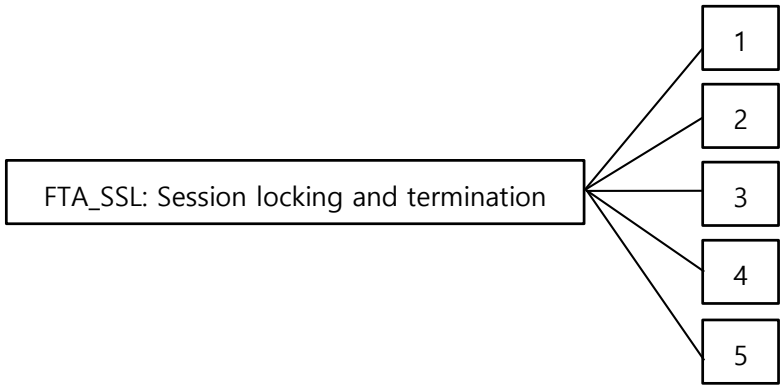
Family Behaviour	This family defines rules to protect TSF data stored within containers controlled by the TSF from the unauthorized modification or disclosure.	
Component leveling	FPT_PST Protection of stored TSF data — 1	
	FPT_PST.1	Basic protection of stored TSF data requires the protection of TSF data stored in containers controlled by the TSF.
Management	FTP_PST.1	There are no management activities foreseen.
Audit	FTP_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

<table>
<tr><td>Family Behaviour</td><td colspan="2">This family defines requirements for the TSF to provide the capability for TSF-initiated and user-initiated locking, unlocking, and termination of interactive sessions.</td></tr>
<tr><td rowspan="2">Component leveling</td><td colspan="2"> 
</td></tr>
<tr><td>FTA_SSL.5</td><td>The management of TSF-initiated sessions provides requirements that the TSF locks or terminates the session after a specified time interval of user inactivity.</td></tr>
<tr><td>Management</td><td>FTA_SSL.5</td><td>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</td></tr>
<tr><td>Audit</td><td>FTA_SSL.5</td><td>The following action should be auditable if FAU_GEN Security audit data generation is included in the PP/ST.
a) Minimal: Termination of an interactive session by the user.
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</td></tr>
</table>

4.6.1.1. FTA_SSL.5 Management of TSF-initiated sessions

Hierarchical to	No other components.
Dependencies	FIA_UAU.1 authentication
FTA_SSL.5.1	The TSF shall [*lock the session/or re-authenticate the user before unlocking the session*] an interactive session [after time interval of **administrator** inactivity].

5. Security requirements

The security requirements specify security functional requirements and assurance requirements that must be satisfied by the TOE that claims conformance to this ST.
The security functional requirements included in this ST are derived from CC Part 2 and Chapter 4 Extended Components Definition.

5.1. Security functional requirements

The TOE that claims conformance to this ST must meet the following 'SFRs'.

Security functional class	Security functional component	
FAU	FAU_ARP.1	Security alarms
	FAU_GEN.1	Audit data generation
	FAU_SAA.1	Potential violation analysis
	FAU_SAR.1	Audit review
	FAU_SAR.3	Selectable audit review
	FAU_STG.3	Protected audit trail storage
	FAU_STG.4(1)	Action in case of possible audit data loss
	FAU_STG.4(2)	Action in case of possible audit data loss
FCS	FCS_CKM.1(1)	Prevention of audit data loss
	FCS_CKM.1(2)	Cryptographic key generation (TSF data encryption)
	FCS_CKM.2	Cryptographic key distribution
	FCS_CKM.4	Cryptographic key destruction
	FCS_COP.1(1)	Cryptographic operation (User data encryption)
	FCS_COP.1.(2)	Cryptographic operation (TSF data encryption)
	FCS_RBG.1(Extended)	Random bit generation
FDP	FDP_UDE.1(Extended)	User data encryption
	FDP_RIP.1	Subset residual information protection
FIA	FIA_AFL.1	Authentication failure handling
	FIA_IMA.1(Extended)	TOE Internal mutual authentication
	FIA_SOS.1	Verification of secrets

Security functional class	Security functional component	
	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
	FTP_PST.1(Extended)	Basic protection of stored TSF data
	FTP_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 3. Summary of 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 [Expose warning screen in Beacon, Notify Manager as Popup] 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 4],** *no other components*].
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 4],** *no other components*].

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	

Security functional component	Auditable event	Additional audit record
FIA_UAU.4	Attempts to reuse authentication data	
FIA_UID.2	All use of the user identification mechanism, including the user 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	

Table 4. Auditable 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 selftest failure event of validated cryptographic module among auditable events of FPT_TST.1, [audit event for response Behaviour when threshold is exceeded among the auditable events of FAU_STG.3, audit event for response actions if audit arrest fails among the auditable event of FAU_STG.4.]] known to indicate a potential

	security violation b) [no other rules]

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 [Table 5. Selectable audit review methods] of audit data based on [criteria with following logical relations].

Item	Selection/ordering		Logical relation
Manager	query		AND of the entered value among the items below Total, Server, Database Name, Workgroup, In Workgroup, Item
Beacon	Service error	query	AND of the entered value among the items below - server name, date(start~end), level (inform, warning, critical, fatal)
		ordering	ascending/ descending order based on one of the items below - no., date, server name, server type, detail description, level
	Detection of massive decryption	query	AND of the entered value among the items below - server name, date(start~end), level (inform, warning, critical, fatal), user name, IP, program name
		ordering	ascending/ descending order based on one of the items below - no., date, server name CubeOne type, username, table, decryption/encryption count, IP, program name
	Audit log	query	AND of the entered value among the items below - server name, date(start~end), level (inform, warning, critical,

Item	Selection/ordering		Logical relation
			fatal), user name, IP, program name
		ordering	ascending/ descending order based on one of the items below - no., date, server name CubeOne type, username, table, column, sql statement, item, IP, program name, detail of audit

Table 5. Selectable audit review methods

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

Hierarchical to	No other components.
Dependencies	FAU_GEN.1 Protected audit trail storage
FAU_STG.3.1	The TSF shall [**Warnings on pop-ups and Beacon screens to the authorized policy/log administrator,** [no other rule]] if the audit trail exceeds [when reached threshold (80%) of audit storage].

