Security Target

CubeOne V3.0

eGlobal Systems Co., Ltd

This document is a translation of the Security Target written in Korean which has been evaluated.

Copyright © 2018. All rights reserved eGlobal Systems Co., ltd.

Security Target

CubeOne™ are registered trademark of eGlobal Systems Co., ltd
All other trademarks and/or service marks are the property of their respective owners

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
V3.0.0.1	2023.02.02	First Release in English	h.m.yang
V3.0.0.2	2023.06.19	Contents modified according to changes in verified cryptographic module update (COLib V1.1.0 → COLib V1.2.0)	c.y.park
V3.0.0.3	2023.07.18	Modification of contents according to observation report	c.y.park
V3.0.0.4	2023.10.25	Modification of contents according to changes in TOE and 3rd party versions	c.y.park
V3.0.0.5	2023.11.22	Modification of content according to review comments	c.y.park

Table of Contents

List of Figures

List of Tables

1. ST Introduction

This Document is ST of CubeOne V3.0 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 V3.0 Security Target
Document identification	CubeOne_ST(ENG)_V3.0.0.5_ENG
Version	V3.0.0.5
Developer	eGlobal Systems Co., ltd.
Issue date	2023.11.22
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 V3.0
TOE type	Database, Encryption
Detail version	rev.0025

Item		Specification
TOE components	CubeOne Manager	- CubeOne_Manager_V3.0.00.03
	CubeOne Server	[Plug-In] - CubeOne_Server_V3.0.00.03_L64_4.18_OR19C - CubeOne_Server_V3.0.00.03_A64_7.2_DB11.5 - CubeOne_Server_V3.0.00.03_L64_4.18_TI7 - CubeOne_Server_V3.0.00.03_L64_4.18_MY8 - CubeOne_Server_V3.0.00.03_W64_10_MS19 [API] - CubeOne_Server_V3.0.00.03_A64_7.2_API - CubeOne_Server_V3.0.00.03_S64_5.11_API - CubeOne_Server_V3.0.00.03_H64_B.11.31_API - CubeOne_Server_V3.0.00.03_L64_4.18_API - CubeOne_Server_V3.0.00.03_W64_10_API
	CubeOne Security Server	CubeOne_SServer_V3.0.00.03_L64_4.18_MY
Manual	Operating Manual	- CubeOne_OPE_V3.0.0.3
	Installation Manual	- CubeOne_PRE_V3.0.0.4
Developer		eGlobal Systems Co., Ltd

1.3. TOE overview

CubeOne V3.0 (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

Item	Specification
CubeOne Server	Perform cryptographic operation of user data for TOE
CubeOne Security Server	Save cryptographic policy and security audit log of TOE. Perform latent violation analysis and security alert of TOE.

Encryption keys and TSF data used to encrypt and decrypt user data are created and managed through CubeOne Manager. The encryption key and TSF data are encrypted using the verification target encryption algorithm of the verified encryption module.

1.3.2. TOE usage and major security features

The TOE provides the ability to server user data according to sections set by authorized administrators to prevent unauthorized opposition to information to be protected. The TOE provides a security audit function that records and manages audit data on major auditable events so that authorized administrators can safely operate the TOE within the organization's operating environment. In addition, the TOE provides cryptographic support functions such as encryption key management for user and TSF data encryption, cryptographic operation, user data protection function to encrypt user data and protect residual information, authorized administrator identity verification, authentication failure handling, and TOE configuration. Identification and authentication functions such as mutual authentication between elements, security management functions for security functions and role definitions, environment settings, etc., protection of TSF data transmitted between TOE components, protection of TSF data stored in storage controlled by the TSF, and the TSF itself. It provides TSF protection functions such as testing, and TOE access functions for access session management by authorized administrators.

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. The authorized policy administrator can connect to CubeOne Manager for security control. The authorized log administrator can connect CubeOne Security Server to check security alert and audit log.

1.3.3.1. Plug-In Type

The authorized policy administrator creates user data encryption/decryption keys and sets policies through the GUI provided in CubeOne Manager. CubeOne Manager sends and receives TSF data to CubeOne Security Server when login/logout.

CubeOne Server encrypts user data according to the policy set in CubeOne Manager and deletes the original plaintext data. Additionally, when an application service user requests user data, CubeOne Server decrypts it according to policy and delivers it to the Application Server.

CubeOne Security Server stores user data encryption/decryption keys and policies transmitted from CubeOne Manager, and stores encryption and decryption performance history generated by CubeOne Server and audit data generated from TOE components. Additionally, when the CubeOne Server is restarted, the key and policy for encrypting and decrypting user data are transmitted to the CubeOne Server.

The authorized log administrator can check security alert and audit log through CubeOne Security Server.

