Unification of gateway through the Whole network generation.
World Best Security Solution
Next Generation Gateway. NXG

nxG IPS 6000 V1.6

Security Target

Version 1.13

SECUI

Revision History

ST_NXG IPS 6000_V1.13.doc		
Revision	**Date**	**Description**
V1.13	2007.06.26	♦ Corrected typing bugs.
V1.13	2007.04.23	♦ Deleted "Network line failure" in *5.1.5.2. FPT_FLS keeping safe conditions*. ♦ Deleted "2-tiered defense" in FDP_IFF.1.6's f of *5.1.2.2. FDP_IFF data flow control function.* ♦ Deleted "Network line failure," "High availability failure line detour" in *5.1.6.1 FRU_FLT resistance against failure*. ♦ Deleted 6.1.7 Network traffic control (Traffic_Control). ♦ Deleted the description for 2-tiered defense on page 96 and the 2-tiered defense on page 97. ♦ Deleted "Network traffic control (Traffic_Control)" for FPT_FLS.1, FRU_FLT.1 in *7.5.1. Security functions for IT security requirements*.
V1.13	2007.04.13	♦ Deleted "Traffic shaping" in *1.4 Terminology and acronyms*. ♦ Deleted the description for network traffic control in *2.2.2. Logical boundary and scope*. ♦ Deleted *6.1.7.1. Network traffic control (Traffic_Control).* ♦ Changed FRU_RSA.1 Security functions for assigned maximum values to Illegal access and intrusion prevention (IPS) in *7.5.1. Security functions for IT security requirements.*
V1.12	2007.04.11	♦ Added the description for policy conversion tool into *2.6.3. TSF protection.* ♦ Added interfaces into *3.11. Policy conversion tool*.
V1.11	2007.03.26.	♦ Added descriptions for bridge mode and added requirements for accepting PP into *2.1.1. TOE environment*. ♦ Added the description for the CPU using AUX and console into *2.2.1. Physical boundary and scope*.
V1.10	2007.03.21.	♦ Added descriptions for AUX and console into *2.2.1. Physical boundary and scope*.
V1.9	2007.02.15	♦ Corrected typing bugs. ♦ Deleted the exceptional scope for evaluating the bridge mode.
V1.8	2007.02.09	♦ Changed the operating system for management console to Windows XP Service Pack 2.

		♦ Added the description for the reason why *8.2. Redefining security profile* has no user manual. ♦ Deleted the client separation function and items for block agent from the evaluation exception scope. ♦ Corrected according to the FSP modification items. ♦ Deleted the anti-virus server after the syslogd server has been added into the IT environment.
V1.7	2007. 01.31	♦ Added the description for client separation (IPS-NCQA) into *1.4 Terminology and acronyms*. ♦ "2.2.3. Added the agent for TOE bridge mode and client separation mode into the *2.2.3. Evaluation exception scope*.
V1.6	2007. 1. 11	♦ Corrected additional items. ♦ Added an assumption "A. Safe TOE external server" and a security purpose for environment "OE. Safe TOE external server."
V1.5	2006. 12. 29	♦ Corrected according to OR.
V1.4	2006. 10. 13	♦ Modified cover and document format. ♦ Applied the IPS P.P (Network Intrusion Protection System Protection Profile).
V1.3	2006.08.02	♦ Modified cover and document format (In whole document).
V1.2	2005.10.04	♦ Deleted descriptions for internal interfaces (In whole document). ♦ Modified <*Figure 2-1> TOE security functions and external interfaces* on page 17. ♦ Modified <*Figure 2-28> Internal interfaces of FWmon and FWmonbak* on page 170 ♦ Deleted FWB_Execute_FW_if on page 171.
V1.1	2005.09.23	♦ Modified the interface error messages (In whole document). ♦ Added the description on the TOE scope for the anti-virus engine (on page 18, 95, and 96).
V1.0	2005.08.04	♦ The draft has been newly registered.

Table of contents

List of figures

List of tables

1. Security Target Introduction

This security target describes the Security Target of Evaluation (TOE) functions and scope for a SECUI.COM product and provides TOE security environment and objectives, IT security requirements, TOE summary specification, protection profile claims and rationale.

TOE is an intrusion protection system that is using the SECUI developed operating system (SecuiOS), which comprises minimum specifications to support the TOE security function.

It is assumed that you are familiar to TCP/IP, Internet, network security, intrusion protection system (IPS).

1.1. ST Identification

File name	ST_NXG IPS 6000_V1.13.doc
ST version	NXG IPS 6000 V1.6 Security Target Specification version 1.13
Document logging	Included in Revision History
ST author	Technology Planning Team/Information Protection Institute/SECUI.COM
ST date	June 26, 2007
Evaluation standards	Information Protection System Common Evaluation Criteria V2.3 (Ministry of Information and Communication Official Notice 2005-25)
Protection profile identification	Network intrusion protection system protection profile V1.1
Evaluation Assurance Level	EAL4
TOE identification	NXG IPS 6000 V1.6
Product group	Intrusion protection system
Key words	Intrusion Protection System, IPS, Security Target, ST, Security Objectives, Identification & Authentication, Access Control, Information Flow Control
Evaluation institute	Korea Information Security Agency
Certification institute	National Intelligence Service—IT Security Certificate Office Center

1.2. ST Overview

TOE is the intrusion protection system (IPS) that is installed on a vulnerable network point of contact to block any illegal external intrusion and attack. TOE provides the following functions: security audit function that creates or retrieves audit logs, security function, security management (controls parameters, TSF data, and security roles), access control and intrusion protection by accessing subject and information, TOE administrator identification and authentication, and guaranteeing TSF stability.

This document is divided into the following chapters:

Chapter 1. **Security Target Introduction**: Introduces Security Target, TOE, and common evaluation criteria, and provides the document summary.

Chapter 2. **TOE Description**: Describes the TOE boundaries and scope for this evaluation and its functions and objectives.

Chapter 3. **TOE Security Environment:** Describes assumptions for TOE environment, TOE or TOE environment threat to property, and organization security policy that TOE must keep.

Chapter 4. **Security Objectives**: Defines the TOE security objectives and security objectives for environment.

Chapter 5. **IT Security Requirements**: Identifies functions and assurance requirements satisfied by IT environment to guarantee TOE security objectives and explains about operations for IT security requirements.

Chapter 6. **TOE Summary Specification**: Describes information about IT security functions and assurance measures for the TOE .

Chapter 7. **Rationale**: Verifies the validity of security objectives against assumptions, threats, and organization security objectives, and verifies the validity of IT security requirements against security objectives and IT security functions against TOE security requirements.

Chapter 8. **Protection Profile Claims**: Identifies the protection profile for acceptance as well as protection profile acceptance condition, identifies IT security requirements to satisfy allowed operations, and provides rationale to explain about security objectives and requirements for the protection profile and difference between them.

1.3. Common Criteria Conformance Claims

TOE conforms to the following common evaluation criteria:

Information Protection System Common Evaluation Criteria V2.3, and Ministry of Information and Communication Official Notice 2005-25

Ministry of Information and Communication/Korea Information Security Agency (Common Criteria, Version 2.3)

- Conform to Part 2 of common evaluation criteria
- Conform to Part 3 of common evaluation criteria
- Evaluation Assurance level: EAL4

・PP Claims: Network Intrusion Protection System Protection Profile V1.1

1.4. Terminology and Acronyms

A

ActiveX

HTML component or program on Internet explorer that provides sophisticated functions. It provides the function to vitalize the Web page and implements the function for users to manipulate it by interaction.

Assignment

The specification of an identified parameter in a component.

Attack potential

The perceived potential for success of an attack, should an attack be launched, expressed in terms of an attacker's expertise, resources and motivation.

Audit trail

A set of disk records that has recorded system access users and activities.

Augmentation

The addition of one or more assurance component(s) from CC Part 3 to an EAL or assurance package.

Authorized administrator

Authorized user that securely operates or manages TOE according to the TOE security policies.

Authorized user

Information used to verify the claimed identity of a user.

Authentication data

Information used to verify the claimed identity of a user.

C

CGI

Interface used for exchanging information between the world-wide Web(WWW) server and back-end program (called gateway). It is used to create interactive Web pages that generate queries to the database server via the Web server. The interface specification may vary depending on the platform such as UNIX, Windows, and the like. Common languages for developing gateways are Pearl in

UNIX that can easily process text strings and Visual Basic in Windows NT.

Class

A grouping of families that share a common focus.

Communication object

External IT entities that are mutually authenticated for security communications with TOE.

Component

The smallest selectable set of elements that may be included in a PP, an ST, or a package.

D

Dependency

A relationship between requirements such that the requirement that is depended upon must normally be satisfied for the other requirements to be able to meet their objectives.

Dynamic rule

Temporary policy for allowed sessions that appear in a short period.

E

EAL (Evaluation Assurance Level)

A package consisting of assurance components from CC Part 3 that represents a point on the CC predefined assurance scale.

Element

An indivisible security requirement.

Ethernet

A LAN model that has been collaboratively developed by DEC, Intel, and Xerox.

Extension

The addition to an ST or PP of functional requirements not contained in CC Part 2 and/or assurance requirements not contained in CC Part 3.

External IT entity

Any IT product or system, untrusted or trusted, outside of the TOE that interacts with the TOE.

F

Family

A grouping of components that share security objectives but may differ in emphasis or rigour.

H

Hash

An activity that replaces the variable length data blocks and messages with unique hash signs with fixed length.

Hash bucket

A set of hash data with the same category when they are classified based on certain criteria.

Host

Host means the computer that can perform duplex communication in case of Internet and the mainframe computer in the mainframe computing environment.

HSRP (Hot standby router protocol): Constant waiting router protocol

HSRP is the route decision protocol that operates multiple routers linked to host computers as a virtual router. If the first-hop router does not work, the other routers are changed to “Hot standby” to maintain the connection. Configured on Cisco routers running over IPoE, FDDI, and Token Ring LANs, HSRP provides automatic router backup. The protocol is fully compatible with Novell's IPX, AppleTalk, and Banyan VINES, and with Xerox Network Systems (XNS) and DECnet.

Developed by Cisco and specified in IETF RFC 2281, HSRP ensures that only a single router (called active router) forwards packets on behalf of the virtual router at any given time. A standby router is chosen to be ready to become the active router, when the current active router fails. HSRP defines a mechanism used to determine active and standby routers by referring to their IP addresses. If these are determined, the failure of an active router will not cause any significant interruption of connectivity.

HTML (Hyper Text Mark-up Language)

An authoring language used to create home pages on the Internet's World Wide Web.

Human user

Any person who interacts with the TOE.

I

Identity

A representation (e.g. a string) uniquely identifying an authorised user, which can either be the full or abbreviated name of that user or a pseudonym.

Information protection system common evaluation criteria

The evaluation criteria for information protection systems that has been announced by the minister of Information and Communications in May 21, 2005. The common evaluation criteria is the Korean version of International Common Criteria that was developed based on the common understanding and universal language extracted from existing evaluation criteria documents in different countries.

Information flow control

An activity that applies security policies to user data in the FPD class based on the security attribute and information security attribute type for the information flow initiator.

Integrity

The feature preventing information and resources from illegal change.

Internal IT entity

All IT products or systems that interoperate with TOE under the internal network environment.

Internet

The vast communications network that connects computer networks around the world. A group of networks that interconnects large and small networks (including LAN) around the globe.

IPS-NCQA

IPS-NCQA is the function that separates worms and viruses coming through various routes. TOE puts block agents into all internal IT entities and uses them to separate the computers that generate abnormal traffics, which can be detected by communications between TOE and agents.

Iteration

The use of a component more than once with varying operations.

Intrusion detection

A function that detects any intrusion activity in the information system.

J

Java

An object-oriented language developed by Sun Microsystems.

K

Kernel

The part of an operating system that is most commonly used. Is resident in main memory, consists of

special processes and process monitors that manage system initialization and interruption, and includes the module that exchanges environments between processes and creates processes.

L

LINUX

Free open-source operating system that can run UNIX of the mainframe computers also on 386 personal computers, which was developed by Linus Torvalds, a student of Helsinki University, Finland in 1991.

M

MAC (Message Authentication Code)

A certain value or part transmitted with a message to verify the validity of parameters such as message content, creator, place of dispatch, etc.

Mail bomb

The sending of a massive amount of email or its sent data to paralyze the mailer program of a target email user or to interfere in receiving emails.

O

Object

An entity within the TSC that contains or receives information and upon which subjects perform operations.

Operation

An activity that uses a component for responding to a certain threat or satisfies a certain security policy. (e.g. repetition, assignment, selection, refinement)

Operation system

Basic software or integrated control program that provides an environment to efficiently execute applications by operating and managing the computer. Its acronym is OS. OS is first loaded when a computer boots and its kernel is resident in main memory.

Organizational security policies

One or more security rules, procedures, practices, or guidelines imposed by an organisation upon its operations.

P

Packet

A group of data used for data transmission in the Internet's network. The packet transmission does not transfer continuous data between two sites but divides the data to transfer into packets with an eligible size. Each packet contains the data control information like data recipient, address, and control marks as well as the fixed size of data.

POP3

Post office protocol version 3 used to receive emails. It is the Internet standard protocol used for the internet mail client to retrieve emails from the mail server.

PP (Protection Profile)

An implementation-independent set of security requirements for a category of TOEs that meet specific consumer needs.

Property

Information and resources protected by the TOE security policies.

Proxy

Originally used in the firewall (intrusion protection system) for Internet security and now used for accessing the proxy server for a Web browser. When a proxy has been set for a Web browser, the URL issued from the Web client is not requested to the corresponding server but proxy server. The requested proxy server sends a request after accessing the corresponding URL server and relays the reply to the client.