5.1.1.7. FAU_STG.4 (1) 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 pop-up message to authorized user with special rights] if the audit trail is full. .

* Application notes: This requirement applies to audit data loss of CubeOne Manager.

5.1.1.8. FAU_STG.4 (2) 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 *overwrite the oldest stored audit records* and [show alert screen on CubeOne Beacon] if the audit trail is full. .

* Application notes: This requirement applies to audit data loss of CubeOne Beacon.

5.1.2. Cryptographic support (FCS)

The password algorithm supported by TOE is as follows and supports only the approved cryptographic algorithm.

Item	Approved algorithm	Detail	Standard criteria
Block cipher	ARIA	Operation mode: CBC, CFB-128 Key Length: 128/192/256 bit	KS X 1213-1 KS X 1213-2
	SEED	Operation mode: CBC, CFB-128 Key Length: 128 bit	TTAS.KO-12.0004/R1 TTAS.KO-12.0025
Hash function	SHA-224 SHA-256 SHA-384 SHA-512		ISO/IEC 10118-3
Random number generator	HASH_DRBG	Hash: SHA-256	TTAK.KO-12.0190
Public key cipher	RSAES	|n|: 2048/3072 bit e: 65537 Hash: SHA-256	ISO/IEC 18033-2
Digital signatures	RSA-PSS	|n|: 2048/3072 bit e: 65537 Hash: SHA-256	ISO/IEC 14888-2
MAC	HMAC	Hash:SHA-256	ISO/IEC 9797-2

Table 6. Approved Cryptographic Algorithm

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 [Random number generator standard (TTAK.KO-12.0190) of "Table 6. appoved Cryptographic Algorithm"] and specified cryptographic key sizes [HASH_DRBG of "Table 6. appoved Cryptographic Algorithm"] that meet the following: [128, 192, 256 bit].

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
FSC_CKM.1.1	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [Key generation algorithm of "Table 7. Cryptographic key generation"] and specified cryptographic key sizes [key length of "Table 7. Cryptographic key generation"] that meet the following: [standard of "Table 7. Cryptographic key generation"]

Item		Standard	Key generation algorithm	Key length	Key Description
Key generation for mutual authentication among TOE's components	private key, public key	ISO/IEC 18033-2	RSAES(SHA-256)	2048bit	Asymmetric key pair for session key encryption
	private key, public key	ISO/IEC 14888-2	RSA-PSS(SHA-256)	2048bit	Asymmetric key pair for digital signature
Key generation for basic protection of internally transmitted TSF data	session Key	TTAK.KO-12.0190	HASH_DRBG(SHA-256)	256bit	Session key using session information as a key to encrypt internal transmission data
Key generation for basic protection of stored TSF data	Drived Key(DK)	TTAK.KO-12.0274	Password Based Key Derivation Functions, HMAC-SHA-2	256bit	Generating a derivation key to be used as the key of KEK through user input
	Master Key(KEK)	TTAK.KO-12.0190	HASH_DRBG(SHA-256)	256bit	Generating a KEK to encrypt DEK
	Secondary Key(DEK)	TTAK.KO-12.0190	HASH_DRBG(SHA-256)	256bit	Generating a DEK to encrypt TSF data

Table 7. Cryptographic key generation

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 [Distribution method of "Table 8. Cryptographic key distribution"] that meets the following [standard of "Table 8. Cryptographic key distribution"]

Item	Standard	Approved algorithm	Distribution method
Key distribution for the user data encryption	KS X 1213-1 KS X 1213-2	ARIA-256(CBC)	block cipher (ARIA) and hash function (SHA256) provided by validated cryptographic module
	ISO/IEC 10118-3	SAH256	
Key distribution for the basic protection of internally transmitted TSF data	ISO/IEC 18033-2	RSAES(2048)	public key cipher(RSAES) provided by validated cryptographic module

Table 8. Cryptographic key distribution

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 [Free memory after overwrite the memory area to '0'] that meets the following: [no other standard].

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

Hierarchical to	No other components.
Dependencies	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1	The TSF shall perform the user data encryption/decryption in accordance with a specified cryptographic algorithm [ARIA, SEED, and SHA-256/384/512 of "Table 6. Approved Cryptographic Algorithm"] and cryptographic key sizes [key length of "Table 6. Approved Cryptographic Algorithm" that meet the following [block cipher and hash function of "Table 6. Approved Cryptographic Algorithm"]

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

Hierarchical to	No other components.
Dependencies	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1	The TSF shall perform [Cryptographic operations of "Table 9. TSF data Cryptographic operation"] in accordance with a specified cryptographic algorithm [algorithm of "Table 9. TSF data Cryptographic operation"] and cryptographic key sizes [key length of "Table 9. TSF data Cryptographic operation"] that meet the following: [standard of "Table 9. TSF data Cryptographic operation"]

Cryptographic operation	Standard	Algorithm	Key length
Mutual authentication among the TOE components	ISO/IEC 18033-2	RSAES(SHA-256)	2048bit
	ISO/IEC 14888-2	RSA-PSS(SHA-256)	2048bit
Basic protection of the internally transmitted TSF data	KS X 1213-1 KS X 1213-2	ARIA CBC 모드	256bit
	ISO/IEC 10118-3	SHA-256	
Basic protection of the stored TSF data	KS X 1213-1 KS X 1213-2	ARIA CBC 모드	256bit

Cryptographic operation	Standard	Algorithm	Key length
	ISO/IEC 9797-2	HMAC(SHA-256)	256bit

Table 9. TSF data Cryptographic operation

5.1.2.7. 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 [TTAK.KO-12.0190]

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, [no method]].