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

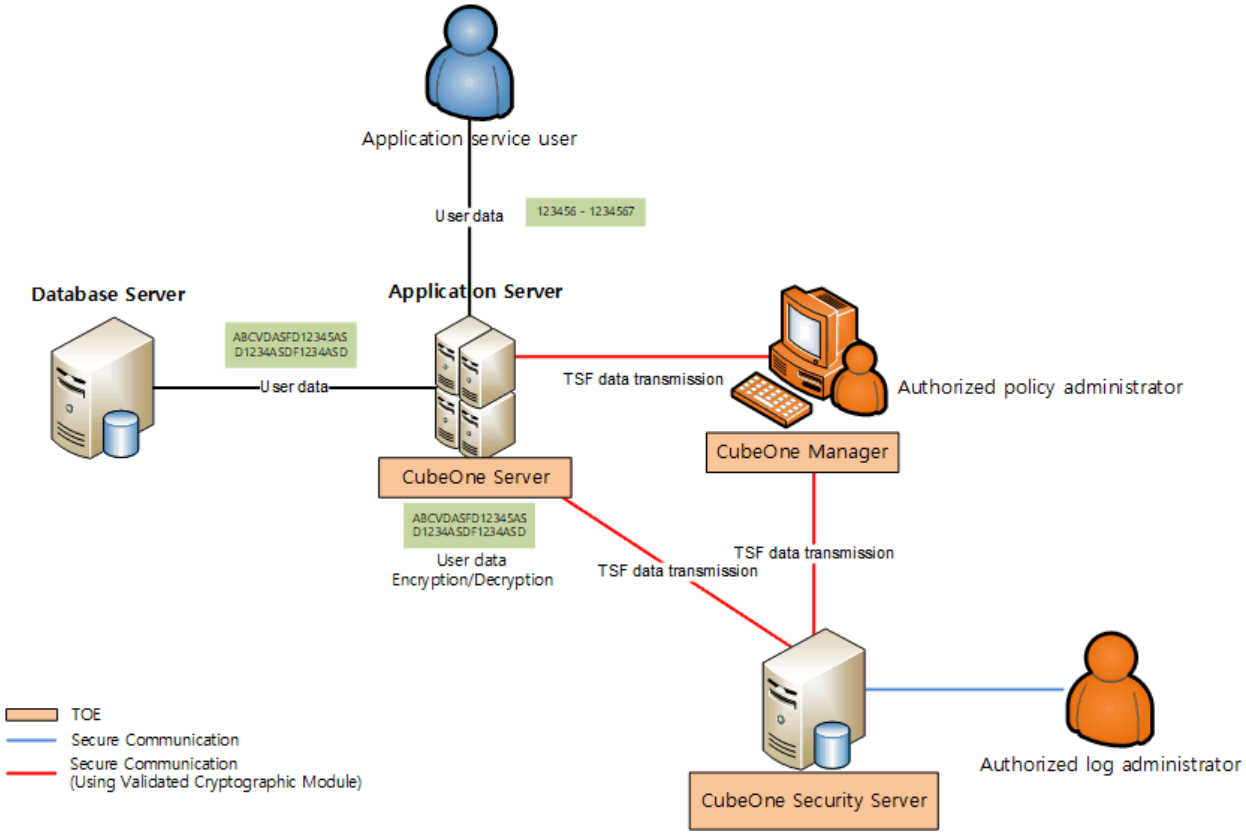


Figure 1. Plug-in type operational environment

1.3.3.2. API Type

The authorized policy administrator creates user data encryption/decryption keys and sets policies through the GUI provided in CubeOne Manager. CubeOne Manager send/receive TSF data to CubeOne Security Server when login/logout. The application service users can encrypt/decrypt data with API provided by TOE and must delete the original data after encrypt it. When application service users save user data, it is encrypted by CubeOne Server and stored in the Database Server. When a user searches, the CubeOne Server decrypts it and delivers it to the application service user.

CubeOne Security Server stores user data encryption/decryption keys and policies transmitted from CubeOne Manager, and stores encryption/decryption performance history generated by CubeOne Server and audit data generated from TOE components. Additionally, when the CubeOne Server is restarted, the key and policy for encrypting and decrypting user data are transmitted to the CubeOne Server.

The authorized log administrator can check security alert and audit log through CubeOne Security Server.

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

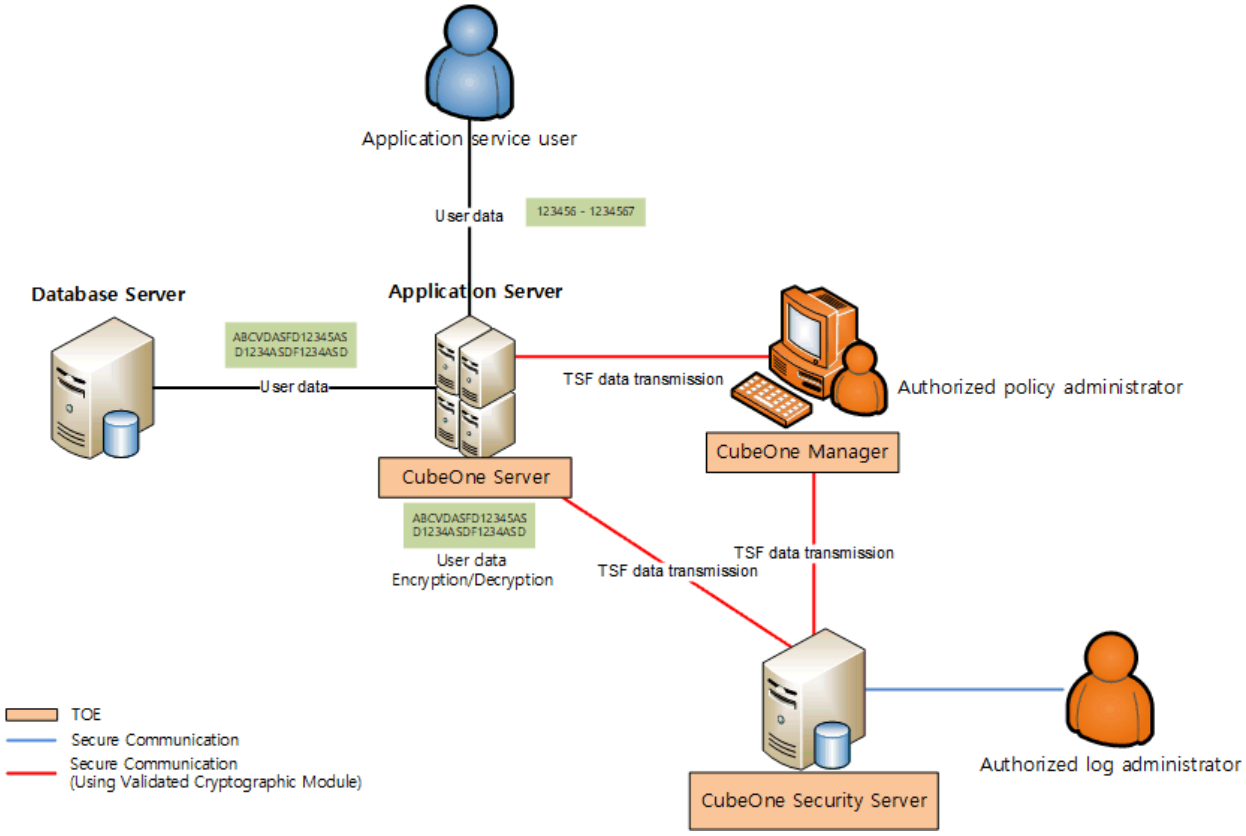


Figure 2. API type operational environment

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.