R

Raw socket

A protocol that directly uses IP services like ICMP or OSPF.

Refinement

The addition of details to a component.

Role

A predefined set of rules establishing the allowed interactions between a user and the TOE.

S

Script

A series of texts that describes sequential executables in software. A kind of program.

Selection

An operation in the common evaluation criteria. The activity to select one or more items from a component list.

SMTP Proxy

The function that protects the mail server of a sub-network from external attacks and hacking attempts, hides the internal network structure, and makes up for the security settings of the internal mail server.

SOF (Strength of Function)

A qualification of a TOE security function expressing the minimum efforts assumed necessary to defeat its expected security behaviour by directly attacking its underlying security mechanisms.

SOF-medium

A level of the TOE strength of function where analysis shows that the function provides adequate protection against straightforward or intentional breach of TOE security by attackers possessing a moderate attack potential.

SSL (Secure Socket Layer)

A standard protocol used to exchange data between World-wide Web (WWW) browser and WWW server.

State confirming skill

Skills to manage packets according to the packet inflow direction. The corresponding packets are packets flowed into TOE, packets generated by TOE, and packets processing IPS block list, kernel IPS, and administrator authentication.

Static rule

An access rule to the traffic to control that the TOE administrator has directly set.

ST (Security Target)

A set of security requirements and specifications to be used as the basis for evaluation of an identified TOE.

Subject

An entity within the TSC that causes operations to be performed.

T

TCP/IP

A communication protocol developed by US Department of Defense that is a combination of TCP and

IP. TCP/IP is a communication protocol that is currently used on Internet and data exchange with different models in any area is enabled using this protocol.

TOE (Target of Evaluation)

An IT product or system and its associated guidance documentation that is the subject of an evaluation.

Treat agent

Unauthorized users or external IT entities that threaten properties by illegal access, modification, deletion, etc.

TSF (TOE Security Function)

A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.

TSP (TOE Security Policy)

A set of rules that regulate how assets are managed, protected and distributed within a TOE.

TSF Data

Data created by and for the TOE, that might affect the operation of the TOE.

TSC (TSF Scope of Control)

A set of interoperations that can occur in TOE and are under the control of TSP rules.

U

Unicode bug pattern

Unicode (International code encoding scheme that can express world languages in a unified format) failures that are vulnerable to the Web server.

UNIX

An operating system for multiple users developed by AT&T Bell Labs to make a circumstance for promoting the programming research and development.

URL

A protocol that uses logical standard addresses (HTTP, FTP, etc.) to display resources such as files and news groups on Internet. The protocol consists of main computer name & address, directory location of a file, and file name.

USB

Universal serial bus. An external peripheral interface standard for communication between a computer and external peripherals collaboratively suggested by 7 companies like Intel, MS, Compaq, DEC, IBM, Nortel, and NEC. It enables to use same interface for a computer to access peripherals including keyboards, mousse, printers, MODEMs, and speakers with different specifications.

User

All entities such as users and external IT entities that interoperate with TOE from outside.

V

Virtual IP address

Addresses that are used in computers but actually do not exist on computers.

Virus

A set of commands that has a capability to copy itself to the executable areas after transforming the targets.

VRRP (Virtual Router Redundancy Protocol)

An Internet protocol that provides one or more backup routers when the static default router is used on LAN. Even though other alternatives exist, the most common routers distribution is to service packets delivered from the host group on a local area network (LAN) by one router. However, when the router does not work, there is no way to use another router for backup. By VRRP, a virtual IP address is set to the default address. A virtual IP address is used as the master router and another is used for backup. When the master has problem, another virtual IP address is assigned for backup. (The backup router is changed to the master router.) VRRP is also used for load balancing. VRRP is applied to both IPv4 and IPv6.

W

Web vulnerability

Well-known Web weakness in Web server.

1.5. Conventions

This document includes commonly used acronyms. The notations, formats, and conventions follow the common evaluation criteria of the information protection system.

The common evaluation criteria allows the iteration, assignment, selection, and refinement operations performed in the security requirements. Each operation is used in the ST.

Iteration

It is used when the same component is repeated in various operations. The result of repetition is displayed with the repetition number in parenthesis after the component identifier.

Selection

It is used when selecting one or more items provided by the common evaluation criteria of the information protection system when describing requirements. Selections are denoted by *underlined italicized text*.

Refinement

It is used for restricting requirements by adding the items detail. The result of refinement is displayed in **bold text**.

Assignment

It is used to assign a certain value to the non-described parameter. (e.g. password length)

The result of assignment is displayed with a value in bracket. (e.g. [*assignment value*])

Application notes

Application notes are provided to clarify the meaning of requirements, offer information about the selected items, and to define the criteria on “Valid/Invalid” for requirements. An application notes is provided with the corresponding requirement if necessary.

2. TOE Description

TOE is an Intrusion Protection System (IPS) developed to protect the internal network, servers, and services, and information and services between internal network and external networks.

2.1. TOE Environment and Configuration

2.1.1. TOE Environment

The customers who use this TOE can detect and block the intrusion and attack by external real-time network traffics from the end point connected by Internet as shown in the picture.

As in the following diagram, TOE operates on the operating system described in the physical boundaries and scope, and it is used after installing in the company-made hardware. (Refer to physical boundaries and scope)

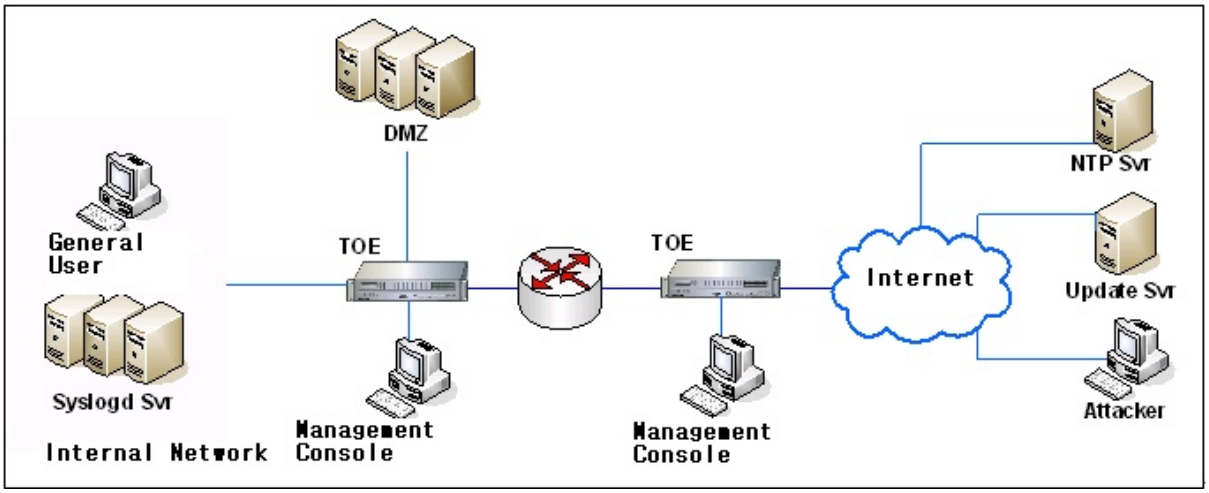

Figure 1 TOE environment

The administrator can manage TOE through the management console. The administrator can also modify the TOE configuration through the management console. For that purpose, the IP address of the administrator PC should have been registered in TOE and is set during initial configuration. The administrator can access to the TOE administrator program to start, stop, and finish the security function. In this case, the administrator channel can be secured using the HTTPS protocol.

The administrator can locate the servers to be disclosed such as intranet server, DNS server, SMTP server, Web

server, and FTP server in DMZ, and can separate the network to protect it. The administrator can manage TOE using the administrator program shown in the picture above and additionally use CLI commands through the management console after connecting the serial cable.

TOE separates internal and external networks with Inline connection so to block the request with subject IP address from the information from an external IT entity, establish policies for all IT entities by adding the IP addresses for all IT entities in internal/external networks. However all policies from outside must not be always allowed.

The TOE administrator updates and manages the vulnerability database managed by TOE using the update server to protect computers from external attacks.

TOE identifies and authenticates internal/external IT entities sending/transmitting information through TOE (the access subject) and then applies the access control rules based on the IP addresses and security attributes (e.g. security lables: 1 to 20) to determine whether each IT entity has access right.

Proxy is the middle gate between server and client and makes TOE kernel to redirect packets in the middle to the proxy and to link sessions between proxy and client and between proxy and server.

Detecting illegal accesses and attacks exhausts the host and network resources or protects computers from hacking attacks that make availability troubles through known vulnerabilities. When any attack is detected by observing the attributes of the information passing TOE, the administrator executes the predefined actions.

The audit logs are stored in TOE and are selectively delivered to the Syslog server outside of TOE. TOE delivers audit logs only to the Syslog server so it is recommended that the TOE administrator use the corresponding function after security is guaranteed.

TOE performs a self test when TOE starts and the authorized administrator requests to show that the abstract machine-related security assumptions properly operate for TSF data and execution files that are stored inside.

The open NTP server that is selectively located in external Internet can be used to adjust the TOE time.

The process watchdog function is used to restart the abnormal processes after checking whether application processes for each TOE security function properly works. This function provides continuous services not to stop the security function that the administrator has defined due to any arbitrary errors.

The SECUI update server that is selectively managed by the company can provide URL update service to block harmful URLs using the Internet content grading service provided by Korea Internet Safety Commission.

2.2. TOE Boundaries and Scope

2.2.1. Physical Boundaries and Scope

The TOE in this security target does not contain hardware and operating system but security function of the identified firmware version shown below.

- NXG IPS 6000 V1.6 firmware: V1.2.5.R
  (On delivery of goods, it is provided to customers after installing the firmware in the operating hardware.)

When installing and operating TOE, the company recommends the following hardware specifications and is not responsible for arbitrary additional hardware installation by customers with no discussion with the company.

Hardware for installing and operating TOE:

Item	Specification	Remarks
CPU	MPC7447 X 7	
Main memory	3.5 GB	
HDD	72 GB	
NIC	8 (1 Gbps x 8) ports + RJ45 type 1 for accessing the management tool	RJ-45 type 4 ports Fiber type 4 ports
Console port	2 ports	DB 9 type
Operating system	SecuiOS V1.1	

※ The console port consists of AUX and console. The console is used to manage CPU-X that is the management CPU out of 7 CPUs and AUX has the console managing CPU-1 that is the first CPU among 6 application handling CPUs.

The management console is the PC that is in charge of operating and managing TOE with administrator program installed. For installing and operating the management console, it is recommended to use the following physical hardware specifications:

Environment for installing and operating TOE:

Management console	Minimum specification	Remarks
CPU	Pentium III 133 MHz or higher	
Main memory	256 MB or higher	

Management console	Minimum specification	Remarks
HDD	40 GB or higher	
NIC	One or more	
Serial	1 EA	DB9 type
Operating system	Windows XP Service Pack 2	
Web browser	Internet Explorer Version 5.5 or higher	Required for patch supporting 128 bit (or longer) SSL

2.2.2. Logical Boundaries and Scope

Function	Description
Blocking non-permit access by organizational policies (Firewall_acc_ctrl)	After identifying and authenticating a user (access subject), TOE applies the arbitrary access control rules for packet filtering using the status-checking technique based on information such as IP addresses, ports, protocols, and flags of the user (place sent) and access object (destination). In addition, the user is controlled by the URL Blocker-based arbitrary access control rules (blocks the URLs set by administrator) and IPS-WEB-based arbitrary access control rules (blocks specific CGI codes, Unicode bugs, etc.) as well as the arbitrary access control rules for packet filtering using the status-checking technique. Such arbitrary access control rules determine whether the user has access privilege or not. After satisfying the abovementioned arbitrary access control rules, the security level is checked through compulsory access control based on the security label (security levels for the object and user are compared). If the user has no appropriate security level for the access object, the user is disconnected.
Blocking information inflow and outflow by organizational policies (Firewall_Proxy)	This function is used to prevent illegal attacks through information leak. The function monitors all user activities and analyzes the incoming packet content to determine whether it violates the security policy or not.

Table 1 Function to protect internal properties and information against illegal accesses

Function	Description
Detection of illegal accesses and attacks (IPS_Detect)	This function detects illegal accesses and attacks against the network and properties protected by TOE.
Reaction to illegal accesses and attacks (IPS_React)	TOE supports the function of taking appropriate measures and responses as defined in the security policies against all attacks detected using the function of detecting illegal accesses and attacks.

Table 2 Blocking illegal accesses and attacks

Function	Description
Registration of TOE users (User_Register)	This function is used to allow access to TOE through the registration of the user account by an authorized administrator. TOE users are authorized administrators for managing TOE. Administrators are registered by the first administrator who has been registered into the TOE system via the IP address registration.
Identification and authentication of TOE users (User_InA)	Once the TOE user registration completes, this function identifies and authenticates the user using the ID and password. Non-active users can be also re-authenticated. The level of password security function used for user identification and authentication satisfies the “Security level—Moderate” and conforms to the tolerance standards for the identification and authentication method (password).

Table 3 Identification and authentication of TOE users

Function	Description
Security function management (Man_Sec_Fun)	This function decides/stops/starts/modifies actions for the security function and limits the rights for the corresponding tasks to the authorized administrators only.

Security attribute management (Man_Sec_attr)	Only the authorized administrator can modify/query/delete the default values for security labels and attributes when performing the access control function and information flow function. This function guarantees that the security attributes have valid values. The default values are restrictively provided so the authorized administrator can specify the selective initial values when the default values are changed.
TSF data management (Man_TSF_Data)	Only the authorized administrator can perform the following: • Audit data control • Backup and recovery • Query and modification to access control rules and information flow control rules • Modification and deletion of identification and authentication data • Time change When the TSF data reaches the limit, the predefined actions are executed.
Security role management (Man_Sec_Role)	TSF maintains the authorized administrator roles and relates the roles between users and authorized administrator.