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 (FIA)

5.1.4.1. FIA_AFL.1 Authentication failure handling

Hierarchical to	No other components.

Dependencies	FIA_UAU.1 Timing of authentication
FIA_AFL.1.1	The TSF shall detect when [5] consecutive unsuccessful authentication attempts occur related to [administrator authentication]
FIA_AFL.1.2	When the defined number of unsuccessful authentication attempts has been *met*, the TSF shall [perform identificationm and authentication function inactivation during 5 minute].

5.1.4.2. FIA_IMA.1 Internal mutual authentication (Extended)

Hierarchical to	No other components.
Dependencies	No dependencies.
FIA_IMA.1.1	The TSF shall perform mutual authentication using [using the public key cipher and digital signatures of validated cryptographic module] in accordance with [no standard] between [CubeOne Manager, CubeOne Server, CubeOne Security Server, CubeOne Beacon]

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 [as follows]. [a) Length: min. 9 ~ max. 30 b) English letter, special , number char c) Combination rules - Must contain at least one English letter, special, number character]

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

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 [authentication mechanisms of "Table 10. Single-use authentication mechanisms"].

Item	authentication mechanisms
Policy administrator password authentication	Ensure that session ID is unique for each session
Log administrator password authentication	Ensure that session ID is unique for each session

Table 10. Single-use authentication mechanisms

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 [feedback as following] to the user while the authentication is in progress. [a) Passwords entered are masked so that they cannot be seen on the screen ("*"). - Password for administrator registration, password entered for policy manager/log administrator authentication b) If the identification is fail, do not provide a reason for their failure.]

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 user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

5.1.5. Security management (FMT)

5.1.5.1. FMT_MOF.1 Management of security functions Behaviour

Hierarchical to	No other components.
Dependencies	No dependencies
FMT_MOF.1.1	The TSF shall restrict the ability to ***conduct management actions of*** the functions ["Table 11. List and Action of security functions"] to [authorized policy administrator and authorized log administrator].

Authorized Administrator	**Security function**	**Action**			
		decision	**stop**	**start**	**change**
Authorized policy administrator	Identification and Authentication	O	X	X	X
	Integrity verification	O	X	X	X
	User encryption policy	O	O	O	X
	Item distribution	O	X	O	X
	Audit data review	O	X	X	X
	Password policy	O	X	X	X
Authorized log administrator	Audit data review	O	X	X	X
	Administrator connection IP	O	X	X	X
	Password policy	O	X	X	X

Table 11. List and Action of security functions

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 12. TSF Data list and management ability"] to [authorized policy administrator and authorized log administrator].

(*Reg.: Registration)

Authorized	**TSF data**	**Ability**

Administrator		Query	Change	*Reg.	Delete
Authorized policy administrator	Audit Data	O	X	X	X
	Administrator password	X	O	O	X
	CubeOne Server information	O	O	O	O
	CubeOne operation type	O	O	O	O
	Group information of cryptographic policy	O	O	O	O
	ITEM information for encryption	O	O	O	O
Authorized log administrator	Audit Data	O	X	X	X
	Administrator connection IP	O	O	O	O
	Administrator password	X	O	O	X

Table 12. TSF Data list and management ability

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 [no function] to [nobody].
FMT_PWD.1.2	The TSF shall restrict the ability to manage the ID of [nobody] to [no function].
FMT_PWD.1.3	The TSF shall provide the capability for *setting ID and password when installing*.

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: [a) security functions lists defined in FMT_MOF.1 b) TSF data management lists defined in FMT_MTD.1 c) password management lists defined in FMT_PWD.1]

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 [a) authorized policy administrator b) authorized log 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 (FPT)

5.1.6.1. FPT_ITT.1 Basic internal TSF data transfer protection

Hierarchical to	No other components.
Dependencies	No dependencies
FPT_ITT.1.1	The TSF shall protect the TSF data from *disclosure, modification* 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 [following TSF data] stored in containers controlled by the TSF from the unauthorized *disclosure, modification*. [a) administrator ID/password b) cryptographic key (symmetric key, public key, DEK) c) TOE setting value (security policy, environment setting parameters) d) critical security parameters e) audit data f) user information(DBMS)]

5.1.6.3. FPT_TST.1 TSF testing

Hierarchical to	No other components.
Dependencies	No dependencies
FTP_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 [*CubeOne Server, CubeOne Security Server*].
FTP_TST.1.2	The TSF shall provide **authorized policy administrators** with the capability to verify the integrity of [*TSF data*].
FTP_TST.1.3	The TSF shall provide **authorized policy administrators** with the capability to verify the integrity of [*TSF*].

5.1.7. TOE access (FTA)

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

Hierarchical to	FTA_MCS.1 Basic limitation on multiple concurrent sessions
Dependencies	FIA_UID.1 Timing of identification
FTA_MCS.2.1	The TSF shall restrict the maximum number of concurrent sessions [belonging to the same **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 {what is determined by the ST author} 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) [no rule].
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	TSF shall *lock the session and/or re-authenticate the **policy administrator** before unlocking the session* after a [10 minutes of the **policy 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 policy/log administrator**, based on [Access IP, *None*].