1.3.4. Non-TOE Hardware/ Software

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

Classification	Minimum Requirement			
CubeOne Server (Plug-In)	CPU	POWER7 3.0 Ghz or above	Intel Dual Core 1.8 GHz or above	Intel Dual Core 1.8 GHz or above
	Memory	4 GB or above		
	HDD	At least 200 MB of space required to install TOE		
	NIC	10/100/1000 Mbps X 1 Port or above		

Classification	Minimum Requirement					
	OS	AIX 7.2 (64 bit)	Rocky Linux 8.7 (64 bit) (kernel 4.18.0)	Windows Server 2019 (64 bit)		
	DBMS	DB2 11.5	Oracle 19c, Tibero 7, Mysql 8.0.35	MSSQL 2019		
CubeOne Server (API)	CPU	POWER7 3.0 Ghz or above	sparcv9 2848 MHz or above	Intel(R) Itanium 2 1.6 GHz or above	Intel Dual Core 1.8 GHz or above	Intel Dual Core 1.8 GHz or above
	Memory	4 GB or above				
	HDD	At least 200 MB of space required to install TOE				
	NIC	10/100/1000 Mbps X 1 Port or above				
	OS	AIX 7.2 (64 bit)	SunOS 5.11 (64 bit)	HP-UX B.11.31 (64 bit)	Rocky Linux 8.7 (64 bit) (kernel 4.18.0)	Windows Server 2019 (64 bit)
CubeOne Manager	CPU	Intel Core 2 Duo 2.40 GHz or above				
	Memory	4 GB or above				
	HDD	At least 200 MB of space required to install TOE				
	NIC	10/100/1000 Mbps X 1 Port or above				
	OS	Windows Server 2019 (64 bit)				
CubeOne Security Server	CPU	Intel Core 2 Duo 2.26 GHz or above				
	Memory	4 GB or above				
	HDD	At least 200 MB of space required to install TOE				
	NIC	10/100/1000 Mbps X 1 Port or above				
	OS	Rocky Linux 8.7 (64 bit) (kernel 4.18.0)				
	essential S/W	- Mysql 8.0.35 - Apache tomcat 9.0.82				
Authorized log administrator	Web browser	Chrome V 118.0 (64 bit)				

Table 1. Minimum operation specification of hardware

The uses of 3rd party software not included in the TOE are as follows.

Item	Software	Specification
CubOne Securty Server	Mysql 8.0.35	Database used for the audit repository of TOE
	Apache tomcat 9.0.82	WAS server for CubeOne Security Server

Table 2. 3rd party software not included in the TOE

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 Security Server.

1.4.1. Physical Scope

The physical scope of the TOE consists of CubeOne Manager, CubeOne Server, and CubeOne Security Server, which are software provided in CD form, and the verified cryptographic module is included in the 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. Software and certificates such as hardware, OS, DBMS, etc. required to operate the TOE are excluded from the physical scope of the TOE.
The physical scope of TOE is graphically represented as follows.

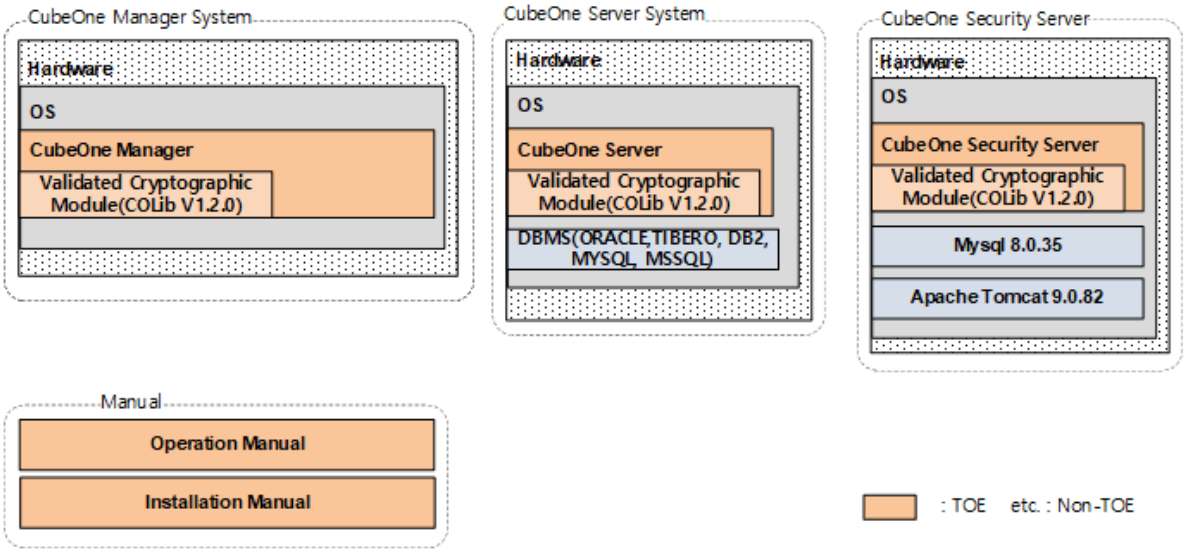


Figure 3. Physical scope of TOE

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 V3.0	
Detail version		rev.0025	
TOE Components	CubeOne Manager	- CubeOne_Manager_V3.0.00.03 : CubeOne_Manager_V3.0.00.03.exe	Included in CD
	CubeOne Server	[Plug-In] - CubeOne_Server_V3.0.00.03_L64_4.18_OR19C : CubeOne_Server_V3.0.00.03_L64_4.18_OR19C.tar - CubeOne_Server_V3.0.00.03_A64_7.2_DB11.5 : CubeOne_Server_V3.0.00.03_ A64_7.2_DB11.5.tar - CubeOne_Server_V3.0.00.03_L64_4.18_TI7 : CubeOne_Server_V3.0.00.03_L64_4.18_TI7.tar - CubeOne_Server_V3.0.00.03_L64_4.18_MY8 : CubeOne_Server_V3.0.00.03_L64_4.18_MY8.tar - CubeOne_Server_V3.0.00.03_W64_10_MS19 : CubeOne_Server_V3.0.00.03_W64_10_MS19.exe [API] - CubeOne_Server_V3.0.00.03_A64_7.2_API : CubeOne_Server_V3.0.00.03_A64_7.2_API.tar - CubeOne_Server_V3.0.00.03_S64_5.11_API : CubeOne_Server_V3.0.00.03_S64_5.11_API.tar - CubeOne_Server_V3.0.00.03_H64_B.11.31_API : CubeOne_Server_V3.0.00.03_H64_B.11.31_API.tar - CubeOne_Server_V3.0.00.03_L64_4.18_API : CubeOne_Server_V3.0.00.03_L64_4.18_API.tar - CubeOne_Server_V3.0.00.03_W64_10_API : CubeOne_Server_V3.0.00.03_W64_10_API.exe	Included in CD
	CubeOne Security Server	- CubeOne_SServer_V3.0.00.03_L64_4.18_MY : CubeOne_SServer_V3.0.00.03_L64_4.18_MY.tar	Included in CD
Manuals	Operation Manual	- CubeOne_OPE_V3.0.0.3 : CubeOne_OPE_V3.0.0.3.pdf	prints, Included in CD

Item		Content	status
	Installation Manual	- CubeOne_PRE_V3.0.0.4 : CubeOne_PRE_V3.0.0.4.pdf	prints, Included in CD

Table 3. Components of TOE

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

Item	Content
Module Name	COLib V1.2.0
Certification Number	CM-231-2028.6
Developer	Eglobal system
Issue Date	2023-06-19
Expiration Date	2028-06-19

Table 4. Validated cryptographic module

1.4.2. Logical Scope

Below represent security function of TOE.