Table 4 Security management functions

Function	Description
Self test on the security function stability (Secu_Self_Test)	This function tests for data integrity, provides counteractions when the TSF data integrity constrains are violated, and tests whether the TOE system properly works or not.
Safe route between TSFs (TSF_Safe_Channel)	To guarantee communication through the safe routes between TSFs, it supports the encrypted communication using the encoding algorithm. This is the function to protect communication data between TSFs.

Table 5 TSF stability

Function	Description
Creation and protection of audit data (Audit_Gen_Protect)	This function creates logs for TOE system start/end, all activities, and start/end of all processes. It also generates alarms to take proper measures when the audit log memory capacity exceeds and protects the system from illegal modification and deletion.
Audit data review (Audit_Review)	The created audit data can be retrieved through a monitor of the administrator program and the users can summarize data in the format that is easy to view after analyzing and integrating audit logs. The audit data can be reviewed on a predefined cycle: daily and weekly. Only the selected content can also be retrieved.

Table 6 Function to create and protect the security audit data

Function	Description
Process monitoring (Mon_Process)	TOE can provide the security function only when the corresponding process against the security policy works properly.

Table 7 Process monitoring

2.2.3. Evaluation Exception Scope

This evaluation excludes the following items:

- Hardware for operating TOE
- TOE operating system (SecuiOS V1.1)
- SSL protocol that is used for connecting between TOE and administrator program in the management console
- External NTP server that is used to adjust the system's clock to the standard time
- External anti-virus server that helps virus checkup
- SECUI update server for URL update
- SECUI update server for signature update

3. TOE Security Environment

The security environment determines the security problem and its scope to be handled by TOE.

The TOE security environment consists of assumptions that describe the stability of the TOE environment, potential threats to the TOE properties and environment, and organizational security policies such as rules, procedures, habitual practices, and guidelines that TOE shall follow.

3.1. Assumptions

The following assumptions are established under the condition that they exist in the TOE operating environment.

A.Physical security	
The TOE is located in a physically secure environment that only the authorized users can access.	
A.Maintaining security	
When the internal network environment changes according to network configuration change, increase or decrease in host range, increase or decrease in service count, the changed environment and security policies are applied to maintain the same security level as the previous.	
A.Authorized administrator	
A TOE authorized administrator is not a malicious user, well trained for the TOE management functions, and performs the duty for manager guidelines.	
A.Operating system reinforcement	
To guarantee credibility and stability of operating system, TOE removes unnecessary services and measures and reinforces the vulnerability of operating system.	
A.Unique connection point	
When the TOE is installed on the network, it branches the network to external and internal ones so that all communications between internal and external networks are enabled only through the TOE.	
A.Operating system time	**Input by a ST author**
The TOE takes an input for time source from lower operating system and external NTP server.	
A.SSL certificate	**Input by a ST author**
The SSL certificate for access to the TOE uses the private certificate of the TOE itself for secure management.	
A.Secure TOE external server	**Input by a ST author**
NTP server, anti-virus server, and SECUI update server for secure TOE operation outside of the TOE are secure.	

Table 8 TOE assumptions

3.2. Threats

This chapter defines threats to the TOE and the TOE operating environment and describes factors for each threat as the following.
Main resources that the TOE protects are computer resources and network services in the internal network or DMZ. External threat actors attack to illegally access computer resources and remove availability of them.

The threat actors are computer users or IT entities that access internal computers from outside. It is assumed that the threat actors have low level of expert knowledge, resources, and motivation, and have low potential to find vulnerabilities that can be used for evil purpose. Threat actors use the vulnerability information and attacking tools (easily acquired through Internet) for operating systems and application programs, damage the computer resources, and illegally acquire information in computers. The TOE protects resources from the threats using those evident vulnerabilities.

3.2.1. Threats to be Addressed by the TOE

The threat actors mean unauthorized users or external IT entities that are not allowed to access TOE and have the following threats to the TOE.

The following are the threats to the TOE.

T.Spoofing
A threat actor can pretend to be an authorized administrator to access TOE.
T.Failure
When the TOE is in use but any failures happen by external attacks, normal services cannot be provided.
T.Writing errors
When the storage capacity is exceeded, TOE security-related events cannot be written.
T.Inflow of illegal information
The inflow of packets with unauthorized information from external networks can penetrate computers under a network.
T.Illegal access to services
A threat actor can access services that are not allowed to host and interfere with normal services by host.
T.Abnormal packet transfer
A threat actor can send network packets with abnormal structure to cause system failures.

T.Attack on new vulnerabilities
A threat actor can attack computers under a network in TOE or TOE operating environment using new vulnerabilities.
T.Denial of service attack
A threat actor can interfere with using service resources by overuse of service resources for computers under a network in the TOE operating environment.
T.Continuous authentication trials
A threat actor can access TOE by continuously requesting authentications.
T.Bypass access
A threat actor can access TOE after bypassing the TOE security function.
T.Address munging
A threat actor can access the internal network after modifying the sender address to an internal address.
T.TSF data modification without permission
When a threat actor performs a buffer overflow attack on TOE, the TSF data can be changed.

Table 9 Threats to the TOE

3.2.2. Threats to be Addressed by the TOE Operating Environment

Threat factors on TOE operating environment are closely related to the TOE misconfiguration and they are like the following:

TE.Poor management
The TOE can be configured and managed using insecure ways by an authorized administrator.
TE.Delivery and installation
The TOE security can be damaged during delivery and installation of it.

Table 10 Threats to operating environment

3.3. Organizational Security Policies

The organization that operates TOE has its own security policies and an authorized administrator sets security policies using TOE, which are like the following:

P.Audit
To track the responsibility on all security-related activities, it is required to record and maintain security-related events, and to review the recorded data.
P.Secure management
An authorized administrator should manage TOE using the secure way.

Table 11 Organizational security policies

4. Security Objectives

This chapter is subdivided into the TOE security objectives (directly related to the TOE) and Environmental security objectives (IT network domain, nontechnical problems, or procedural measures are mentioned), and in order to prove those objectives, this chapter describes a rationale fit to cope with the assumptions, threats, and organizational security policies that have been defined in the security environment.

4.1. TOE Security objectives

The following security objectives are directly handled by TOE.

O.Availability
When any failures happen by accidental or external attacks, TOE shall provide normal services by maintaining minimum security.
O.Audit
The TOE shall record and maintain security-related events to make it possible to track security-related responsibilities, and provide the means to review the recorded data.
O.Management
The TOE shall provide management means using secure ways in order to efficiently manage TOE.
O.Blocking abnormal packets
The TOE shall block the packets with abnormal structures among TOE passing packets. Application notes: Abnormal packets mean IP address-modified packets, broadcasting packets, looping packets, and packets other than TCP/IP packets defined in Internet standard protocols such as RFC 791 Internet protocol, RFC 792 Internet control messaging protocol, and RFC 793 transmission control protocol.
O.Blocking denial of service attack
The TOE shall block it when attackers are abnormally using computer resources in order for normal users to use network services of the computers that are under protection.
O.Identification
The TOE shall identify all external IT entities under TOE information flow control and users who want to access TOE.
O.Authentication

The TOE shall authenticate the administrator's identity before allowing its access to the TOE. Application notes: When a threat actor makes continuous authentication trials using the identity of the administrator, the authentication data is likely to be acquired. The TOE shall implement the authentication mechanism fit to the security function strength level to block continuous authentication trials.
O.Information flow control
The TOE shall control the inflow of unauthorized information according to security policies. Application notes: This security objectives enables deny-all and allow-all policies that TSF performs. The deny-all policy blocks all packets except for explicitly allowed ones and the allow-all policy allows all packets except for explicitly declined ones.
O.TSF data protection
The TOE shall protect TSF data from unauthorized exposure, modification, and deletion.

Table 12 TOE security objectives

4.2. Environmental Security Objectives

Environmental security policies are handled by IT area or nontechnical/procedural measures.

The following are security policies that TOE handles.

OE.Physical security
The TOE shall be located in a physically secure environment that only the authorized administrator can access.
OE.Security maintenance
When the internal network environment changes according to network configuration change, increase or decrease in host range, increase or decrease in service count, the changed environment and security policies are applied to maintain the same security level as the previous.
OE.Authorized administrator
A TOE authorized administrator shall not be a malicious user, well trained for the TOE management functions, and shall perform the duty for manager guidelines.
OE.Secure management
TOE shall be deployed and installed by secure ways and be configured, managed, and used in a secure manner.
OE.Operating system reinforcement
To guarantee credibility and stability of operating system, TOE shall remove unnecessary services and means and reinforces the vulnerability of operating system.
OE.Unique connection point

When the TOE is installed on the network, it branches the network to external and internal ones so that all communications between internal and external networks are enabled only through the TOE.	
OE.Vulnerability list renewal	
The TOE administrator shall update and manage the vulnerability database managed by TOE using the update server to protect computers from external attacks.	
OE.Operating system time	**Input by a ST author**
The TOE takes an input for time source from the NTP server or lower operating system.	
OE.SSL certificate	**Input by a ST author**
The SSL certificate for access to the TOE uses the private certificate of the TOE itself for secure management.	
OE.Secure TOE external server	**Input by a ST author**
The NTP server and SECUI update server for secure TOE operation outside of the TOE are secure.	

Table 13 Environmental security objectives

5. IT Security Requirements

This chapter describes the functions and assurance requirements that shall be satisfied in the TOE. IT security requirements in the ST consist of function components in Part 2 of the common evaluation criteria and assurance components in Part 3.

5.1. TOE Security Requirements

The ST functional strength for the security requirements is functional strength—medium.

The TOE security requirements of this specification contain the function components in Part 2 of the common evaluation requirements.

The following table shows a summary for function components of the TOE.

Security function class	Security function component	
FAU (Security audit)	FAU_ARP.1	Security alarms
	FAU_GEN.1	Audit data generation
	FAU_GEN.2	User identity association
	FAU_SAA.1	Potential violation analysis
	FAU_SAR.1	Audit review
	FAU_SAR.3	Selectable audit review
	FAU_SEL.1	Selective audit
	FAU_STG.1	Protected audit trail storage
	FAU_STG.3	Action in case of possible audit data loss
	FAU_STG.4	Prevention of audit data loss
FDP (User data protection)	FDP_IFC.1	Subset information flow control
	FDP_IFF.1	Simple security attributes
FIA (Identification and authentication)	FIA_AFL.1	Authentication failure handling
	FIA_ATD.1(1)	User attribute definition (1))
	FIA_ATD.1(2)	User attribute definition (2)
	FIA_UAU.1	Timing of authentication
	FIA_UAU.7	Protected authentication feedback
	FIA_UID.2(1)	User identification before any action (1)
	FIA_UID.2(2)	User identification before any action (2)
FMT (Security management)	FMT_MOF.1	Management of security functions behaviour
	FMT_MSA.1	Management of security attributes
	FMT_MSA.3	Static attribute initialization
	FMT_MTD.1 (1)	Management of TSF data (1)
	FMT_MTD.1 (2)	Management of TSF data (2)
	FMT_MTD.1 (3)	Management of TSF data (3)
	FMT_MTD.1 (4)	Management of TSF data (4)
	FMT_MTD.1 (5)	Management of TSF data (5)
	FMT_MTD.1 (6)	Management of TSF data (6)
	FMT_MTD.2	Management of limits on TSF data
	FMT_SMF.1	Specification of management functions
	FMT_SMR.1	Security roles
FPT	FPT_AMT.1	Abstract machine testing

	FPT_FLS.1	Failure with preservation of secure state
	FPT_RVM.1	Non-bypassability of the TSP
	FPT_SEP.1	TSF domain separation
	FPT_STM.1	Reliable time stamps
	FPT_TST.1	TSF testing
FRU (Resource utilization)	FRU_FLT.1	Degraded fault tolerance
	FRU_RSA.1	Maximum quotas
FTA (Access to the TOE)	FTA_SSL.1	TSF-initiated session locking
	FTA_SSL.3	TSF-initiated termination
FTP (Secure route/channel)	FTP_ITC.1	Inter-TSF trusted channel

Table 14 Function components of the TOE

5.1.1. Security Audit

5.1.1.1. FAU_ARP Security Audit automatic Response

FAU_ARP.1 Security alarms

Hierarchical to: No other components.

Dependencies: FAU_SAA.1 Potential violation analysis Potential violation analysis

FAU_ARP.1.1 TSF shall take [notify authorized administrators of this through the administrator program and send this to email addresses registered by authorized administrators] upon detection of a potential security violation.

5.1.1.2. FAU_GEN Security Audit data Generation

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 shut-down of the audit functions;
b) All auditable events for the *minimum* level of audit; and
c) [Refer to "*Auditable events*" in Table 15 Auditable events
d)]

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, 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 Table 15 Auditable events]

Function component	Auditable event	Content of additional audit log
FAU_ARP.1	Actions taken due to imminent security violations	Recipient identity for counteraction
FAU_SAA.1	Enabling and disabling of any of the analysis mechnisms; and Automated responses perfomed by the tool	-
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating.	-
FDP_IFF.1	Decisions to permit requested information flows.	Object identification information
FIA_AFL.1	the reaching of the threshold for the unsuccessful authentication attempts and the actions (e.g. disabling of a terminal) taken and the subsequent, if appropriate, restoration to the normal state(e.g. re-enabling of a terminal)	-
FIA_UAU.1	Unsuccessful use of the user identification mechanism, including the user identity provided	
FIA_UID.2	Failure in using the user identification mechanism and the provided user identity provided	-
FMT_SMF.1	Use of management functions	-
FMT_SMR.1	Modifications to the group of users that are part of a role	-
FPT_STM.1	Changes to the time	
FRU_FLT.1	All failure detected by the TSF.	
FRU_RSA.1	Rejection of allocation operation due to resource limits.	-
FTA_SSL.1	Locking of an interactive session by the session locking mechanism.	-
FTA_SSL.3	Termination of an interactive session by the session locking mechanism.	-
FTP_ITC.1	Failure of the trusted channel fuctions.	-

Table 15 Auditable events

FAU_GEN.2 User identity association

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit data generation

FIA_UID.1 Timing of Identification

FAU_GEN.2.1 The TSF shall be able to associate each auditable event with the identity of the user that caused the event.