5.2. Security assurance requirements

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

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

Table 13. Security assurance requirements

5.2.1. Security Target evaluation

5.2.1.1. ASE_INT.1 ST introduction

Dependencies	ASE_INT.1	ST introduction
	ASE_ECD.1	Extended components definition
	ASE_REQ.1	Stated security requirements
Developer action	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	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	ASE_CCL.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.1.2. ASE_OBJ.1 Security objectives for the operational environment

Dependencies	No dependencies.	
Developer action	ASE_OBJ.1.1D	The developer shall provide a statement of security objectives.
Content and presentation	ASE_OBJ.1.1C	The statement of security objectives shall describe the security objectives for the operational environment.
Evaluator action	ASE_OBJ.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.1.3. ASE_ECD.1 Extended components definition

Dependencies	No dependencies.	
Developer action	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	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	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.

5.2.1.4. ASE_REQ.1 Stated security requirements

Dependencies	ASE_ECD.1	Extended components definition
Developer action	ASE_REQ.1.1D	The developer shall provide a statement of security requirements
	ASE_REQ.1.2D	The developer shall provide security requirements rationale.
Content and presentation	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	ASE_REQ.1.1.E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

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

Content and presentation	ASE_TSS.1.1C	The TOE summary specification shall describe how the TOE meets each SFR.
Evaluator action	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	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	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	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	AGD_OPE.1.1D	The developer shall provide operational user guidance
Content and	AGD_OPE.1.1C	The operational user guidance shall describe, for each user role,

presentation		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	AGD_OPE.1.7E	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	AGD_PRE.1.1D	The developer shall provide the TOE including its preparative procedures.
Content and presentation	AGD_PRE.1.1C	The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.
	AGD_PRE.1.2C	The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of

		the operational environment in accordance with the security objectives for the operational environment as described in the ST.
Evaluator action	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 Leveling of the TOE

Dependencies	ALC_CMS.1	TOE CM coverage
Developer action	ALC_CMC.1.1D	The developer shall provide the TOE and a reference for the TOE.
Content and presentation	ALC_CMC.1.1C	The TOE shall be labelled with its unique reference.
Evaluator action	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	ALC_CMS.1.1D	The developer shall provide a configuration list for the TOE.
Content and presentation	ALC_CMS1.1C	The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.
	ALC_CMS1.2C	The configuration list shall uniquely identify the configuration items.
Evaluator action	ALC_CMS1.1E	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	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	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	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	ATE_IND.1.1D	The developer shall provide the TOE for testing.
Content and presentation	ATE_IND.1.1C	The TOE shall be suitable for testing
Evaluator action	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	AVA_VAN.1.1D	The developer shall provide the TOE for testing
Content and presentation	AVA_VAN.1.1C	The TOE shall be suitable for testing
Evaluator action	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	Rationale(1)
3	FAU_SAA.1	FAU_GEN.1	2
4	FAU_SAR.1	FAU_GEN.1	2
5	FAU_SAR.3	FAU_SAR.1	4
6	FAU_STG.3	FAU_STG.1	Rationale(2)
7	FAU_STG.4(1)	FAU_STG.1	Rationale(2)
8	FAU_STG.4(2)	FAU_STG.1	Rationale(2)
9	FCS_CKM.1(1)	[FCS_CKM.2 or FCS_COP.1]	11, 13
		FCS_CKM.4	12
10	FCS_CKM.1(2)	[FCS_CKM.2 or FCS_COP.1]	11, 13
		FCS_CKM.4	12
11	FCS_CKM.2	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9
		FCS_CKM.4	12
12	FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9
13	FCS_COP.1(1)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9
		FCS_CKM.4	12
14	FCS_COP.1(2)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9
		FCS_CKM.4	12
15	FCS_RBG.1	-	-
16	FDP_UDE.1	FCS_COP.1	13
17	FDP_RIP.1	-	-
18	FIA_AFL.1	FIA_UAU.1	21
19	FIA_IMA.1	-	-

No	Security functional requirements	Dependency	Reference No.
20	FIA_SOS.1	-	-
21	FIA_UAU.2	FIA_UID.1	24
22	FIA_UAU.4	-	-
23	FIA_UAU.7	FIA_UAU.1	20
24	FIA_UID.2	-	-
25	FMT_MOF.1	FMT_SMF.1	29
26	FMT_MTD.1	FMT_SMR.1	30
27	FMT_PWD.1	FMT_SMF.1	29
		FMT_SMR.1	30
28	FMT_SMF.1	FMT_SMF.1	29
		FMT_SMR.1	30
29	FMT_SMR.1	-	-
		FIA_UID.1	24
30	FPT_ITT.1	-	-
31	FPT_PST.1	-	-
32	FPT_TST.1	-	-
33	FTA_MCS.2	FIA_UID.1	24
34	FTA_SSL.5	FIA_UAU.1	20
35	FTA_TSE.1	-	-

Table 14. Rationale for the dependency of the security functional requirements

- Rationale (1): FAU_GEN.1 has the dependency on FAU_STG.1. However, This ST satisfies the dependent relationship by using the reliable time stamp provided by the OE.TIMESTAMP for security purposes of operation environment.
- Rationale (2): FAU_STG.3 and FAU_STG.4 have the dependency on FAU_STG.1. However, This ST satisfies the dependent relationship by using the trusted audit storage provided by the OE. SECURE_DBMS for security purposes of operation environment. In addition, the policy manager (CubeOne Manager) is supported in the operating environment through OE.TRUSTED_ADMIN to satisfy FAU_STG.1.
- FIA_AFL.1 and FIA_UAU.7 have the dependency on FIA_UAU.1. However FIA_UAU.2 satisfies in hierarchical relationships with FIA_UAU.1

- FIA_UAU.2, FMT_SMR.1 and FTA_MCS.2 have the dependency on FIA_UID.1. However FIA_UID.2 satisfies in hierarchical relationships with FIA_UID.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 represents the overview of security function required by TOE.