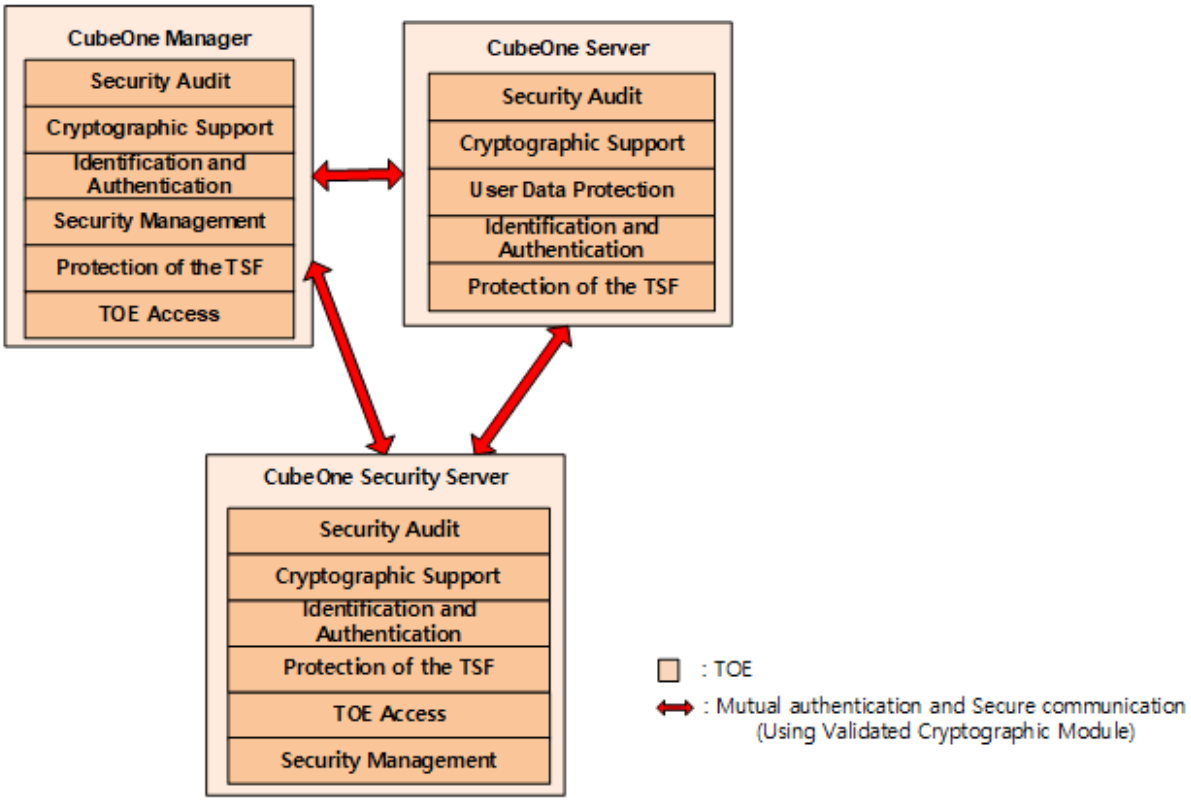


Figure 4. Logical scope of TOE

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. Audit records created in CubeOne Manager are sent to CubeOne Security Server when CubeOne Manager logs out.

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 Authorized policy administrators can read all audit data generated by CubeOne Manager 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. If the audit trail is saturated, the policy manager is notified in a pop-up and the audited event is ignored.

[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 and CubeOne Security Server.

When encrypting and decrypting TSF data, the ARIA encryption algorithm of the verified encryption module is used. For stored data, a key length of 256 bits is used, and for transmitted data, a key length of 128 bits is used.

After using the TSF data encryption key, the memory area of key is overwritten by '0' 3 times.

[Identification and authentication]

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. Random number are used to prevent reuse of administrator authentication information.

CubeOne Manager 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.

[Security Management]

The security function provided by Cubeone Manager 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, administrators 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 Security Server 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(SHA-256) and block cipher (ARIA-128) 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. When encrypting and decrypting TSF data, the ARIA encryption algorithm of the verified

encryption module is used. For stored data, a key length of 256 bits is used, and for transmitted data, a key length of 128 bits is used.
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' 3 times.

[User data protection]

CubeOne Server provides encryption and decryption functions for each column when encrypting and decrypting user data, and after encryption, the original user data in plain text is overwritten with '0' three times to completely delete it, ensuring that the previous information is not available.

[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(SHA-256) and block cipher (ARIA-128) 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]

CubeOne Security Server creates audit records according to auditable events such as password support, identification, and authentication. Audit records include event date, event type, identity, and event outcome. Audit records created in CubeOne Security Server, CubeOne Server, CubeOne Manager are stored in the DBMS where CubeOne Security Server is installed.

If a potential security violation, such as an authentication failure audit event or an integrity violation audit event, is detected during an audit event, a real-time warning screen is provided to the authorized log administrator.

Authorized log administrator can search all audit data stored in DBMS through CubeOne Security Server and can perform selective audit searches based on criteria with logical relationships.
If the audit trail exceeds 80% of the audit storage capacity, a real-time warning screen is provided to the authorized log administrator.
If the audit trail is saturated, a real-time warning screen is provided to the authorized log administrator and old audit records are overwritten.

[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 bits. After using the TSF data encryption key, the memory area of key is overwritten by '0' 3 times.

[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. Session ID are used to prevent reuse of administrator authentication information.

CubeOne Security Server 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.

[Security Management]

The password for authentication of CubeOne Security Server is registered during installation. The log administrator connects to the CubeOne Security Server 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(SHA-256) and block cipher (ARIA-128) 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]

Simultaneous access to CubeOne Security Server is limited to a maximum of 1 person.

CubeOne Security Server terminates sessions after 10 minutes of administrator inactivity, and security functions can be performed only after administrator must be identified and authenticated. CubeOne Security Server provides IP-based management access session control.