5.1.1.3. FAU_SAA Security Audit Analysis

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 TSP.

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

a) Accumulation or combination of [identification and authentication security policy violation, access control rule violation, decline of excessive packets, excessive mails, excessive logs, too low traffics, excessive traffics from external networks, interface shut-down, excessive sessions, excessive CPU usage, excessive memory usage, file system alert, virtual IP address alert, main system problem alert, alert for abnormal process termination and duplicate execution, duplicate IP addresses, more than 100 dynamic rules on the same hash bucket, and data integrity violation] known to indicate a potential security violation;

b) [Miscellaneous rules: no rule]

5.1.1.4. FAU_SAR Security Audit Review

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 [the authorized administrators] with the capability to read [all audit data] from the audit records.

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

FAU_SAR.3 Selectable audit review

Hierarchical to: No other components.
Dependencies: FAU_SAR.1 Audit review

FAU_SAR.3.1 The TSF shall provide the ability to perform *searches, sorting* of audit data based on [subject identity, target object, date and time of the event, event type, event importance, and event results].

5.1.1.5. FAU_SEL Security audit Event Selection

FAU_SEL.1 Selective audit

Hierarchical to: No other components.
Dependencies: FAU_GEN.1 Audit data generation
FMT_MTD.1 TSF data management

FAU_SEL.1.1 The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

a) Event type

b) [IPS rule item]

5.1.1.6. FAU_STG Security Audit Event Storage

FAU_STG.1 Protected audit trail storage

Hierarchical to: No other components.
Dependencies: FAU_GEN.1 Audit data generation

FAU_STG.1.1 The TSF shall protect the stored audit records from unauthorized deletion.

FAU_STG.1.2 The TSF shall be able to *prevent* unauthorized modifications to the stored audit records in the audit trail.

FAU_STG.3 Action in case of possible audit data loss

Hierarchical to: No other components.

Dependencies: FAU_STG.1 Protected audit trail storage

FAU_STG.3.1 The TSF shall [take measures for the administrator to respond by emailing or sending information through the alarm window on the administrator program] if the audit trail exceeds [99% of the audit storage capacity (an authorized administrator can change it)].

FAU_STG.4 Prevention of audit data loss

Hierarchical to: FAU_STG.3

Dependencies: FAU_STG.1 Protected audit trail storage

FAU_STG.4.1 The TSF shall *prevent auditable events, except those taken by the authorized user with special rights* and [no action] if the audit trail is full.

5.1.2. User Data Protection

5.1.2.1. FDP_IFC Information Flow Control Policy

FDP_IFC.1 Subset information flow control

Hierarchical to: No other components.

Dependencies: FDP_IFF.1 Simple security attributes

FDP_IFC.1.1 The TSF shall enforce the [NXG IPS policy] on [operations that cause controlled information to flow to and from controlled subjects covered by the SFP].

a) [Subject: Unauthorized external IT entities on the sender side
b) Information: Traffic that is sent from the subject to any destination through the TOE
c) Operation: Pass it when any allowed rules the administrator has defined exist after default decline and block it when any blocking rules the administrator has defined exist].

5.1.2.2. FDP_IFF Information Flow Control Functions

FDP_IFF.1 Simple security attributes

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control

FMT_MSA.3 Static attribute initialization

FDP_IFF.1.1 The TSF shall enforce the [NXG IPS policy] based on the following types of subject and information security attributes: [list of subjects and information controlled under the indicated SFP, and for each, the security attributes].

[

a) Subject list: IT entities on internal/external networks through the TOE that sends or receives information

Subject security attributes: IP address information and security levels (1 to 20)

b) Information list: Network packets that are sent through the TOE

Information security attributes: Destination URI (uniform resource identifier), time, header and data information of a packet

]

FDP_IFF.1.2 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [Rules]

[

a) Information flow is allowed only for the allowed ones after comparing between the security attributes of the network packet information that passes through the TOE and the NXG IPS policy that an authorized administrator has defined.

b) Information flow is allowed when the security level of From is higher than the one of To or the packets are the same.

]

FDP_IFF.1.3 The TSF shall enforce the [Deny-all rule in case of the default policy rule].

FDP_IFF.1.4 The TSF shall provide the following [conversion, blocking, and predefined exceptional handlings of the anti-spam that an authorized administrator has defined in line with the internal anti-virus server].

FDP_IFF.1.5 The TSF shall explicitly authorize an information flow based on the following rules: [no rule].

FDP_IFF.1.6 The TSF shall explicitly deny an information flow based on the following rules:

[

a) The TOE shall block the connection request from an external IT entity when its information has a

subject IP address of the internal network.

b) The TOE shall block the connection request from an internal IT entity when its information has a subject IP address of an external network.

c) The TOE shall block the connection request from an external IT entity when its information has a subject IP address for broadcasting.

d) The TOE shall block the connection request from an external IT entity when its information has a subject IP address for looping.

e) The TOE shall block the connection request from an external IT entity when its information has an abnormal packet structure.

f) The TOE shall block the connection request when the information is in the IPS selection item list (URL blocking list, Web vulnerability blocking list, abnormal traffic/abnormal protocol/general hacking blocking list).

]

5.1.3. Identification and Authentication

5.1.3.1. FIA_AFL Authentication Failures

FIA_AFL.1 Authentication failure processing

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1 The TSF shall detect when *administrator configurable positive integer within [3 to 10]* unsuccessful authentication attempts occur related to [authentication trials].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [perform the actions: prevent the corresponding user from being authenticated and Records audit events until an authorized administrator take an action].

5.1.3.2. FIA_ATD User Attribute Definition

FIA_ATD.1 (1) User attribute definition (1)

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual **IT entity**. [Security attributes].

a) IP address

b) {no value}

Application notes: The security requirements function is used to identify unauthorized external users who communicate with an internal computer being protected through the TOE. Using the function, the TOE can identify external IT entities, records audit events of external IT entities, and later track the responsibility for any failure.

FIA_ATD.1 (2) User attribute definition (2)

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual **administrator**: [Security attributes].

a) Identifier

b) { no value }

Application notes: The security requirements function is used to identify administrators who want to manage by interoperating with TOE after accessing TOE. Using this function, the TOE identifies an authorized administrator and requests authentication after identifying it.

5.1.3.3. FIA_UAU User Authentication

FIA_UAU.1 Timing of authentication

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FIA_UAU.1.1 The TSF shall allow [download and installation of Welf File Converter, SNMP MIB FILE, JRE 1.4.2_05, NXG Manager] on behalf of the **administrator** to be performed before the **administrator** is authenticated.

FIA_UAU.1.2 The TSF shall require each **administrator** to be successfully authenticated before allowing any other TSF-mediated actions except for actions mentioned in FIA_UAU.1.1.

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 [constructing the input character with dots] to the **administrator** while the authentication is in progress.

5.1.3.4. FIA_UID User Identification

FIA_UID.2 (1) User identification before any action (1)

Hierarchical to: FIA_UID.1 Timing of identification

Dependencies: No dependencies.

FIA_UID.2.1 The TSF shall require each **IT entity** to identify itself before allowing any other TSF-mediated actions on behalf of that user.

Application notes: TOE users are identified by administrators and IT entities and this component requires identification of IT entities.

FIA_UID.2 (2) User identification before any action (2)

Hierarchical to: FIA_UID.1 Timing of identification

Dependencies: No dependencies.

FIA_UID.2.1 The TSF shall require each **administrator** to identify itself before allowing any other TSF-mediated actions on behalf of that user.

Application notes: TOE users are identified by administrators and IT entities and this component requires identification of the administrator.

5.1.4. Security Management

5.1.4.1. FMT_MOF Management of Functions in TSF

FMT_MOF.1 Management of security functions behavior

Hierarchical to: No other components.

Dependencies: FMT_SMF.1 Specification of Management Functions

FMT_SMR.1 Security roles

FMT_MOF.1.1 The TSF shall restrict the ability to [determine the behavior of, disable, enable, modify the behavior of] the following functions to [authorized administrators].

[

List of functions	Capability
Audit data statistics processing	Determine and enable the behavior
Maintaining the counteraction to take on possible audit storage failure	Determine, disable, and modify the behavior
Maintaining the counteraction to take on audit storage failure	Determine, disable, enable, and modify the behavior
Backup and recovery of audit trails	Determine, disable, enable, and modify the behavior
Selection of an item in audit trails	Determine, disable, enable, and modify the behavior
Maintaining the threshold value on audit trails	Determine and modify the behavior
Managing the authentication data by the administrator	Disable and enable the behavior
Specification of maximum resource limit for subject by the administrator	Determine, disable, enable, and modify the behavior
Management of security attributes for the NXG IPS policy that has been used to determine explicit access	Determine, disable, enable, and modify the behavior

Management of the counteractions against security alarms	Disable, enable, and modify the behavior
Maintaining the user group with the read right for the security audit review	Determine and modify the behavior
Management of security attributes	Determine the behavior
Management of allowable range of values used to verify the confidential information	Determine and modify the behavior
Management of security attributes for users	Determine the behavior
Management of the list of behaviors that can be performed before authenticating users	Modify the behavior
Specification of the user inactive time when the user locking happens	Determine and modify the behavior
Specification of the default user inactive time when the interoperation session is closed	Determine and modify the behavior
Time management	Determine and modify the behavior
System shut-down and rebooting	Determine and enable the behavior
Identification and authentication data management	Determine and modify the behavior
Management of the limit of failed authentication trials	Modify the behavior
Configuration of behaviors requesting the secure channel	Modify the behavior
Management of user groups that assume roles	Determine, enable, and modify the behavior
Management of counteractions to take in case of authentication failure	Determine and modify the behavior
Specification of the default user inactive time when any locking happens	Modify the behavior
Maintaining rules for analyzing potential violations	Determine, disable, enable, and modify the behavior
Manual test for abstract machines	Enable the behavior
Management of anti-spam settings	Determine, disable,

	enable, and modify the behavior
IPS signature update management	Enable the behavior
Management of TSF data limit and role groups that can interoperate	Determine and modify the behavior
Management of TSF data limits	Modify the behavior
Management of the administrator group that can interoperate with TSF data	Determine and modify the behavior

]

5.1.4.2. FMT_MSA Management of Security Attributes

FMT_MSA.1 Management of security attributes

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control or
FDP_IFC.1 subset information flow control]
FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

FMT_MSA.1.1 The TSF shall enforce the [NXG IPS policy] to restrict the ability to *query, modify, delete, {create}* the following security attributes to [the authorized administrators].

[

List of security attributes	Operation list
Security label (1 to 20)	Modify
Group that the subject belongs to	Query, modify, delete
Destination URI	Modify, delete, create
Time	Modify
IP address	Modify, create
Port number	Modify, create
Data information	Modify, delete, create

]

FMT_MSA.3 Static attribute initialization

Hierarchical to: No other components.

Dependencies: FMT_MSA.1 Management of security attributes

FMT_SMR.1 Security roles

FMT_MSA.3.1 The TSF shall enforce the [NXG IPS policy] to provide *restrictive* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the [authorized administrators] to specify alternative initial values to override the default values when an object or information is created.

5.1.4.3. FMT_MTD Management of TSF data

FMT_MTD.1 (1) Management of TSF data (1)

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 *process statistics of* the [audit data] to [authorized administrators].

FMT_MTD.1 (2) Management of TSF data (2)

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 *back up on a semi-permanent secondary memory unit* the [important files comprising TOE] to [the authorized administrators].

FMT_MTD.1 (3) Management of TSF data (3)

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 *modify, delete, {create}* the [identification and authentication data] to [the authorized administrators].

FMT_MTD.1 (4) Management of TSF data (4)

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 *modify, delete, {create}* the [anti-spam settings] to [the authorized administrators].

FMT_MTD.1 (5) Management of TSF data (5)

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 *change* the [time] to [the authorized administrators].

FMT_MTD.1 (6) Management of TSF data (6)

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 *update* the [IPS signatures] to [the authorized administrators].

FMT_MTD.2 Management of limits on TSF data

Hierarchical to: No other components.

Dependencies: FMT_MTD.1 Management of TSF data

FMT_SMR.1 Security roles

FMT_MTD.2.1 The TSF shall restrict the specification of the limits for [audit storage capacity, the number of failed authentication trials, self test interval, UDP flooding, SYN flooding, scan attack, and ping flooding] to [the authorized administrators].

FMT_MTD.2.2 The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: [counteractions specified in FAU_STG.3 and FIA_AFL.1, the self test specified in FPT_TST.1, detection, blocking and recording audit log for each attack].

5.1.4.4. FMT_SMF Specification of Management Functions

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 security management functions: [List of security management functions provided by TSF]

[Security functions provided by TSF:

- Items that are specified in the FMT_MOF.1 TSF function management
- Items specified in the FMT_MSA.1 Management of security attributes
- Items specified in the FMT_MTD.1 (1) Data management
- Items specified in the FMT_MTD.1 (2) Data management
- Items specified in the FMT_MTD.1 (3) Data management
- Items specified in the FMT_MTD.1 (4) Data management
- Items specified in the FMT_MTD.1 (5) Data management
- Items specified in the FMT_MTD.1 (6) Data management
- Items specified in the FMT_MTD.2 Management of TSF data limits
- Items specified in the FMT_SMR.1 Security roles

]

5.1.4.5. FMT_SMR Security Management Roles

FMT_SMR.1 Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Identification.