6.1. Security audit (FAU)

TOE uses the reliable timestamp provided by the TOE operating environment at the time of the event to ensure that audit data are generated sequentially during the generation of audit data. TOE sends all logs that occur during operation to the CubeOne Security Server for storing audit data. CubeOne Security Server stores the received logs in the DBMS (MariaDB) and can review audit data through CubeOne Beacon.

6.1.1. Potential security violation and security alert

The TOE can detect potential security violations like Table 15.

Security function component	Event of potential security violations
FAU_UAU.2	Authentication failure audit event
FPT_TST.1	Integrity violation audit event and self-tests failure event of validated cryptographic module among auditable events
FAU_STG.3	Audit event of actions taken due to exceeding of a threshold
FAU_STG.4	Audit event of actions taken due to the audit storage failure

Table 15. Potential security violations audit event

TOE generates audit data on such potential violation events, exposes the warning screen to the CubeOne Beacon screen, and notifies the user with a pop-up of the CubeOne Manager.

Satisfied security function component
FAU_SAA.1, FAU_ARP.1

6.1.2. Audit data generation

The TOE component generates an audit data of the events to be audited as defined in "Events to be audited" below. The audit data generated by CubeOne Manager is stored in PC of policy administrator. And the audit data generated by CubeOne Server, CubeOne Security Server, and CubeOne Beacon are stored in the storage of CubeOne Security Server.

Auditable event
Actions taken due to potential security violations
Actions taken due to exceeding of a threshold, the audit storage failure
Success and failure about generation/operation/destruction/distribution of key related to user data encryption
Actions taken due to the reaching of the threshold for the unsuccessful authentication attempts
Success and failure of mutual authentication between TOE components
Success and failure of identification/authentication of administrator for policy and log
Attempts to reuse authentication data
All modifications to the functions in the TSF, the values of TSF data
Execution of the TSF self-tests and the results of the tests
Denial of a new session based on the limitation of multiple concurrent sessions
Locking or termination of interactive session

The audit data generated by TOE shall be recorded as follows.

Information
Date and time, type, identity and the outcome (success or failure) of the event

Satisfied security function component
FAU_GEN.1

6.1.3. Audit review

The audit data can be reviewed through CubeOne Manager and CubeOne Beacon, and only authorized administrators can be interrogated.
It provides the functions of security alert, review, and analysis of security audit generated in TOE.

Authorized policy administrator can review audit data stored encrypted on the administrator's PC via the CubeOne Manager.
An authorized log administrator can review audit data stored in the audit storage (DBMS) through CubeOne Beacon.

The auditable records which administrator can review are as follows.

Item	Selection/ordering		Logical relation
Manager	selection		AND operation of input value which is listed below. Total, Server, Database Name, Workgroup, In Workgroup, Item
Beacon	Service error	selection	AND operation of input value which is listed below. - Server Name, Date (start ~ end), Level (Inform, warning, critical, fatal)
		ordering	Ascending/descending order based on selected one values which is listed below - Serial No. ,Date, Server name, Server type, detail expression, level
	Detection of massive decryption	selection	AND operation of input value which is listed below. - Server Name, Date (start ~ end), Level (Inform, warning, critical, fatal) , Username, IP, Program name
		ordering	Ascending/descending order based on selected one values which is listed below - Serial No., Date, Server name, CubeOne type, Username, Table, Count of encryption/decryption, IP, Program name
	Audit log	selection	AND operation of input value which is listed below. - Server Name, Date (start ~ end), Level (Inform, warning, critical, fatal) , Username, IP, Program name
		ordering	Ascending/descending order based on selected one values which is listed below - Serial No., Date, Server name, CubeOne type, Username, Table, Column, Query statement, IP, Program name, Audit detail

Satisfied security function component
FAU_SAR.1, FAU_SAR.3

6.1.4. Action in case of possible audit data loss and Prevention of audit data loss

If the audit trail exceeds 80% of the audit repository capacity, CubeOne Manager sends alert to policy administrator through pop-up window. If the audit tail storage is saturated, audited event is ignored.

When the CubeOne Security Server is reached at 80% of the audit repository capacity, it exposes a real-time warning screen to the CubeOne Beacon. If the audit tail storage is saturated, new audit data overwrites the oldest one.

Satisfied security function component
FAU_STG.3, FAU_STG.4(1), FAU_STG.4(2)

6.2. Cryptographic support (FCS)

The contents of validated cryptographic module used in TOE are as follows.

Item		Content
Module Name		COLib V1.1.0
Certification Number		CM-150-2024.2
Developer		Eglobal system
Issue Date		2020-05-08
Expiration Date		2024-02-20
Library name	Windows	colib.dll
	AIX	libcolib.so
	Linux	libcolib.so
	HP-UX	libcolib.sl
	Sun	libcolib.so

6.2.1. Cryptographic key generation (User data encryption)

The Cryptographic key used for user data encryption at TOE is generated through CubeOne Manager, the administration tool of TOE, according to user key length. In TOE, encryption keys that are used for user data encryption/decryption created during ITEM creation and are used for cryptographic operation. Block cipher algorithm, encryption key length, and operation mode supported by TOE are as follows.

Item	Approved function	Key length
Block cipher algorithm	ARIA	128/192/256 bit
	SEED	128 bit
	Operation mode	CBC, CFB-128, OFB

The encryption key generation is generated through the random number generator (HASH_DRBG) of validated cryptographic module used by TOE.

Item	Approved function	Remark
Random number generator	HASH_DRBG	Hash: SHA-256

Satisfied security function component
FCS_CKM.1(1), FCS_RBG.1

6.2.2. Cryptographic key generation (TSF data encryption)

The cryptographic keys used for TSF data encryption stored in TOE create KEK and DEK through random number generator of validated cryptographic module. DEK is used for TSF data encryption and KEK is used for DEK encryption.
The using cryptographic algorithm and targets are as follows.