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. 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 V1.1'

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 V1.1' - 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	FDP_UDE.1	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	FPT_ITT.1	The same as DBEnc-PP
	FPT_PST.1(Extended)	FPT_PST.1(Extended)	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

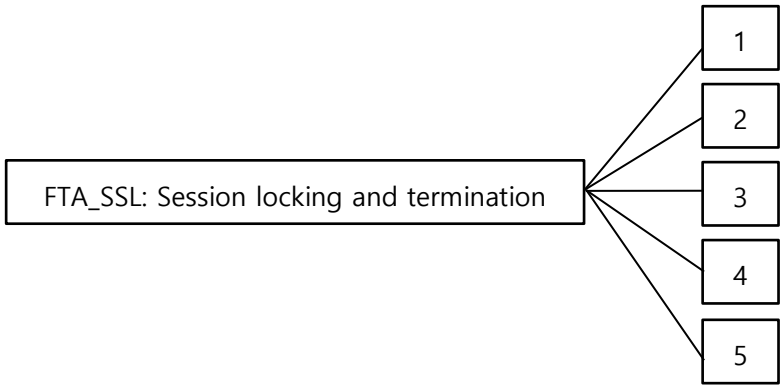
Family Behaviour	This family defines rules to protect TSF data stored within containers controlled by the TSF from the unauthorized modification or disclosure.	
Component leveling	FPT_PST Protection of stored TSF data — 1	
	FPT_PST.1	Basic protection of stored TSF data requires the protection of TSF 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 [selection: • *lock the session and re-authenticate the user before unlocking the session,* • *terminate*] an interactive session after a [assignment: *time interval of user inactivity*].

5. Security requirements

The security requirements specify security functional requirements and assurance requirements that must be satisfied by the TOE 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(CubeOne Manager)
	FAU_STG.4(2)	Action in case of possible audit data loss(CubeOne Security Server)
FCS	FCS_CKM.1(1)	Cryptographic key generation (User data encryption)
	FCS_CKM.1(2)	Cryptographic key generation (TSF data encryption)
	FCS_CKM.2	Cryptographic key distribution
	FCS_CKM.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

Security functional class	Security functional component	
	FIA_IMA.1(Extended)	TOE Internal mutual authentication
	FIA_SOS.1	Verification of secrets
	FIA_UAU.2	User authentication before any action
	FIA_UAU.4	Single-use authentication mechanisms
	FIA_UAU.7	Protected authentication feedback
	FIA_UID.2	User identification before any action
FMT	FMT_MOF.1	Management of security functions Behaviour
	FMT_MTD.1	Management of TSF data
	FMT_PWD.1(Extended)	Management of ID and password
	FMT_SMF.1	Specification of management functions
	FMT_SMR.1	Security roles
FPT	FPT_ITT.1	Basic internal TSF data transfer protection
	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(1)(Extended)	Management of TSF-initiated sessions(CubeOne Manager)
	FTA_SSL.5(2)(Extended)	Management of TSF-initiated sessions(CubeOne Security Server)
	FTA_TSE.1	TOE session establishment

Table 5. 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 real-time Security Server, 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 6], *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 6], *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(1) FAU_STG.4(2)	Actions taken due to the audit storage failure	
FCS_CKM.1(1) FCS_CKM.1(2)	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) FCS_COP.1(2)	Success and failure of the activity	
FDP_UDE.1(Extended)	Success and failure of user data	

Security functional component	Auditable event	Additional audit record
	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(Extended)	Success and failure of mutual authentication Modify of authentication protocol	
FIA_UAU.2	All use of the authentication mechanism	
FIA_UAU.4	Attempts to reuse authentication data	
FIA_UID.2	All use of the 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(Extended)	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(1)(Extended) FTA_SSL.5(2)(Extended)	Locking or termination of interactive session	
FTA_TSE.1	Denial of a session establishment due to the session establishment mechanism All attempts at establishment of a user session	

Table 6. 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 log 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 log 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 7. Selectable audit review methods] of audit data based on [criteria with following logical relations].

Item	Selection/ordering	Logical relation
Manager	query	AND of the entire period with one of the items below

Item	Selection/ordering		Logical relation
			Total, Server, Database Name, Workgroup, In Workgroup, Item
Security Server	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 (warning, critical, fatal)
		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, level
	Audit log	query	AND of the entered value among the items below - server name, date(start~end), level (success, fail)
		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 7. 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 [Notification to the authorized administrator**,** [a pop-up warning is provided to authorized policy administrator and a warning screen is displayed to authorized log administrator.]] 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 policy administrator] 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 Security Server] if the audit trail is full. .

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

5.1.2. Cryptographic support (FCS)

The encryption algorithms supported by the TOE are as follows, supports only the approved cryptographic algorithm.

Item	Approved algorithm	Detail	Standard criteria
Block cipher	ARIA	Operation mode: CBC, CFB-128, OFB Key Length: 128/192/256 bit	KS X 1213-1 KS X 1213-2
	SEED	Operation mode: CBC, CFB-128, OFB 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-12.0331
Public key cipher	RSAES	|n|: 2048bit e: 65537 Hash: SHA-256	ISO/IEC 18033-2
Digital signatures	RSA-PSS	|n|: 2048bit e: 65537 Hash: SHA-256	ISO/IEC 14888-2
MAC	HMAC	Hash:SHA-256	ISO/IEC 9797-2

Table 8. 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-12.0331) of "Table 8. appoved Cryptographic Algorithm"] and specified cryptographic key sizes [HASH_DRBG of "Table 8. 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 9. Cryptographic key generation"] and specified cryptographic key sizes [key length of "Table 9. Cryptographic key generation"] that meet the following: [standard of "Table 9. Cryptographic key generation"]

Item		Standard	Key generation algorithm	Key length	Key Description
Mutual authentication among TOE's components	private key, public key	ISO/IEC 18033-2	RSAES(SHA-256)	2048bit	Asymmetric key to encrypt mutual authentication data
	private key, public key	ISO/IEC 14888-2	RSA-PSS(SHA-256)	2048bit	Asymmetric key pair for digital signature
Basic protection of internally transmitted TSF data	session Key	TTAK.KO-12.0331	HASH_DRBG(SHA-256)	128bit	Session key using session information as a key to encrypt internal transmission data
Basic protection of stored TSF data	Drived Key(DK)	TTAK.KO-12.0334	Password Based Key Derivation Functions, HMAC-	256bit	Generating a derivation key to be used as the key of KEK through user

Item		Standard	Key generation algorithm	Key length	Key Description
			SHA-2		input
	Master Key(KEK)	TTAK.KO-12.0331	HASH_DRBG(SHA-256)	256bit	Generating a KEK to encrypt DEK
	Secondary Key(DEK)	TTAK.KO-12.0331	HASH_DRBG(SHA-256)	256bit	Generating a DEK to encrypt TSF data

Table 9. 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 10. Cryptographic key distribution"] that meets the following [standard of "Table 10. 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	SHA256	
Key distribution for the basic protection of internally transmitted TSF data	KS X 1213-1 KS X 1213-2	ARIA-128(CBC)	block cipher (ARIA), hash function (SHA256) and public key encryption(RSAES) provided by validated cryptographic module
	ISO/IEC 10118-3	SHA256	
	ISO/IEC 18033-2	RSAES(2048)	