FMT_SMR.1.1 The TSF shall maintain the roles **[authorized administrator and access levels for each administrator group (level 1, 2, 3)**].

FMT_SMR.1.2 The TSF shall be able to associate users with the **authorized administrator** role.

5.1.5. Protection of the TSF

5.1.5.1. FPT_AMT Underlying Abstract Machine Test

FPT_AMT.1 Abstract machine testing

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_AMT.1.1 The TSF shall run a suite of tests *during initial start-up, periodically during normal operation, at the request of an authorized user* to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF.

5.1.5.2. FPT_FLS Fail Secure

FPT_FLS.1 Failure with preservation of secure state

Hierarchical to: No other components.

Dependencies: ADV_SPM.1 Informal TOE security policy model.

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [Abnormal termination of the application level daemon]

5.1.5.3. FPT_RVM Reference Meditation

FPT_RVM.1 Non-bypassability of the TSP

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_RVM.1.1 The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

5.1.5.4. FPT_SEP Domain Separation

FPT_SEP.1 TSF domain separation

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_SEP.1.1 The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

FPT_SEP.1.2 The TSF shall enforce separation between the security domains of subjects in the TSC.

5.1.5.5. FPT_STM Time Stamps

FPT_STM.1 Reliable time stamps

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps for its own use.

Application notes: The TOE security requirements are used to provide the time stamp function that guarantees sequential generation of audit data in connection with the security audit function. Accordingly the abovementioned security requirements are not implemented for the TOE security requirements and TOE can use the time provided in the TOE environment.

5.1.5.6. FPT_TST TSF Self test

FPT_TST.1 TSF testing

Hierarchical to: No other components.

Dependencies: FPT_AMT.1 Abstract machine testing.

FPT_TST.1.1 The TSF shall run a suite of self tests *during initial start-up, periodically during normal operation, at the request of the authorized user, at the conditions* [*no value*]] to demonstrate the correct operation of the [*TSF*]

FPT_TST.1.2 The TSF shall provide **authorized administrators** with the capability to verify the integrity of [*TSF data except for audit trails*].

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

5.1.6. Resource Utilization

5.1.6.1. FRU_FLT Fault Tolerance

FRU_FLT.1 Degraded fault tolerance

Hierarchical to: No other components.

Dependencies: FPT_FLS.1 Failure with preservation of secure state

FRU_FLT.1.1 The TSF shall ensure the operation of [redundant power supply] when [a hardware power module failure] occurs.

Application notes: This function is used to guarantee that users can utilize network services even though TOE failures occur. Accordingly a developer shall implement types of the TOE failures and counter functions in order for users to utilize minimum network services even in TOE failures, and specify them in the ST.

5.1.6.2. FRU_RSA Resource Allocation

FRU_RSA.1 Maximum quotas

Hierarchical to: No other components.

Dependencies: No dependencies.

FRU_RSA.1.1 The TSF shall enforce maximum quotas of the following resources: [expression of the transfer layer] that *individual **IT entity*** can use *simultaneously*.

Application notes: Expression of the transfer layer means the connection of SYN packets of TCP. The connection of SYN packets causes the semiconnection state and enables the SYN attack, which interferes with connection services of normal users due to exhaustion of connection table resources. The attacking subject is an IT entity. This function blocks the service declining attacks by the protocol stack of TCP, according to the IT entity identifier.

5.1.7. TOE Access

5.1.7.1. FTA_SSL Session Locking

FTA_SSL.1 TSF-initiated session locking

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FTA_SSL.1.1 The TSF shall lock an **authorized administrator** session after [10 minutes, time interval of the **authorized administrator** inactivity] by:

a) clearing or overwriting display devices, making the current contents unreadable;

b) disabling any activity of the **authorized administrator**'s data access/display devices other than unlocking the session.

FTA_SSL.1.2 The TSF shall require the following events to occur prior to unlocking the session: [re-authentication].

FTA_SSL.3 TSF-initiated termination

Hierarchical to: No other components.

Dependencies: No dependencies.

FTA_SSL.3.1 The TSF shall terminate an interactive session after a [10 minutes, time interval of ***IT entity*** inactivity].

Application notes: The TSF terminates the session when the interconnection between IT entities that interoperates through the TOE is inactive for a certain period of time because the TSF mediates the connection

of internal and external networks.

5.1.8. Trusted Path/Channels

5.1.8.1. FTP_ITC.1 Inter-TSF Trusted Channel

FTP_ITC.1	Inter-TSF trusted channel

Hierarchical to: No other components.
Dependencies: No dependencies.

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 The TSF shall permit *the TSF* to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [access to the administrator program].

Application notes: This component is the requirement to implement a secure channel when the communications between trusted IT products outside of the TOE are performed. A secure channel is required when an administrator remotely accesses TOE, or TOE communicates with external vulnerability data servers other than the administrator locally accesses TOE. The TOE provides a secure channel by connecting the SSL session that is provided in IT environment.

5.2. TOE Assurance Requirements

The TOE assurance requirements of the ST is based on the assurance component in Part 3 of the common evaluation criteria and its certificate level is **EAL4**. [Table 16] shows the summary assurance components list.

Assurance class	Assurance component		Remarks
Configuration management	ACM_AUT.1	Partial CM automation	
	ACM_CAP.4	Generation support and acceptance procedures	
	ACM_SCP.2	Problem tracking CM coverage	
Delivery and operation	ADO_DEL.2	Detection of modification	
	ADO_IGS.1	Installation, generation, and start-up procedures	
Development	ADV_FSP.2	Fully defined external interfaces	
	ADV_HLD.2	Security enforcing high-level design	
	ADV_IMP.1	Expression of implementation for partial TSF	
	ADV_LLD.1	Descriptive low-level design	
	ADV_RCR.1	Informal correspondence demonstration	
	ADV_SPM.1	Informal TOE security policy model	
Administrator guidance	AGD_ADM.1	Administrator guidance	
	AGD_USR.1	User guidance	
Development security	ALC_DVS.1	Identification of security measures	
	ALC_LCD.1	Lifecycle model defined by developers	
	ALC_TAT.1	Well-defined development tools	
Coverage	ATE_COV.2	Analysis of coverage	
	ATE_DPT.1	Basic design test	
	ATE_FUN.1	Functional testing	
	ATE_IND.2	Independent testing: sample	
Vulnerability evaluation	AVA_MSU.2	Verification of administrator guidance analysis	
	AVA_SOF.1	Evaluation for strength of the TOE security functions	
	AVA_VLA.2	Independent vulnerability analysis	

Table 16 Assurance component: EAL4

5.2.1. Configuration Management

ACM_AUT.1 Partial CM automation

Developer action elements:

ACM_AUT.1.1D	The developer shall use a CM system.
ACM_AUT.1.2D	The developer shall provide a CM system.

Content and presentation of evidence elements:

ACM_AUT.1.1C	The CM system shall provide an automated means by which only authorized changes are made to the TOE implementation representation.
ACM_AUT.1.2C	The CM system shall provide an automated means to support the generation of the TOE.
ACM_AUT.1.3C	The CM plan shall describe the automated tools used in the CM system.
ACM_AUT.1.4C	The CM plan shall describe how the automated tools are used in the CM system.

Evaluator action elements:

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

ACM_CAP.4 Generation support and acceptance procedures

Developer action elements:

ACM_CAP.4.1D	The developer shall provide a reference for the TOE.
ACM_CAP.4.2D	The developer shall use a CM system.
ACM_CAP.4.3D	The developer shall provide CM documentation.

Content and presentation of evidence elements:

ACM_CAP.4.1C	The reference for the TOE shall be unique to each version of the TOE.
ACM_CAP.4.2C	The TOE shall be labeled with its reference.
ACM_CAP.4.3C	The CM documentation shall include a configuration list, a CM plan, and an acceptance plan.
ACM_CAP.4.4C	The configuration list shall uniquely identify all configuration items that comprise the

	TOE.
ACM_CAP.4.5C	The configuration list shall describe the configuration items that comprise the TOE.
ACM_CAP.4.6C	The CM documentation shall describe the method used to uniquely identify the configuration items that comprise the TOE.
ACM_CAP.4.7C	The CM system shall uniquely identify all configuration items that comprise the TOE.
ACM_CAP.4.8C	The CM plan shall describe how the CM system is used.
ACM_CAP.4.9C	The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.
ACM_CAP.4.10C	The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.
ACM_CAP.4.11C	The CM system shall provide measures such that only authorized changes are made to the configuration items.
ACM_CAP.4.12C	The CM system shall support the generation of the TOE.
ACM_CAP.4.13C	The acceptance plan shall describe the procedures used to accept modified or newly created configuration items as part of the TOE.

Evaluator action elements:

ACM_CAP.4.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ACM_SCP.2 Problem tracking CM coverage

Developer action elements:

ACM_SCP.2.1D	The developer shall provide a list of configuration items for the TOE.

Content and presentation of evidence elements:

ACM_SCP.2.1C	The list of configuration items shall include the following: implementation representation; security flaws; and the evaluation evidence required by the assurance components in the ST.

Evaluator action elements:

ACM_SCP.2.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.2. Delivery and Operation

ADO_DEL.2 Detection of modification

Developer action elements:

ADO_DEL.2.1D	The developer shall document procedures for delivery of the TOE or parts of it to the user.
ADO_DEL.2.2D	The developer shall use the delivery procedures.

Content and presentation of evidence elements:

ADO_DEL.2.1C	The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site.
ADO_DEL.2.2C	The delivery documentation shall describe how the various procedures and technical measures provide for the detection of modifications, or any discrepancy between the developer's master copy and the version received at the user site.
ADO_DEL.2.3C	The delivery documentation shall describe how the various procedures allow detection of attempts to masquerade as the developer, even in cases in which the developer has sent nothing to the user's site.

Evaluator action elements:

ADO_DEL.2.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADO_IGS.1 Installation, generation, and start-up procedures

Developer action elements:

ADO_IGS.1.1D	The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE.

Content and presentation of evidence elements:

ADO_IGS.1.1C	The installation, generation and start-up documentation shall describe all the steps necessary for secure installation, generation and start-up of the TOE.

Evaluator action elements:

ADO_IGS.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADO_IGS.1.2E	The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration.

5.2.3. Development

ADV_FSP.2 Fully defined external interfaces

Developer action elements:

ADV_FSP.2.1D	The developer shall provide a functional specification.

Content and presentation of evidence elements:

ADV_FSP.2.1C	The functional specification shall describe the TSF and its external interfaces using an informal style.
ADV_FSP.2.2C	The functional specification shall be internally consistent.
ADV_FSP.2.3C	The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing complete details of all effects, exceptions and error messages.
ADV_FSP.2.4C	The functional specification shall completely represent the TSF.
ADV_FSP.2.5C	The functional specification shall include rationale that the TSF is completely represented.

Evaluator action elements:

ADV_FSP.2.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_FSP.2.2E	The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional requirements.

ADV_HLD.2 Security enforcing high-level design

Developer action elements:

ADV_HLD.2.1D	The developer shall provide the high-level design of the TSF.

Content and presentation of evidence elements:

ADV_HLD.2.1C	The presentation of the high-level design shall be informal.
ADV_HLD.2.2C	The high-level design shall be internally consistent.
ADV_HLD.2.3C	The high-level design shall describe the structure of the TSF in terms of subsystems.
ADV_HLD.2.4C	The high-level design shall describe the security functionality provided by each subsystem of the TSF.
ADV_HLD.2.5C	The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software.
ADV_HLD.2.6C	The high-level design shall identify all interfaces to the subsystems of the TSF.
ADV_HLD.2.7C	The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.
ADV_HLD.2.8C	The high-level design shall describe the purpose and method of use of all interfaces to the subsystems of the TSF, providing details of effects, exceptions and error messages, as appropriate.
ADV_HLD.2.9C	The high-level design shall describe the separation of the TOE into TSP-enforcing and other subsystems.

Evaluator action elements:

ADV_HLD.2.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_HLD.2.2E	The evaluator shall determine that the high-level design is an accurate and complete instantiation of the TOE security functional requirements.

ADV_IMP.1 Subset of the implementation of the TSF

Developer action elements:

ADV_IMP.1.1D	The developer shall provide the implementation representation for a selected subset of the TSF.

Content and presentation of evidence elements:

ADV_IMP.1.1C	The implementation representation shall unambiguously define the TSF to a level of detail such that the TSF can be generated without further design decisions.

ADV_IMP.1.2C	The implementation representation shall be internally consistent.

Evaluator action elements:

ADV_IMP.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_IMP.1.1E	The evaluator shall confirm that the least abstract TSF representation provided is an accurate and complete instantiation of the TOE security functional requrements.

ADV_LLD.1 Descriptive low-level design

Developer action elements:

ADV_LLD.1.1D	The developer shall provide the low-level design of the TSF.

Content and presentation of evidence elements:

ADV_LLD.1.1C	The presentation of the low-level design shall be informal.
ADV_LLD.1.2C	The low-level design shall be internally consistent.
ADV_LLD.1.3C	The low-level design shall describe the TSF in terms of modules.
ADV_LLD.1.4C	The low-level design shall describe the purpose of each module.
ADV_LLD.1.5C	The low-level design shall define the interrelationships between the modules in terms of provided security functionality and dependencies on other modules.
ADV_LLD.1.6C	The low-level design shall describe how each TSP-enforcing function is provided.
ADV_LLD.1.7C	The low-level design shall identify all interfaces to the modules of the TSF.
ADV_LLD.1.8C	The low-level design shall identify which of the interfaces to the modules of the TSF are externally visible.
ADV_LLD.1.9C	The low-level design shall describe the purpose and method of use of all interfaces to the modules of the TSF, providing details of effects, exceptions and error messages, as appropriate.
ADV_LLD.1.10C	The low-level design shall describe the separation of the TOE into TSP-enforcing and other modules.