Item		Standard	Key generation algorithm	Key length	Key Description
Key generation for mutual authentication among TOE's components	private key, public key	ISO/IEC 18033-2	RSAES(SHA-256)	2048bit	Asymmetric key pair for session key encryption
	private key, public key	ISO/IEC 14888-2	RSA-PSS(SHA-256)	2048bit	Asymmetric key pair for digital signature
Key generation for basic protection of internally transmitted TSF data	session Key	TTAK.KO-12.0190	HASH_DRBG(SHA-256)	256bit	Session key using session information as a key to encrypt internal transmission data
Key generation for basic protection of stored TSF data	Drived Key(DK)	TTAK.KO-12.0274	Password Based Key Derivation Functions, HMAC-SHA-2	256bit	Generating a derivation key to be used as the key of KEK through user input
	Master Key(KEK)	TTAK.KO-12.0190	HASH_DRBG(SHA-256)	256bit	Generating a KEK to encrypt DEK
	Secondary	TTAK.KO-	HASH_DRBG(SHA-	256bit	Generating a DEK to

Item		Standard	Key generation algorithm	Key length	Key Description
	Key(DEK)	12.0190	256)		encrypt TSF data

Key generation for mutual authentication among the TOE components is created using the public key cipher of validated cryptographic module. The encryption key generated for basic protection of the internally transmitted TSF data is generated by the random number generator of the verification handwriting cryptographic module.

To encrypt TSF data, first generating a DEK, encrypt the TSF data using the DEK as a key, and then generating a KEK to encrypt the DEK. Here, the DK used to encrypt KEK uses the PBKDF2 method. The method used and the pseudorandom function are as follows.

function	Algorithm	Remark
Derivation function	PBKDF2 (Password-Based Key Derivation Function 2)	- PCKS#5 - reference to NIST SP 800-132
Pseudo random number function using in PBKDB2	HMAC(SHA-256) of validated cryptographic module	ISO/IEC 9797-2

Satisfied security function component
FCS_CKM.1(2), FCS_RBG.1

6.2.3. Cryptographic key distribution

The cryptographic key and policy generated in CubeOne Manager of TOE is distributed to CubeOne Server by using the block cipher and hash function of validated cryptographic module. And the cryptographic key for mutual authentication of TOE components is distributed by using public key cipher of validated cryptographic module.

The algorithms used are as follows.

Item	Standard	Approved algorithm	Remark
Key distribution for the user data encryption	KS X 1213-1 KS X 1213-2	ARIA	block algorithm(ARIA) and hash-function(SHA256) of validated cryptographic module
	ISO/IEC 10118-3	SAH256	
Key distribution for the	ISO/IEC 18033-2	RSAES	public key cipher (RSAES) of

Item	Standard	Approved algorithm	Remark
basic protection of internally transmitted TSF data			validated cryptographic module

Satisfied security function component
FCS_CKM.2

6.2.4. Cryptographic key destruction

The kind of cryptographic keys generated by TOE and destruction time are as follows.

Item	Destruction method	Destruction time
key destruction related to user data encryption	Free memory after overwrite the memory area to '0' through initialization function of memory provided by validated cryptographic module.	Destruction after cryptographic operation of user data. (encryption/decryption)
	Free memory through Shutdown command of CubeOne Server.	Destruction when execute Shutdown command by administrator
key destruction related to TSF data encryption	Free memory after overwrite the memory area to '0' through initialization function of memory provided by validated cryptographic module.	Destruction after cryptographic operation of user data
key destruction related to transmitted TSF data	Free memory after overwrite the memory area to '0' through initialization function of memory provided by validated cryptographic module.	Destruction after cryptographic operation of user data

Satisfied security function component
FCS_CKM.4

6.2.5. Cryptographic operation (User data encryption)

The cipher algorithm, key length and operation mode of cryptographic operation are determined by creation of ITEM in CubeOne Manager. For the block cipher algorithm in TOE, the same cryptogram is not generated for the same statement because it uses IV.

The algorithms and key length used for ITEM and key length are as follows.

Item	Algorithm	Mode of operation	Key length
Block cipher	ARIA	CBC/CFB/OFB	128/192/256
	SEED	CBC/CFB/OFB	128
HASH function	SHA256 SHA384 SHA512	-	-

There is the function for plug-in and API according to operational environment supported in TOE. It uses the encryption/decryption function that cryptographic operation of validated cryptographic module provides.

Satisfied security function component
FCS_COP.1(1), FDP_UDE.1

6.2.6. Cryptographic operation (TSF data encryption)

The lists of cryptographic operation used to encryption of TSF data are follows.

Cryptographic operation	Standard	Algorithm	Key length
Mutual authentication among the TOE components	ISO/IEC 18033-2	RSAES(SHA-256)	2048bit
	ISO/IEC 14888-2	RSA-PSS(SHA-256)	2048bit
Basic protection of the internally transmitted TSF data	KS X 1213-1 KS X 1213-2	ARIA, CBC mode	256bit
	ISO/IEC 10118-3	SHA-256	
Basic protection of the stored TSF data	KS X 1213-1 KS X 1213-2	ARIA, CBC mode	256bit
	ISO/IEC 9797-2	HMAC(SHA-256)	256bit

The approved functions of validated cryptographic module used in TOE are follows.