Table 10. 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' 3 times] 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 8. Approved Cryptographic Algorithm"] and cryptographic key sizes [key length (ARIA 128/192/256, SEED 128) of "Table 8. Approved Cryptographic Algorithm"] that meet the following: [block cipher and hash function of "Table 8. 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 11. TSF data Cryptographic operation"] in accordance with a specified cryptographic algorithm [algorithm of "Table 11. TSF data Cryptographic operation"] and cryptographic key sizes [key length of "Table 11. TSF data Cryptographic operation"] that meet the

	following: [standard of "Table 11. 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	128bit
	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
	ISO/IEC 10118-3	SHA-256	-

Table 11. 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 a cryptographic key using the specified random bit generator that meets [TTAK.KO-12.0331]

5.1.3. User data protection (FDP)

5.1.3.1. FDP_UDE.1 User data encryption

Hierarchical to	No other components.
Dependencies	FCS_COP.1 Cryptographic operation
FDP_UDE.1.1	The TSF shall provide 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*] 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 Server ↔ CubeOne Security Server, CubeOne Manager ↔ CubeOne Security Server]

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

5.1.4.5. FIA_UAU.4 Single-use authentication mechanisms

Hierarchical to	No other components.
Dependencies	No dependencies
FIA_UAU.4.1	The TSF shall prevent reuse of authentication data related to [authentication mechanisms of "Table 12. Single-use authentication mechanisms"].

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

Table 12. 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/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 **authorized administrator** to be successfully identified before allowing any other TSF-mediated actions on behalf of that **authorized administrator**.

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 13. 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 13. 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 14. TSF Data list and management ability"] to [authorized policy administrator and authorized log administrator].

(*Reg.: Registration)

Authorized Administrator	**TSF data**	**Ability**			
		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	X	O	O
	Group information of cryptographic policy	O	O	O	O
	ITEM information for encryption	O	X	O	O
Authorized log administrator	Audit Data	O	X	X	X
	Administrator connection IP	O	O	O	X
	Administrator password	X	O	O	X

Table 14. 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 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) ID and 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 *the TSF*.
FTP_TST.1.2	The TSF shall provide **authorized administrators** with the capability to verify the integrity of *TSF data*.
FTP_TST.1.3	The TSF shall provide **authorized 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 {1} for management

	access by the same administrator who doesn't have the right to perform FMT_MOF.1.1 "Management actions" but has the right to perform a query in FMT_MTD.1.1 "Management" only c) [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(1) Management of TSF-initiated sessions (Extended)

Hierarchical to	No other components.
Dependencies	FIA_UAU.1 authentication or No dependencies.
FTA_SSL.5.1	The TSF shall *lock the session and/or re-authenticate the* ***policy administrator*** *before unlocking the session* the administrator's interactive session after a [10 minutes of the **policy administrator** inactivity].

* Application note: This requirement applies to session management by TSF for CubeOne Manager.

5.1.7.3. FTA_SSL.5(2) Management of TSF-initiated sessions (Extended)

Hierarchical to	No other components.
Dependencies	FIA_UAU.1 authentication or No dependencies.
FTA_SSL.5.1	The TSF shall *terminate* the administrator's interactive session after a [10 minutes of the **log administrator** inactivity].

* Application note: This requirement applies to session management by TSF for CubeOne Security Server.

5.1.7.4. 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 15. 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, 14
		FCS_CKM.4	12
11	FCS_CKM.2	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9, 10
		FCS_CKM.4	12
12	FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	9, 10
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]	10
		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	21
24	FIA_UID.2	-	-
25	FMT_MOF.1	FMT_SMF.1	28
		FMT_SMR.1	29
26	FMT_MTD.1	FMT_SMF.1	28
		FMT_SMR.1	29
27	FMT_PWD.1	FMT_SMF.1	28
		FMT_SMR.1	29
28	FMT_SMF.1	-	-
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(1)	FIA_UAU.1	21
35	FTA_SSL.5(2)	FIA_UAU.1	21
36	FTA_TSE.1	-	-

Table 16. 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 (MySQL) and can review audit data through CubeOne Security Server.

6.1.1. Potential security violation and security alert

The TOE can detect potential security violations like Table 17.

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(1) FAU_STG.4(2)	Audit event of actions taken due to the audit storage failure

Table 17. Potential security violations audit event

TOE generates audit data on such potential violation events, exposes the warning screen to the CubeOne Security Server, 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 policy manager can check the audit data generated in CubeOne Manager by

accessing CubeOne Manager, or the log manager can check the audit data generated in CubeOne Manager by accessing the web browser of CubeOne Security Server.
All audit data generated by the TOE are stored in the storage of CubeOne Security Server. The audit data can be checked through the log administrator.

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(1) FAU_STG.4(2)	Actions taken due to the audit storage failure	
FCS_CKM.1(1) FCS_CKM.1(2)	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) FCS_COP.1(2)	Success and failure of the activity	
FDP_UDE.1(Extended)	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(Extended)	Success and failure of mutual authentication Modify of authentication protocol	
FIA_UAU.2	All use of the authentication mechanism	
FIA_UAU.4	Attempts to reuse authentication data	
FIA_UID.2	All use of the user identification mechanism, including the user identity provided	

Security functional component	Auditable event	Additional audit record
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(Extended)	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(1) FTA_SSL.5(2)	Locking or termination of interactive session	
FTA_TSE.1	Denial of a session establishment due to the session establishment mechanism All attempts at establishment of a user session	

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 Security Server, and only authorized administrators can be interrogated.
It provides the functions of security alert, review, and analysis of security audit generated in TOE.

An authorized policy administrator can review audit data generated in CubeOne Manager through CubeOne Manager.
An authorized log administrator can review all audit data of TOE stored in the audit storage (DBMS) through CubeOne Security Server.

The auditable records which administrator can review are as follows.

Item	Selection/ordering		Logical relation
Manager	query		AND of the entire period with one of the items below Total, Server, Database Name, Workgroup, In Workgroup, Item
Security Server	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 (warning, critical, fatal)
		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, level
	Audit log	query	AND of the entered value among the items below - server name, date(start~end), level (success, fail)
		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

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 trail is saturated, the policy administrator is notified in a pop-up and the audited events are 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 Security Server. If the audit trail is saturated, a warning screen is displayed in real time and the oldest audit data is overwritten.