Evaluator action elements:

ADV_LLD.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_LLD.1.2E	The evaluator shall determine that the low-level design is an accurate and complete instantiation of the TOE security functional requirements.

ADV_RCR.1 Informal correspondence demonstration

Developer action elements:

ADV_RCR.1.1D	The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.

Content and presentation of evidence elements

ADV_RCR.1.1C For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.

Evaluator action elements:

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

ADV_SPM.1 Informal TOE security policy model

Developer action elements:

ADV_SPM.1.1D The developer shall provide a TSP model.

ADV_SPM.1.2D The developer shall demonstrate correspondence between the functional specification and the TSP model.

Content and presentation of evidence elements:

ADV_SPM.1.1C The TSP model shall be informal.

ADV_SPM.1.2C The TSP model shall describe the rules and characteristics of all policies of the TSP that can be modeled.

ADV_SPM.1.3C The TSP model shall include a rationale that demonstrates that it is consistent and complete with respect to all policies of the TSP that can be modeled.

ADV_SPM.1.4C The demonstration of correspondence between the TSP model and the functional specification shall show that all of the security functions in the functional specification are consistent and complete with respect to the TSP model.

Evaluator action elements:

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

5.2.4. Guidance Documents

AGD_ADM.1 Administrator guidance

Developer action elements:

AGD_ADM.1.1D The developer shall provide administrator guidance addressed to system administrative personnel.

Content and presentation of evidence elements:

AGD_ADM.1.1C The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE.

AGD_ADM.1.2C The administrator guidance shall describe how to administer the TOE in a secure manner.

AGD_ADM.1.3C The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment.

AGD_ADM.1.4C The administrator guidance shall describe all assumptions regarding user behavior that are relevant to secure operation of the TOE.

AGD_ADM.1.5C The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate.

AGD_ADM.1.6C The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_ADM.1.7C The administrator guidance shall be consistent with all other documentation supplied for evaluation.

AGD_ADM.1.8C The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator.

Evaluator action elements:

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

AGD_USR.1 User guidance

Developer action elements:

AGD_USR.1.1D The developer shall provide user guidance.

Content and presentation of evidence elements:

AGD_USR.1.1C	The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE.
AGD_USR.1.2C	The user guidance shall describe the use of user-accessible security functions provided by the TOE.
AGD_USR.1.3C	The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment.
AGD_USR.1.4C	The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behavior found in the statement of the TOE security environment.
AGD_USR.1.5C	The user guidance shall be consistent with all other documentation supplied for evaluation.
AGD_USR.1.6C	The user guidance shall describe all security requirements for the IT environment that are relevant to the user.

Evaluator action elements:

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

5.2.5. Life Cycle Support

ALC_DVS.1 Identification of security measures

Developer action elements:

ALC_DVS.1.1D	The developer shall produce development security documentation.

Content and presentation of evidence elements:

ALC_DVS.1.1C	The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.
ALC_DVS.1.2C	The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE.

Evaluator action elements:

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

ALC_DVS.1.2E The evaluator shall confirm that the security measures are being applied.

ALC_LCD.1 Developer defined life-cycle model

Developer action elements:

ALC_LCD.1.1D The developer shall establish a life-cycle model to be used in the development and maintenance of the TOE.

ALC_LCD.1.2D The developer shall provide life-cycle definition documentation.

Content and presentation of evidence elements:

ALC_LCD.1.1C The life-cycle definition documentation shall describe the model used to develop and maintain the TOE.

ALC_LCD.1.2C The life-cycle model shall provide for the necessary control over the development and maintenance of the TOE.

Evaluation action elements:

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

ALC_TAT.1 Well-defined development tools

Developer action elements:

ALC_TAT.1.1D The developer shall identify the development tools being used for the TOE.

ALC_TAT.1.2D The developer shall document the selected implementation-dependent options of the development tools.

Content and presentation of evidence elements:

ALC_TAT.1.1C All development tools used for implementation shall be well-defined.

ALC_TAT.1.2C The documentation of the development tools shall unambiguously define the meaning of all statements used in the implementation.

ALC_TAT.1.3C The documentation of the development tools shall unambiguously define the meaning of all implementation-dependent options.

Evaluation action elements:

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

5.2.6. Tests

ATE_COV.2 Analysis of coverage

Developer action elements:

ATE_COV.2.1D The developer shall provide an analysis of the test coverage.

Content and presentation of evidence elements:

ATE_COV.2.1C The analysis of the test coverage shall demonstrate the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification.

ATE_COV.2.2C The analysis of the test coverage shall demonstrate that the correspondence between the TSF as described in the functional specification and the tests identified in the test documentation is complete.

Evaluator action elements:

ATE_COV.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_DPT.1 Testing: high-level design

Developer action elements:

ATE_DPT.1.1D The developer shall provide the analysis of the depth of testing.

Content and presentation of evidence elements:

ATE_DPT.1.1C The depth analysis shall demonstrate that the tests identified in the test documentation are sufficient to demonstrate that the TSF operates

In accordance with its high-level design.

Evaluator action elements:

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

of evidence.

ATE_FUN.1 Functional testing

Developer action elements:

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

ATE_FUN.1.2D The developer shall provide test documentation.

Content and presentation of evidence elements:

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

ATE_FUN.1.2C The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed.

ATE_FUN.1.3C The test procedure descriptions shall identify the tests to be performed and describe the scenarios for testing each security function. These scenarios shall include any ordering dependencies on the results of other tests.

ATE_FUN.1.4C The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE_FUN.1.5C The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified.

Evaluator action elements:

ATE_FUN.1.1E The evaluator shall confirm that the information provided meets all requirements for

content and presentation of evidence.

ATE_IND.2 Independent testing - sample

Developer action elements:

ATE_IND.2.1D The developer shall provide the TOE for testing.

Content and presentation of evidence elements:

ATE_IND.2.1C The TOE shall be suitable for testing.

ATE_IND.2.2C The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

Evaluator action elements:

ATE_IND.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.2.2E The evaluator shall test a subset of the TSF as appropriate to confirm that the TOE operates as specified.

ATE_IND.2.3E The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

5.2.7. Vulnerability Assessment

AVA_MSU.2 Validation of analysis

Developer action elements:

AVA_MSU.2.1D The developer shall provide guidance documentation.

AVA_MSU.2.2D The developer shall document an analysis of the guidance documentation.

Content and presentation of evidence elements:

AVA_MSU.2.1C The guidance documentation 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.

AVA_MSU.2.2C The guidance documentation shall be complete, clear, consistent and reasonable.

AVA_MSU.2.3C	The guidance documentation shall list all assumptions about the intended environment.
AVA_MSU.2.4C	The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).
AVA_MSU.2.5C	The analysis documentation shall demonstrate that the guidance documentation is complete.

Evaluator action elements:

AVA_MSU.2.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_MSU.2.2E	The evaluator shall repeat all configuration and installation procedures and other procedures selectively, to confirm that the TOE can be configured and used securely using only the supplied guidance documentation.
AVA_MSU.2.3E	The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected.
AVA_MSU.2.4E	The evaluator shall confirm that the analysis documentation shows that guidance is provided for secure operation in all modes of operation of the TOE.

AVA_SOF.1 Strength of the TOE security function evaluation

Developer action elements:

AVA_SOF.1.1D	The developer shall perform a strength of the TOE security function analysis for each mechanism identified in the ST as having a strength of the TOE security function claim.

Content and presentation of evidence elements:

AVA_SOF.1.1C	For each mechanism with a strength of the TOE security function claim the strength of the TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the PP/ST.
AVA_SOF.1.2C	For each mechanism with a specific strength of the TOE security function claim the strength of the TOE security function analysis shall show that it meets or exceeds the specific strength of function metric defined in the PP/ST.

Evaluator action elements:

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

AVA_SOF.1.2E The evaluator shall confirm that the strength claims are correct.

AVA_VLA.2 Independent vulnerability analysis

Developer action elements:

AVA_VLA.2.1D The developer shall perform a vulnerability analysis.

AVA_VLA.2.2D The developer shall provide vulnerability analysis documentation.

Content and presentation of evidence elements:

AVA_VLA.2.1C The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for ways in which a user can violate the TSP.

AVA_VLA.2.2C The vulnerability analysis documentation shall describe the disposition of identified vulnerabilities.

AVA_VLA.2.3C The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

AVA_VLA.2.4C The vulnerability analysis documentation shall justify that the TOE, with the identified vulnerabilities, is resistant to obvious penetration attacks.

Evaluator action elements:

AVA_VLA.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VLA.2.2E The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure the identified vulnerabilities have been addressed.

AVA_VLA.2.3E The evaluator shall perform an independent vulnerability analysis.

AVA_VLA.2.4E The evaluator shall perform independent penetration testing, based on the independent vulnerability analysis, to determine the exploitability of additional identified vulnerabilities in the intended environment.

AVA_VLA.2.5E The evaluator shall determine that the TOE is resistant to penetration attacks performed by an attacker possessing a low attack potential.

5.3. IT Environmental Security Requirements

The IT environmental security requirements in the ST comprise the functional components in Part 2 of the common evaluation criteria.

The following table shows the summary functional components list in the IT environment.

Security function class	Security function component	
FPT (TSF protection)	FPT_STM.1	Reliable time stamps
FTP (Secure route/channel)	FTP_ITC.1	Inter-TSF trusted channel

FPT_STM.1 Reliable time stamps

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_STM.1.1 The **IT environment** shall be able to provide reliable time stamps for the TSF use.

Application notes: The TOE security requirements are used to provide the time stamp function that guarantees sequential generation of audit data in connection with the security audit function. Accordingly the abovementioned security requirements are not implemented for the TOE security requirements and TOE can use the time provided in the TOE environment.

FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1 The **IT environment** shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 The **IT environment** shall permit the *TSF* to initiate communication via the trusted channel.

FTP_ITC.1.3 The **IT environment** shall initiate communication via the trusted channel for [signature update, harmful site list update].

6. TOE Summary Specification

This section describes the security functions of the TOE and the assurance measures taken to ensure correct implementation, and maps them to the security requirements.

6.1. TOE Security Requirements

This section presents the IT security requirements to meet the functional requirements and the evidences about how the security requirements satisfy TOE security requirements.

The TOE security functions are as follows:

6.1.1. Security Auditing, Audit Data Generation, and Protection (Audit)

It generates audit data on all packets passing through the TOE and all processes executed in TOE for administrators to query, analyze, and integrate the generated audit data for management. It also manages the file system that stores audit data in order to protect audit data.

6.1.1.1. Audit Data Generation and Protection (Audit_Gen_Protect)

The audit data generation and protection function generates audit data and stores audit data on the file system when all activities and processes performed through the TOE start and end. It generates audit data for auditable events defined by administrators. It generates audit data such as the activity log that records TOE activities, counter log that records the number of bytes and counters generated by each application for a certain period of time, IPS log that records events using the IPS function, and warning log for informing to administrators.

To support the security requirements defined in FAU_GEN.1, the TOE audit data generation and protection function generates the following audit records:

- Start-up and shut-down of the audit functions.
- Actions taken due to imminent security violations
- Enabling and disabling of any of the analysis mechnisms; and Automated responses perfomed by the tool
- All modifications to the audit configuration that occur while the audit collection functions are operating.
- Decisions to permit requested information flows.

- the reaching of the threshold for the unsuccessful authentication attempts and the actions (e.g. disabling of a terminal) taken and the subsequent, if appropriate, restoration to the normal state(e.g. re-enabling of a terminal)
- Unsuccessful use of the user identification mechanism, including the user identity provided
- Failure in using the user identification mechanism and the provided user identity provided
- Use of management functions
- Modifications to the group of users that are part of a role
- Changes to the time
- Any failure detected by the TSF.
- Rejection of allocation operation due to resource limits.
- Locking of an interactive session by the session locking mechanism.
- Termination of an interactive session by the session locking mechanism.
- Failure of the trusted channel fuctions.

For the abovementioned auditable events, each audit record writes the following:

- Date and time of the event
- Event type
- Subject identity
- Result of the event (success or failure)

For the abovementioned auditable events, it records the additional audit records as follows:

- Actions taken due to imminent security violations: Identity of the recipient of actions
- Decisions on requests for information flow: Object identification information

The date and time of an event represents when the event has occurred: year, month, date, time. The identity of the subject for each audit data contains the policy ID (Rule ID), FW name, Src IP, Src port, protocol, Dest IP, and Dest port. Using this identity of the subject, the TSF can associate the identity of the user who has generated an event with auditable events.

In case of excessive logs, each log keeps “level/time/FW name/message” in order and the details are as follows: “Minor / 2006-09-22 09:34:18 / TOE-1 / excessive logs: 128467 (logs/sec).”

The authorized administrator of the TOE can optionally include or exclude audit data based on the event type (activity log, counter log, IPS log, and warning log on the log configuration menu) and items for each IPS rule.

The TOE can detect a potential security violation through identification and security policy violations and warning logs (decline of excessive packets, excessive mails, excessive logs, too low traffics, excessive traffics

from external networks, interface shut-down, excessive sessions, excessive CPU usage, excessive memory usage, file system alert, virtual IP address alert, main system problem alert, alert for abnormal process termination and duplicate execution, duplicate IP addresses, more than 100 dynamic rules on the same hash bucket, and data integrity violation).