Function call	Approved function	Description
Klib_Cypher	COLibSetCipherInfo COLibEncrypt COLibDecrypt	Data encryption/decryption function
svgendRSAkey	COLibGetRsaKey	Key pair generation function for RSA (encryption/decryption/sign/verify)
RSA_OAEP_Enc	COLibRsaKeyInit COLibSetRsaInfo COLibEncryptRsa	Encryption function of Public key cipher
RSA_OAEP_Dec	COLibRsaKeyInit COLibSetRsaInfo COLibDecryptRsa	Decryption function of Public key cipher
RSA_PSS_Verify	COLibRsaKeyInit COLibSetRsaInfo COLibDecryptRsa	Certification function of Digital signature
RSA_PSS_Sign	COLibRsaKeyInit COLibSetRsaInfo COLibEncryptRsa	Significance function of Digital signature
T_MakeKey	COLibGetKey	Key generation function
T_MessageDigest	COLibEncrypt	Hash function
DeriveKey	Klib_Cypher	PBKDF2 function
svgendkey_with_passwd	COLibSetCipherInfo COLibEncrypt	Creation and reservation function of KEK and DEK
encdatfile_with_passwd	COLibSetCipherInfo COLibEncrypt	Encryption function for TSF data
decdatfile_with_passwd	COLibSetCipherInfo COLibDecrypt	Decryption function for TSF data

Satisfied security function component
FCS_COP.1(2), FPT_PST.1

6.3. User data protection

If "Plug-In" type of TOE is running, the policy administrator can perform user data encryption by using the security function of CubeOne Manager. Select the target encrypted table and column from the CubeOne Manager and request the CubeOne Server to perform the encryption. CubeOne Server deletes the original table after performing encryption on the Encrypted Target column. When deleting the original table, perform the query "DROP table name PURGE;" The deleted table is not recovered because of PURGE option. To prevent the same encryption value for the same plain data, IV values is used for user data encryption.

If the column needs to be encrypted using an API type, the developer shall delete the original data to which the encryption is applied.

Satisfied security function component
FDP_UDE.1, FDP_RIP.1

6.4. Identification and authentication (FIA)

6.4.1. Authentication failure handling

The authentication method for CubeOne Manger and CubeOne Beacon is based on their ID and password. If five consecutive failed certifications occur, the authentication function is prevented for five minutes to avoid repeated attempts by the authentication process...

Item	Content
Count of Authentication failure	Default: 5 Times. * There is no method that change default value.
Action taken	Identification/authentication function inactivation during 5 minute

Satisfied security function component
FIA_AFL.1

6.4.2. Verification of secrets

The first time you run CubeOne Manager, the administrator tool of TOE, you must register a new administrator ID and password. The administrator password can be changed through the menu of the

CubeOne Manager after initial registration. When registering the administrator, the following items must be entered, and the verification criteria and requirements are as follows.

Item	Description	Verification criteria
CubeOne Username	CubeOne Manager ID	- Length: min. 9 ~ max. 30 - English letter, special , number char.
Password	Password of ID	- Length: min. 9 ~ max. 30 - Must include English letter, special, number character
Authentication password	Authentication password for generating secret key which is used TSF data encryption through PBKDF2.	

6.4.3. Identification and authentication

The administrator enters the administrator ID and password when installing the CubeOne Manager that performs the security management function of the TOE. For CubeOne Beacon, the installer password must be registered. Password combination rules can be created with not less than 9 to 30 characters including letters, special characters, and numbers. Passwords entered during authentication are masked so that they cannot be seen on the screen ("*") and do not provide a reason for their failure. If the identification of CubeOne Manager is fail, only the license and version information of TOE can be confirmed. In case of failure of CubeOne Beacon's identification, all the functions are disabled.

For CubeOne Manager, version information for licenses and TOEs can only be checked without certification and all functions cannot be performed without certification by CubeOne Beacon.

The authentication data for administrator authentication creates session IDs as a random number to prevent reuse.

Satisfied security function component
FIA_SOS.1, FIA_UAU.2, FIA_UAU.4, FIA_UAU.7, FIA_UID.2, FIA_UID.2

6.5. Security management (FMT)

6.5.1. Security functions and Protection of stored TSF data

After identification with CubeOne Manger, the policy adminiatrator can manage the keys and policies used to encrypt user data, manage CubeOne Server, review audit data, and change the administrator password. The administrator password has a rule of not less than 9 to 30 characters, including letters, special characters, and numbers.
In case of CubeOne Beacon, the authorized log administrator can perform following security functions: query audit data, control the approved IP to connect as log administrator, change the administrator's password. And the rule of changing password is the same as the CubeOne Manager.

Satisfied security function component
FMT_MOF.1, FMT_MTD.1, FMT_SMF.1

6.5.2. Management of ID and password

You can register the administrator ID and password on the first connection after installing CubeOne Manager, which is responsible for managing the security functions of TOE. CubeOne Manager can only register one administrator. The rules for registering IDs and passwords are as follows.

Item	Content	Description
New CubeOne Username	User ID of CubeOne Manager	- Length: min. 9 ~ max. 30 - English letter, special , number char.
New Password	Password of user ID	- Length: min. 9 ~ max. 30 - Must include English letter, special, number character
Confirm Password	Confirm password of user ID	
Authentication New password	Authentication password of CubeOne Manager	
Authentication Confirm password	Confirm authentication password of CubeOne Manager	
Date Format	Data format used in CubeOne Manager	yyyy-MM-dd: year-month-day
		yyyy/MM/dd: year/month/day
		MM/dd/yyyy: month/day/ year-
		dd/MM/yyyy: day/ month /year

The administrator password of CubeOne Beacon provides the ability to set passwords during installation, and the combination rules are the same as the CubeOne Manager.

Satisfied security function component
FMT_PWD.1, FMT_SMF.1

6.5.3. Security roles

The user provided by TOE is an authorized administrator. TOE's policy administrator can register only one administrator and manage all management functions provided by the TOE. The log administrator connects to the CubeOne Beacon and performs security management.

Satisfied security function component
FMT_SMR.1

6.6. Protection of the TSF (FPT)

6.6.1. Basic internal TSF data transfer protection

TOE performs mutual authentication and secure communication of each component and performs encryption through validated cryptographic module to protect TSF data transmitted between TOE components from exposure and change.