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 V1.2.0
Certification Number		CM-231-2028.6
Developer		Eglobal system
Issue Date		2023-06-19
Expiration Date		2028-06-19
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	Standard	Approved function	Key length	Operation mode
Block cipher algorithm	KS X 1213-1 KS X 1213-2	ARIA	128/192/256 bit	CBC, CFB-128, OFB
	TTAS.KO-12.0004/R1 TTAS.KO-12.0025	SEED	128 bit	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
Mutual authentication among TOE's components	private key, public key	ISO/IEC 18033-2	RSAES(SHA-256)	2048bit	Asymmetric key to encrypt mutual authentication data
	private key, public key	ISO/IEC 14888-2	RSA-PSS(SHA-256)	2048bit	Asymmetric key pair for digital signature
Basic protection of internally transmitted TSF data	session Key	TTAK.KO-12.0331	HASH_DRBG(SHA-256)	128bit	Session key using session information as a key to encrypt internal transmission data
Basic protection of stored TSF data	Drived Key(DK)	TTAK.KO-12.0334	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-12.0331	HASH_DRBG(SHA-256)	256bit	Generating a KEK to encrypt DEK
	Secondary Key(DEK)	TTAK.KO-12.0331	HASH_DRBG(SHA-256)	256bit	Generating a DEK to encrypt TSF data

The key used 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 validated cryptographic module.

To encrypt stored TSF data, create a DEK, use the DEK as a key to encrypt the TSF data, and create 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 is distributed to CubeOne Server by using the block cipher and hash function of validated cryptographic module.
Distribution of encryption keys for encryption communication to protect TSF data transmitted between TOE components uses block ciphers, hash functions, and public key encryption algorithms provided by verified encryption modules.

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-256(CBC)	block algorithm(ARIA) and hash-function(SHA256) of validated cryptographic module
	ISO/IEC 10118-3	SHA256	
Key distribution for the basic protection of internally transmitted TSF data	KS X 1213-1 KS X 1213-2	ARIA-128(CBC)	block algorithm(ARIA), hash-function(SHA256) and public key encryption(RSAES) of validated cryptographic module
	ISO/IEC 10118-3	SHA256	
	ISO/IEC 18033-2	RSAES(2048)	

Satisfied security function component

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' 3 times through initialization function of memory provided by validated cryptographic module.	Destruction after cryptographic operation of user data. (encryption/decryption)
	Free memory after overwrite the memory area to '0' 3 times when CubeOne Server is terminated	Destroyed upon termination of CubeOne Server
key destruction related to TSF data encryption	Free memory after overwrite the memory area to '0' 3 times 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' 3 times 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	Standard	Algorithm	Mode of operation	Key length
Block cipher	KS X 1213-1 KS X 1213-2	ARIA	CBC/CFB/OFB	128/192/256
	TTAS.KO-12.0004/R1 TTAS.KO-12.0025	SEED	CBC/CFB/OFB	128
HASH function	ISO/IEC 10118-3	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	128bit
	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 10118-3	SHA-256	-

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

Approved function	Description
COLibSetCipherInfo COLibEncrypt COLibDecrypt	Data encryption/decryption function

Approved function	Description
COLibGetRsaKey	Key pair generation function for RSA (encryption/decryption/sign/verify)
COLibRsaKeyInit COLibSetRsaInfo COLibEncryptRsa	Encryption function of Public key cipher
COLibRsaKeyInit COLibSetRsaInfo COLibDecryptRsa	Decryption function of Public key cipher
COLibRsaKeyInit COLibSetRsaInfo COLibDecryptRsa	Certification function of Digital signature
COLibRsaKeyInit COLibSetRsaInfo COLibEncryptRsa	Significance function of Digital signature
COLibGetKey	Key generation function
COLibEncrypt	Hash function
COLibSetCipherInfo COLibEncrypt COLibDecrypt	Pseudorandom number function used in the PBKDF2 function
COLibSetCipherInfo COLibEncrypt	Creation and reservation function of KEK and DEK
COLibSetCipherInfo COLibEncrypt	Encryption function for TSF data
COLibSetCipherInfo COLibDecrypt	Decryption function for TSF data

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

6.3. User data protection

It provides a column-specific encryption and decryption method for user data stored in the DBMS that the TOE wants to protect, and performs encryption and decryption in the application server or DBMS according to the API and plug-in method.

After encrypting and decrypting user data, it provides a mechanism to delete the original data, which is plain text, so that it cannot be recovered by overwriting it three times with '0'.

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 Security Server 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 minutes

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 - Must include 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 to generate key encryption key through PBKDF2 derivation	

Item	Description	Verification criteria
	function	

The log administrator must register password during installation of CubeOne Security Server. The password combination rule is 9 to 30 characters, including all English letters, special characters, and numbers. After initial registration, the administrator password can be changed After logging in to the web security management screen of CubeOne Security Server

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 Security Server, 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 failed, only the version information of TOE can be confirmed. In case of failure of CubeOne Security Server's identification, all the functions are disabled.

When the policy administrator attempts CubeOne Manager authentication, uniqueness is guaranteed using a random number generated by a verified random number generator to prevent reuse of authentication attempts.
When the log administrator attempts to authenticate with the CubeOne Security Server, uniqueness is guaranteed through a session ID for each session, preventing reuse of authentication attempts.

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

6.4.4. Mutual authentication between components

TOE components perform mutual authentication with each other. The timing of mutual authentication and the encryption algorithm used are as follows.

Item	Mutual authentication time	Algorithm

Item	Mutual authentication time	Algorithm
CubeOne Manager ↔ CubeOne Security Server	- When logging in/logging out of Manager - When distributing policies	RSAES(SHA-256) RSA-PSS(SHA-256)
CubeOne Server ↔ CubeOne Sercurity Server	- When running the server	RSAES(SHA-256) RSA-PSS(SHA-256)
CubeOne Manager ↔ CubeOne Server	- When distributing policies	RSAES(SHA-256) RSA-PSS(SHA-256)

The mutual authentication method is as follows. Assume that the element requesting mutual authentication is A, and the element receiving the request is B.

1. Both A and B generate an RSA key pair and exchange public keys with each other.
2. B generates a random number key with a verified random number generator and then performs public key encryption (RSAES) on the random number key with A's public key.
3. B attaches the public key encrypted value and the digital signature sign value created with B's private key and then sends it to A.
4. A decrypts the received encryption value with B's public key and authenticates the sign value. A decrypts the public key encryption value with the private key and verifies the random number key.
5. A performs public key encryption (RSAES) by attaching AuthID and the decrypted random number key to B's public key.
6. A attaches the public key encrypted value and the digital signature sign value created with A's private key and then sends it to B.
7. B decrypts the received encryption value with A's public key and authenticates the sign value. B decrypts the public key encryption value with the private key and verifies the random number key.
8. B separates the AuthID and the random number key and checks whether the AuthID and the random number key are the same.
9. B combines the session ID and the session key value generated by a random number generator. B generates a session value by using ARIA encryption with random number key and sends it to A.
10. A decrypts(ARIA) the encrypted session value received from B using random number key.
11. A encrypts data including the session ID with the session key and transmits it to B.
12. B decrypts the encryption value received from A using the session key and compares the session ID to confirm whether it is an authenticated session.