When a potential security violation is detected, the TOE notifies to the authorized administrator this event on the alarm window using the administrator program and sends emails registered by the authorized administrator.

In preparation for the case the stored data exceeds the capacity, when the specified size of the file system is used, the TOE generates audit data for sending a warning message to the administrator using the administrator program and sends emails. When the file system storing audit data is exhausted, the TOE prevents auditable events except for actions taken by the authorized administrator with a special privilege from happening.

The TOE allows the access to audit records only by the administrator program to prevent any unauthorized deletion and modification.

The TOE generates audit data with the log module and log daemon. The daemon generating audit data periodically fetches the audit data that has been generated in the kernel and stored in the buffer. Because the increased traffic can increase the size of the audit data, the audit daemon prepares a large buffer to provide it in the kernel.

The TOE has no general user accounts but the administrator account so only the authorized administrator can query all the audit data. The CLI using the console does not provide the delete command so even the authorized administrator cannot modify any audit records.

The TOE time is set by the RTC (real-time clock) that the hardware provides during the system booting. The RTC has its own battery to provide reliable time with no power. The system time is updated by the CPU clock interrupt after system booting to provide reliable time stamps to be used for recording security audit data.

The TOE also controls the time through the NTP (network time protocol) server. The TOE adjusts the time by corresponding its own clock with the clock of the server. The TOE provides reliable time after synchronizing the time with the external NTP server to offer reliable time stamps by utilizing it during recording security audit data.

When the file system storing audit data is exhausted (over 99% of the audit storage capacity), the TOE sends an email to the administrator or notifies this event for the administrator to take possible measures.

When the file system is fully filled, the TOE blocks all network packets passing with the session connected by

the authorized administrator and does not leave any audit records except for administrator activities. The TOE detects illegal accesses or attacks based on the generated warning audit data.

Type of audit data		Description
Log Module (Kenel_ LogD)	**Session Log**	The session logs indicate the audit data on the allowed packets and declined packets and are generated by the log module in the kernel for delivery to the log daemon. All audit data allowed or declined through the packet filtering is recorded with a fixed format into the log buffer through a regular processing procedure of the log module (LOG_Kernel) and the recorded audit data is periodically read by the log daemon (LOGD).
	ASCII Log	The ASCII log indicates the kernel-initiated kernel IPS (hacking log), kernel warning log, and such. It is generated in the log module (LOG_Kernel) inside the kernel and is delivered to the log daemon. The ASCII logs are flowed into the log module (LOG_Kernel) inside the kernel and the logs are structured into the UDP packets by the log module (LOG_Kernel) to be delivered to the log daemon (LOGD).
Log Daemon (LOGD)		The log daemon is used to manipulate all logs generated by each TOE function for getting practical log information. The manipulated logs are stored into each predefined log file. The log daemon (LOGD)-processed audit data is subdivided into the kernel logs, application logs, and the logs that are generated after fetching information from the system. The log daemon (LOGD) generates declined packet logs, allowed session logs, user activity logs, kernel IPS logs (or hacking logs), and kernel warning logs after reading the data in the kernel buffer. In case of the application logs, the logs are directly delivered from the application programs to generate the application logs for the SMTP proxy, URL blocking, traffic balancing, IPS, administrator and user authentication, and warnings.

	The logs that the log daemon (LOGD) generates by itself after getting the information from the system are divided into hardware logs including the CPU usage-related logs, memory capacity logs, and network interface card (NIC)-related logs, and software logs including the policy-related logs and process information-related logs.

Table 17 Types of audit data

The audit data that can be monitored by the administrator program is divided into the **activity logs**, **counter logs**, **IPS logs**, and **warning logs** and the meaning of each audit data is shown for monitoring visibility in Table 18 Types of audit data that can be monitored.

Type	Meaning
Activity Log	This audit data generates records for each activity in regard to the TOE activities recording for each packet (in case of packet filter) and transaction (in case of proxy), system shut-down/start-up events, and connection/configuration/modification events by the administrator. The activity logs record the following: Date and time of the event Information of the place sent and the place received Received data size in bytes Administrator information
Counter Log	The audit data records the processed bytes for each interface, the number of drop packets, and the like at intervals of a certain period. The counter log records the following: Date and time of the evnet The number of cases for each counter log The usage rate for each counter log
IPS Log	If any hacking attempts are detected when the TOE intrusion protection function is performed, it records the audit data like drop, reset, alarm, and logging events for the corresponding packet.

	The IPS log records the following: Activity/alarm-related Importance Date and time of the event Information of the place sent and the place received Policy code Type of attack Counter-related Date and time of the event Number of total detections Number of detections for each IPS action (log, alarm, reset, drop) Client separation-related Importance Date and time of the event Client address ACTION RESULT Information details Hacking alarm-related Date and time of the event Type of hacking Information of the place sent and the place received Whether to detect and block or not Blocking period
Warning Log	The warning logs are notified to the authorized administrator on the alarm window through the administrator program or using the emails registered by the authorized administrator. Because they are very important audit data that can be directly linked to the system failures, their levels can be set depending on the data importance and the corresponding audit data can be monitored for each level. The importance levels are subdivided into Critical, Major, and Minor, and the administrator can set the corresponding level. When the warning window appears on the administrator program according to the settings, the warning levels are also displayed. The warning log records the following:

	Date and time of the event Importance level for the event Message for event detection

Table 18 Types of audit data that can be monitored

6.1.1.2. Audit Data Review (Audit_Review)

The administrator can review the audit data and intrusion detection results by converting the TOE audit data using the administrator program. The administrator can also review the audit data after searching and sorting them according to the subject identity, target object, the date and time of the event, the time, type of the event, importance of the event, and consequences of the event, and can review the result intrusion results after searching and sorting them according to the sequence number, policy code, protocol, risk, source port, destination port, and explanation.

The review of audit data can be executed after setting the review cycle: daily, weekly. The desired data can be selectively reviewed by including or excluding the type of the event (activity log, counter log, IPS log, and warning log on the log configuration menu) and items for each IPS rule.

The review of audit data is performed after checking the subject for audit data review (target and review cycle) by retrieving the audit data review target and configuration file. After analyzing only the corresponding log file for daily or weekly review, the administrator can see the results on the screen or in a stored document after converting to text formats.

6.1.2. Security Management (SEC_MAN)

6.1.2.1. Management of Security Functions (Man_Sec_Fun)

The administrator can determine, disable, enable, and modify the behavior on the following functions only through the console port and the administrator program that are directly connected to the TOE. In order to directly interoperate between the TOE and the administrator using the administrator program, first of all, the administrator identification and authentication procedure must be passed. Accordingly the right to perform the abovementioned functions is restricted to the authorized administrators.

List of functions	Capability
Audit data statistics processing	Determine and enable the behavior

Maintenance of actions to be taken in case of imminent audit storage failure	Determine, disable, and modify the behavior
Maintaining the actions to be taken on audit storage failure	Determine, disable, enable, and modify the behavior
Backup and recovery of audit trails	Determine, disable, enable, and modify the behavior
Selection of the choices in audit trails	Determine, disable, enable, and modify the behavior
Maintaining the threshold on audit trails	Determine and modify the behavior
Management of the authentication data by an administrator	Disable and enable the behavior
Specification of maximum limits for a resource for subjects by an administrator	Determine, disable, enable, and modify the behavior
Management of security attributes for the SECUINXG IPS policy used to make explicit access decisions	Determine, disable, enable, and modify the behavior
Management of the actions against security alarms	Disable, enable, and modify the behavior
Maintenance of the group of users with read access right for the security audit review	Determine and modify the behavior
Management of security attributes	Determine the behavior
The management of the metric used to verify the secrets	Determine and modify the behavior
The user security attribute administration	Determine the behavior
Managing the list of actions that can be taken before the user is authenticated	Modify the behavior
Specification of the time of user inactivity after which lock-out occurs for an individual user	Determine and modify the behavior
Specification of the default time of user inactivity after which termination of the interactive session occurs	Determine and modify the behavior
Management of the time	Determine and modify the behavior
System shut-down and rebooting	Determine and enable the behavior
Identification and authentication data management	Determine and modify the behavior

Management of the threshold for unsuccessful authentication attempts	Modify the behavior
Configuring the actions that require trusted channel	Modify the behavior
Managing the group of users that are part of a role	Determine, enable, modify the behavior
Management of actions to be taken in the event of an authentication failure	Determine and modify the behavior
Specification of the default time of user inactivity after which lock-out occurs	Modify the behavior
Maintaining rules for analyzing potential violations	Determine, disable, enable, and modify the behavior
Manual test for abstract machines	Enable the behavior
Management of anti-spam settings	Determine, disable, enable, and modify the behavior
IPS signature update management	Enable the behavior
Managing the group of roles that can interact with the limits on the TSF data	Determine and modify the behavior
Management of limits on TSF data	Modify the behavior
Managing the group of administrator that can interact with the limits on the TSF data	Determine and modify the behavior

Table 19 Security list of functions

6.1.2.2. Management of Security Attributes (Man_Sec_attr)

Only through the administrator program, the administrator can query, modify, delete, and generate the following security attributes: In order to operate the administrator program, first of all, the administrator identification and authentication procedure must be passed. Accordingly the right to perform the abovementioned functions is restricted to the authorized administrators.

List of security attributes	Operation list
Security label (1 to 20)	Modify
Group that the subject belongs to	Query, modify, delete
Destination URI	Modify, delete, create
Time	Modify
IP address	Modify, create
Port number	Modify, create
Data information	Modify, delete, create

Table 20 List of security attributes

6.1.2.3. Management of TSF Data (Man_TSF_Data)

6.1.2.3.1 Restricting TSF Data Management to the Authorized Administrator

Only through the administrator program, the administrator can manage the following TSF data: In order to operate the administrator program, first of all, the administrator identification and authentication procedure must be passed. (Refer to Administrator identification and authentication in the TOE user identification and authentication items) Accordingly the right to perform the abovementioned functions is restricted to the authorized administrators.

- **Management list**
  - Audit data statistics processing
  - Backup and recovery of important files of the TOE on the semi-permanent secondary memory unit
  - Modification, deletion, and creation of identification and authentication data
  - Modification, deletion, and creation of anti spam settings
  - Changes to the time
  - IPS signature update

6.1.2.3.2 Management of limits on TSF data

The range of the audit trails limit (default: 99%) is from minimum 1% to maximum 100%. When the input value is out of the range, the administrator program does not allow it. The prefixed range cannot be modified even by the authorized administrator.

The allowable number of failed authentication trials (default: 3) is 3 to 10. When the input value is out of the range, the administrator program does not allow it. The prefixed range cannot be modified even by the authorized administrator.

The self-testing interval is determined by the day and time that has been defined by the administrator.

The TSF shall take the following actions, if the TSF data are at, or exceed the indicated limits:

- When the audit trails reach the indicated limits (exceeding 99% of the memory capacity)
  - Sends an email for the administrator to take possible measures
  - Notifies the event on the alarm window when the administrator has logged on the administrator

program

- When the failed authentication trials reach or exceed the limit
  - Protects the authentication of the corresponding user until the authorized administrator takes possible actions
  - Records audit events
  - Notifies the authorized administrator on the alarm window of the administrator program
  - Notifies the authorized administrator through the registered email

- When a TSF data integrity failure is detected
  - Notifies the event on the alarm window when the administrator has logged on the administrator program
  - Notifies the authorized administrator through the registered email

- When UDP flooding attacks reach or exceed the limit
  - Detects and blocks the attacks
  - Notifies the event on the alarm window when the administrator has logged on the administrator program
  - Records audit events

- When SYN flooding attacks reach or exceed the limit
  - Detects and blocks the attacks
  - Notifies the event on the alarm window when the administrator has logged on the administrator program
  - Records audit events

- When scanning attacks reach or exceed the limit
  - Detects and blocks the attacks
  - Notifies the event on the alarm window when the administrator has logged on the administrator program
  - Records audit events

- When ping flooding attacks reach or exceed the limit
  - Detects and blocks the attacks
  - Notifies the event on the alarm window when the administrator has logged on the administrator program
  - Records audit events

6.1.2.3.3 Secure TSF Data List

- Time value: The time value allows only positive integers. Its unit is fixed (e.g. second) and it can be changed by the authorized administrator.
- IP address: Only the decimal-dotted IP format is allowed.

6.1.2.3.4 IPS Signature Update

IPS signature can be updated periodically or urgently. The periodic update is set by the administrator so its cycle can have a value for daily or weekly. The urgent update is downloaded to the IP address that the administrator has set when any urgent update event happens promptly or by SECUI servers.

■ **Review of IPS signature update cycle**

The administrator can review the settings for signature update cycle and signature urgent update by selecting “IPS > Malicious signature test > Signature update configuration” in the administrator program.

■ **Setting, modification, and deletion of IPS signature update cycle**

The administrator can set, modify and delete the settings for signature update cycle and signature urgent update by selecting “IPS > Malicious signature test > Signature update configuration” in the administrator program.

- Automatic signature update configuration
  - SUN
  - MON
  - TUE
  - WED
  - THU
  - FRI
  - SAT
  - Time: 0 to 23 O’clock, 0 to 59 minute
- Urgent signature update configuration
  - Recipient IP
  - Update server address

6.1.2.4. Management of Security Roles (Man_Sec_Role)

The TSF plays the authorized administrator role using the following methods:

When the administrator is registered (refer to Administrator registration in TOE user administration), the

administrator ID, password, and access level are stored as an entry for the administrator information list file that exists in a semi-permanent storage unit. Accordingly the administrator ID can associate it with the corresponding password and access level. In this manner, the administrator can play the administrator role.