Item	TOE components		Cryptographic algorithm
mutual authentication	CubeOne Manager	CubeOne Security Server	1) public key algorithm - RSAES(2048) 2) Digital signature algorithm - RSA-PSS(2048)
	CubeOne Manager	CubeOne Server	
	CubeOne Security Server	CubeOne Server	
	CubeOne Security Server	CubeOne Beacon	
secure communication	CubeOne Manager	CubeOne Security Server	1) random number generator - HASH_DRBG(SHA-256) 2) symmetric key algorithm - ARIA-256(CBC) 3) hash algorithm - SHA-256 4) public key algorithm - RSAES(2048)
	CubeOne Manager	CubeOne Server	
	CubeOne Security Server	CubeOne Server	
	CubeOne Security Server	CubeOne Beacon	

Satisfied security function component
FPT_ITT.1

6.6.2. Basic protection of stored TSF data

TSF data stored in TOE is encrypted using ARIA-256 (CBC) of validated cryptographic module. The data stored in TOE is as follows.

TOE components	TSF data
CubeOne Manager	User data encryption key
	TSF Data Encryption Key (KEK, DEK, public key)

TOE components	TSF data
	User data encryption policy
	Identification information
	Audit data
CubeOne Server	TSF Data Encryption Key (KEK, DEK, public key)
	TOE set value
CubeOne Security Server	TSF Data Encryption Key (KEK, DEK, public key)
	TOE set value
CubeOne Beacon	Public key

Satisfied security function component
FTP_PST.1

6.6.3. TSF self-test

TSF performs its own test periodically during normal operation at startup. It also provides integrity verification of the TSF data and the TSF.

6.6.3.1. Self-test

The self-test for correct operation of the TOE is as follows.

TOE components	Program	Function	Period
CubeOne Server	~/bin/cubeone_guard	Manage the daemon processes on the CubeOne Server - cubebeacon - cubeone_auditor - cubeoned	Start-up and 5 minute cycle at CubeOne Beacon
	~/bin/cubebeacon	- encryption/decryption statistics - system usage statistics - send audit log data to CubeOne Security Server	Start-up and Restart by cubeone_guard at end of process
	~/bin/cubeone_auditor	Send success and failure audit log to CubeOne Security Server	Start-up and Restart by cubeone_guard

TOE components	Program	Function	Period
			at end of process
	~/bin/cubeoned	- perform the user data encryption/decryption - perform the mutual authentication among TOE components	Start-up and Restart by cubeone_guard at end of process
CubeOne Security Server	~/bin/sserverd	- store audit log data - perform the mutual authentication among TOE components	Start-up and 5 minute cycle at CubeOne Beacon

6.6.3.2. Integrity verification of TSF data

The TSF data integrity verification functions and intervals of TOE are as follows.

TOE components	Function	Period
CubeOne Manager	User data encryption key	When driven and requested by an authorized administrator
	User data encryption policy	
	Login information of CubeOne Manager	
	Audit log data generated by CubeOne Manager	
CubeOne Server	Configuration file of CubeOne Server	When driven and requested by an authorized administrator
CubeOne Security Server	Configuration file of CubeOne Security Server	When driven and requested by an authorized administrator

6.6.3.3. Integrity verification of TSF

The functions and intervals for verifying integrity of the TOE's execution code and library are as follows.

TOE components	Program	Function	Period
CubeOne Manager	CubeOne.exe	Execution file of CubeOne Manager	When driven and requested by an authorized administrator
	CoNet.dll	Communication module among TOE components	
	cubecmk.dll	Wrap library of validated cryptographic module	
	colib.dll	Validated cryptographic module	
CubeOne Server	~/bin/coinit	Initialize CubeOne Server	- When driven and requested by an authorized administrator - 1 hour cycle
	~/bin/cubebeacon	- encryption/decryption statistics - check Daemon service for CubeOne Server - daemon to send audit log to CubeOne Security Server	
	~/bin/cubeone_auditor	Daemon to send success and failure audit log to CubeOne Security Server	
	~/bin/cubeoned	Daemon Process to communicate with CubeOne Manager	
	~/bin/cubonesql	Perform initial encryption job as child process of cubeoned	
	~/bin/cubeone_guard	Daemon to monitor cubebeacon, cubeone_auditor, cubeoned	
	~/lib/libCOtbenc.so (tibero) ~/lib/libCOdb2enc (DB2)	C library for plug-in type	
	~/lib/libCOencapi.so	C library for API type	

TOE components	Program	Function	Period
	~/lib/libcolib.so	Validated cryptographic module	
CubeOne Security Server	~/bin/sserverd	Daemon Process to communicate among TOE components	- When driven and requested by an authorized administrator - 1 hour cycle
	~/bin/ssagent	Perform specified job as child process of sserverd	
	~/lib/libcolib.so	Validated cryptographic module	
CubeOne Beacon	/CubeOne_Beacon/webapps/eglobalsys/WEB-INF/classes/SqlMapMaria.xml	File related to SQL of WAS	Every administrator login

Satisfied security function component
FPT_TST.1

6.7. TOE access (FTA)

6.7.1. TOE session control

The administrator of TOE controls the administrator's management connection based on the connection IP when trying to connect to CubeOne Manager and CubeOne Beacon, and the unauthorized IP access attempt also denies the administrative access session.
The access rights of the CubeOne Manager limit the number of concurrent sessions to one, and for CubeOne Beacon, limit the number of concurrent sessions to three.
Lock the sessions after a 10 minute period of inactivity of the CubeOne Manager and require administrator authentication. CubeOne Beacon terminates the session after the inactive period (10 minutes).

Satisfied security function component
FTA_MCS.2, FTA_SSL.5, FTA_TSE.1