The AuthID used for mutual authentication is created based on the MAC address in the case of CubeOne Manager, and in the case of CubeOne Server and CubeOne Security Server, AuthID is transmitted from the Manager when distributing policies.

Satisfied security function component
FIA_IMA.1

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 Security Server, 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.

The list of security functions and security function management capabilities that the authorized policy/log administrator can perform are as shown in the table below.

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 18. List and Action of security functions

The list of TSF data and management capabilities that the authorized policy/log administrator can perform are listed in the table below.

Authorized Administrator	TSF data	Ability			
		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	X	O	O
	Group information of cryptographic policy	O	O	O	O
	ITEM information for encryption	O	X	O	O
Authorized log administrator	Audit Data	O	X	X	X
	Administrator connection IP	O	O	O	X
	Administrator password	X	O	O	X

Table 19. TSF Data list and management ability

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	Confirm authentication	

password	password of CubeOne Manager	

The administrator password of CubeOne Security Server 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 CubeOne Manager. The log administrator can only access CubeOne Security Server, and as a single administrator, there is only one account and manages all management functions provided by CubeOne Security Server.

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	
secure communication	CubeOne Manager	CubeOne Security Server	1) symmetric key algorithm - ARIA-128(CBC) 2) hash algorithm - SHA-256
	CubeOne Manager	CubeOne Server	
	CubeOne Security Server	CubeOne Server	

Satisfied security function component
FIA_IMA.1, FPT_ITT.1

6.6.2. Basic protection of stored TSF data

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

TOE components	TSF data	encryption algorithm
CubeOne Manager	User data encryption key	ARIA-256(CBC)
	TSF Data Encryption Key (KEK, DEK)	ARIA-256(CBC)
	User data encryption policy	ARIA-256(CBC)
	Audit data	ARIA-256(CBC)
CubeOne Server	TSF Data Encryption Key (KEK, DEK)	ARIA-256(CBC)

TOE components	TSF data	encryption algorithm
	TOE set value	ARIA-256(CBC)
CubeOne Security Server	TSF Data Encryption Key (KEK, DEK)	ARIA-256(CBC)
	TOE set value	ARIA-256(CBC)
	Policy administrator Password	SHA-256
	Log administrator Password	SHA-256

Satisfied security function component
FTP_PST.1

6.6.3. TSF self-test

TSF performs its own test periodically during normal operation and 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 Manager	CubeOne.exe	Cubeone Manager executable file	Start-up and 1 hour cycle at CubeOne Manager
CubeOne Server	~/bin/cubeone_guard64	Manage the daemon processes on the CubeOne Server - cubebeacon - cubeone_auditor - cubeoned	Start-up and 1 hour cycle at CubeOne Server
	~/bin/cubebeacon64	- 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_auditor64	Send success and failure audit log to CubeOne Security Server	Start-up and Restart by cubeone_guard at

TOE components	Program	Function	Period
			end of process
	~/bin/cubeoned64	- 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/sserver.sh	- start CubeOne Security Server - Secruity Server daemon process management	Start-up and 1 hour cycle at CubeOne Security Server
	~/bin/sserverd64	- store audit log data - perform the mutual authentication among TOE components	Start-up and Restart by sserver.sh at end of process

6.6.3.2. Integrity verification of TSF and TSF data

The TOE's TSF and TSF data integrity verification function and cycle 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	
	cubecmc.dll	Wrap library of validated cryptographic module	
	colib.dll	Validated cryptographic module	
CubeOne Server	~/bin/coinit64	Initialize CubeOne Server	- When driven and requested by an authorized administrator
	~/bin/cubebeacon64	- encryption/decryption statistics - check Daemon service for CubeOne Server - daemon to send audit log to CubeOne Security Server	

TOE components	Program	Function	Period
	~/bin/cubeone_auditor64	Daemon to send success and failure audit log to CubeOne Security Server	
	~/bin/cubeoned64	Daemon Process to communicate with CubeOne Manager	
	~/bin/cubonesql64	Perform initial encryption job as child process of cubeoned	
	~/bin/cubeone_guard64	Daemon to monitor cubebeacon, cubeone_auditor, cubeoned	
	~/bin/co_check_integrity	Perform integrity verification on files when running	
	~/lib/libCOtbenc.so (Tibero) ~/lib/libCOdb2enc (DB2) ~/lib/libCubeOnej.so (ORACLE) ~/lib/libCOmysqlenc.so (MySQL) ~₩₩lib₩₩extcube.dll (MSSQL)	C library for plug-in type	
	~/lib/libCOencapi.so	C library for API type	
	~/lib/libcolib.so	Validated cryptographic module	
	~/var/conf/cubeone.conf	Cubeone Server configuration file	
CubeOne Security Server	~/bin/sserverd	Daemon Process to communicate among TOE components	- When driven and requested by an authorized administrator
	~/bin/ssagent	Perform specified job as child process of sserverd	
	~/bin/co_check_integrity	Perform integrity verification on files when running	
	~/lib/libcolib.so	Validated cryptographic module	
	~/var/conf/sserver.conf	Cubeone Security Server configuration file	
	~/data/BACKUP/param.comm ~/data/BACKUP/param.dat ~/data/BACKUP/param_pw.dat	Communicate with CubeOne Manager and perform encryption and decryption	

Satisfied security function component
FPT_TST.1

6.7. TOE access (FTA)

6.7.1. TOE session control

When the policy manager attempts to access CubeOne Manager, the connection is controlled based on the manager's IP entered when installing CubeOne Security Server.

When the log manager attempts to connect to the CubeOne Security Server, the administrator's access is controlled based on the connection IP, and the session is rejected when an unauthorized IP connection is attempted.

CubeOne Manager's access rights limit the number of concurrent sessions to 1, and CubeOne Security Server limits the number of concurrent sessions to 1.

After a period of CubeOne Manager inactivity (10 minutes), the session is locked and administrator re-authentication is required. CubeOne Security Server terminates the session after a period of inactivity (10 minutes).

There is only one IP that can connect to CubeOne Manager, so simultaneous access is not possible. CubeOne Security Server limits the number of IPs that can be connected to one, and also limits the number of simultaneous connection sessions to one, so existing connections are blocked when simultaneous connections are attempted.

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