The administrator's ID, password, and access level that are registered through the administrator program, are delivered to the TOE through SSL communication with integrity and confidentiality guaranteed. (Refer to Administrator program and Secure route for the TOE communication.)

This administrator information list file is periodically tested for integrity checking. (Refer to Stability test for security functions.) Accordingly it guarantees that the information for the authorized administrator and the corresponding role is not damaged.

When the administrator's ID and password are input to identify and authenticate the administrator through the administrator program, the administrator program delivers its ID and password to the TOE as a network message. Those items are delivered to the TOE through SSL communication with the integrity and confidentiality guaranteed. (Refer to Secure route for administrator program and TOE communication.)

During the administrator authentication procedure, the TSF searches the entry that is the same as the ID received from the administrator information list file and then checks whether the corresponding password is the same as the delivered password. When this checking procedure is passed, the administrator's ID, password, and the corresponding access level information are delivered to the administrator program. Those items are delivered to the administrator program through SSL communication with the integrity and confidentiality guaranteed. (Refer to Secure route for administrator program and TOE communication.)

The administrator program receives them and opens the security management GUI (Graphic User Interface). At this time, the performance capability limit is determined depending on the access level. The access levels are divided into level 1, 2, and 3. This access level is determined during the administrator registration. The level 1 is allowed to review the TSF data only, the level 2 to perform all activities except for TOE start-up and shut-down, and the level 3 to perform all activities.

Modifying the access level of the administrator is possible by the administrator with a higher access level. For instance, the access level of the level 1 administrator can be modified by a level 2 or level 3 administrator, and likewise, the access level of the level 2 administrator can be modified by a level 3 administrator.

When the administrator attempts to identify and authenticate the administrator using CLI, the administrator's ID and password are delivered to the TOE through the console port. The TSF searches the entry that is the same as the ID received from the administrator information list file and then checks whether the corresponding password is the same as the delivered password. If an entry is found and its access level is level 3, login is allowed.

It is guaranteed that the TSF plays the authorized administrator role using the abovementioned procedure.

6.1.3. Protection of Internal Property and Information from Illegal Accesses (Firewall)

6.1.3.1. Unauthorized Access Blocking by the Organizational Policy (Firewall_acc_ctrl)

The TOE identifies and authenticates internal/external IT entities sending/transmitting information through the TOE (access subject) and subsequently applies the information flow control rules based on the IP addresses and security attributes (e.g., security levels: 1 to 20) to determine whether each IT entity has access right. The subject allows operations to allow, deny, refuse (ICMP) and refuse (reset) security attributes of the network packets passing through the TOE. The security attributes are the destination URI (uniform resource identifier), time, and packet's header and data information.

When there is no access right to users and access objects, the access control is executed by terminating the corresponding user connection. The TOE information flow control is divided into the discretionary access control and compulsory access control as well as the access control based on the additional rules.

6.1.3.1.1 Discretionary Access Control Function (Firewall_acc_Ctrl_Imp)

The discretionary access control rules are divided into the following functions, which control all subjects and objects passing through this intrusion protection system.

- **Packet filtering discretionary access control by the state checking technique (Firewall_acc_Ctrl_Imp-State)**

The packet filtering discretionary access control using the state checking technique is applied to check whether the authorized user subject has the access right to the object. The discretionary access control is divided into the static access control rules (static rule) and dynamic access control rules (dynamic rule). The packet filtering discretionary access control rules are created using the values for source IP, source port, destination IP, destination port, protocol, direction information, rule type, and mode.

The TOE allows implicitly allowed services and declines the other services. Because the static access control rules are not set during the TOE delivery, the authorized administrator can use them after configuring the static access control rules that the TOE implicitly allows.

The authorized administrator can set the static access control rules with the following items.

Item	Meaning of configuration item
SEQ	Filtering rule-applied priority
Rule ID	Unique number assigned to each filtering rule
From	Traffic sender address
To	Traffic recipient address
Service	Type of application
Action	Operation (allow, deny, refuse (ICMP), refuse (reset))
Time	Configuration for each time
T. S	When applying the traffic adjustment, Queue number that corresponds to the bandwidth for adjustment
Log	Whether to audit the configured filtering rule or not
Data blocking	FTP commands, keyword control, and blocking by the payload length
RPC	Allowed service list among RPC services
Category	Malicious (such as porno) site blocking list
IPS	Item to set whether to perform packet test by the signature or not

■ **Discretionary access control by the URL blocker (Firewall_acc_Ctrl_Imp-URLB)]**

The discretionary access control function by the URL blocker forcefully blocks the internal user's access request to a specific site.

The discretionary access control by the URL blocker is a URL blocking list among the IPS selection items list by the authorized administrator.

When a specific Web server's URL is requested by an internal user, the URL blocker blocks the request for the URL that has been blocked by the TOE authorized administrator.

The administrator decides whether to activate the function or not using a check box in the administrator program.

The URL blocker function extracts the URL transferred for Web connection and compares it with the table

that contains the blocking object information. When the corresponding data is a blocking target, the URL blocker terminates the corresponding page and delivers the blocking page to the requested user. When the URL is blocked by domain, the URL blocker stores the corresponding IP and generates audit data. If the corresponding IP already exists in the blocking table, the URL blocker does not add the IP. It also generates audit data for the blocked page.

When the corresponding data is a blocking target, it terminates the connection to the page and delivers the blocked page to the requested user. The table with the blocking object information is divided into the blocking list that the administrator has registered and the KISCom (Korea Internet Safety Commission) category. The list of exceptions for blocking URL is used to register exceptions for blocking URL and is generated by an administrator through the administrator program.

- **IPS-WEB-initiated discretionary access control (Firewall_acc_Ctrl_Imp-IPSWEB)**

The IPS-WEB-initiated discretionary access control function is used to forcefully control URLs using the Web server vulnerabilities like CGI and Unicode bugs.

The IPS-WEB-initiated discretionary access control is a Web vulnerability blocking list among the IPS selection items list by the authorized administrator.

The IPS-WEB extracts URLs from the HTTP request data that is delivered to the Web server to compare with the blocking list. When it is identical to the pattern in the blocking list, the IPS-WEB generates a log and terminates the corresponding connection.

The administrator decides whether to activate the function or not using a check box in the administrator program and select any application target of inbound and outbound using a check box. When inbound is selected, the function is applied to the HTTP request data that is flowed into the Web server and in case of outbound, the function is applied to the external Web server.

The blocking list consists of the system-defined list and the administrator-defined list. In case of an entry in the system-defined blocking list, the administrator can determine whether to exclude from blocking, whether to generate logs, and whether to block the entry. In case of the administrator-defined blocking list, the blocking objects can be modified/added/deleted for that blocking list. When adding or changing the list, the administrator can arbitrarily select the blocking objects. In addition to the objects, the administrator can determine whether to exclude from blocking, whether to generate logs, and whether to block the entry.

6.1.3.1.2 Compulsory Access Control Function (Firewall_acc_Ctrl_Exp)

Different from the discretionary access control, the compulsory access control function forcefully controls access to objects and services according to the security levels of the subject and object, and it performs the following:

The security label-initiated compulsory access control function forcefully controls access to objects after checking whether the subject that has acquired discretionary access control rules, has the security level that enables the subject to access the corresponding object. The function allows the connection when the security level of the subject is equal to or higher than the one of the object. The authorized administrator can set the security level (1 to 20).

The security label can be set after activating each function for the host object, network object, and service, and its default value is 20.

The connection is allowed when the security level of the subject is equal to or higher than the one of the object. Otherwise, the connection is restricted. The mechanisms for processing the results of the security level comparison are as follows:

- Allow

  When the security level of the subject is equal to or higher than the one of the object

  Allow connection

- Deny

  When the security level of the subject is lower than the one of the object

  Connection denied, packet destruction

6.1.3.1.3 Additional Access Control (Firewall_acc_Ctrl_Additional)

The additional access control function implicitly allows or declines a subject's connection to an object based on the following additional rules:

The additional access control rules are divided into the implicit allowance and the implicit decline. Based on the following rules, the implicit allowance allows access after generating dynamic access control rules and the implicit decline denies access when the packet is in the decline list.

- Implicit allowance

  Allow the implicit access through the registered administrator IP

  Allow the implicit access for the TOE-generated packets

- Implicit decline

  Deny the implicit access using the registered decline list (the decline list sent from the IPS module)

6.1.3.2. Blocking Information Inflow and Outflow by Organizational Policies (Firewall_Proxy)

Data control function (Firewall_Proxy)
Proxy pertains to the middle gate between the server and the client and enables the TOE kernel to redirect packets in the middle to the proxy and link sessions between the proxy and the client and between the proxy and the server. For all operations of external IT entities that send or receive data through the TOE, of the data transmitted through the TOE, and the data flow to/from the controlled subject, the data control function applies the following NXG IPS policies: The authorized administrator can take security measures for the protocol using the TOE proxy security function. Possible security measures that can be taken are the following: ● Selective processing of the configuration in the general tab (using normal/virtual address, blocking spam mails, filtering, restricting the mail size) of the anti-spam on the management tool ● Selective processing of the configuration in the Anti-Virus tab (SMTP configuration, POP3 configuration) of the anti-spam on the management tool ● Selective processing of the configuration in the mail address filtering tab (recipient email address, sender email address) of the anti-spam on the management tool ● Selective processing of the configuration in the keyword filtering tab (keyword, direction, description) of the anti-spam on the management tool ● Selective processing of the configuration in the using the virtual address tab (Address to be resent, Address of address conversion) of the anti-spam on the management tool ● Selective processing of the configuration in the internal mail server of the anti-spam on the management tool ● Selective processing of the configuration in the MAPS tab of the anti-spam on the management tool

<table>
<tr><td colspan="2">

- Selective processing of the configuration in the exception list for size restriction of the anti-spam on the management tool

The TOE can selectively perform anti-virus scanning and scanning & deletion for the proxy function in the ant-virus quarantine configuration tab and the anti-spam tab.

After any security measures, the TOE delivers the corresponding packets to the server after converting packets with making itself the sender. The server recognizes that it links sessions between itself and the TOE for communication. When the server delivers packets to the TOE as a response, the TOE checks the content of those packets and delivers them to the client with a conversion (makes the server sender).

The data control function guarantees that all operations that cause information flow to/from all subjects in the TSC, follow the NXG IPS policies.
</td></tr>
<tr><td>SMTP proxy</td><td>The SMTP proxy is used when host accesses the mail server, and it can apply specific security rules to SMTP services. A part of the function is applied to the POP3 photo mail.</td></tr>
<tr><td>POP3 proxy</td><td>The POP3 proxy is used when host accesses the mail server, and it can apply specific security rules to POP3 services. The POP3 proxy function can be activated or inactivated using a check box in the administrator program.</td></tr>
<tr><td>Virus wall</td><td>For the files transmitted through SMTP, POP3, and HTTP proxy, anti-virus scanning is executed. When a file is delivered through the proxy of the protocol defined in the administrator program, the proxy requests anti-virus scanning and delivers the file to the virus wall. The virus wall that has received the file makes a copy of it to execute the anti-virus scanning and its results are delivered to the corresponding proxy.</td></tr>
</table>

Table 21 Data control function

6.1.4. Protection of Illegal Accesses and Attacks (IPS)

6.1.4.1. Detection of Illegal Accesses and Attacks (IPS_Detect)

There are hacking attacks that exhaust host and network resources and cause availability problems through the known vulnerabilities. In order to prevent those problems, it is necessary to have the intrusion protection system (IPS) that detects illegal accesses and attacks and protect the system from them.

The illegal access and attack detection (IPS_Detect) function observes the attributes of the data passing through the TOE to take counteractions that the administrator has defined for any detected attack.

The illegal access and attack detection (IPS_Detect) function collects the information on the events such as host accesses, service requests, network traffics, data inflow, and the inflow of the information with a specific content and pattern. The information of the collected intrusion detection events includes the subject identity, security attributes, event type and the related information, and the date and time of the event.

The illegal access and attack detection (IPS_Detect) function performs the analysis on the inflow of packets over the limit that can be classified into the analysis of comparison between the collected data and the security-violated event list, abnormal protocol packet analysis, network service decline attempt analysis, and security-violated events on the collected data basis.

The function assigns the maximum value for connection of packets in SYN that can be used by IT entities at the same time. As mentioned above, the function can effectively block service decline attacks.

The illegal access and attack detection (IPS_Detect) function can update signatures by manually pressing the update button or by using the periodic update at the interval that the administrator has defined.

The function is subdivided into the TOE kernel intrusion detection and detection of the intrusion into the TOE application layer, and detects and processes the following: The following are the IPS selection item list details (corresponds to abnormal traffic/abnormal protocol/general hacking protection list):

- **General hacking protection list**
  - Blocking of the packets flowed from the sender with an abnormal address
    - ✓ Unauthorized RFC 1918 IP address
    - ✓ IP address that cannot be a normal sender address
  - Blocking of abnormal IP packets
  - Defense against land attacks
  - Blocking the connection to TCP such as source port and destination port
  - Defense against ping of death attacks
  - Blocking of ICMP destination unreachable packets
  - Blocking the packets with the source route option

- **Abnormal traffic**
  - SYN flooding

- UDP flooding
- Ping flooding
- Scanning attacks
- Watching traffics for each port

■ **Abnormal traffic**

- Blocking of not requested ICMP responses
- Restriction of abnormal HTTP requests

■ **Test on malicious signatures**

- Packet detection and blocking by the signature
- Client separation

6.1.4.2. Reactions on Illegal Accesses and Attacks (IPS_React)

When any hacking attempts are detected, the TOE prevents illegal attacks using the following methods:

Reaction	Method
Alarm	If any hacking attempts are detected, the TOE notifies the administrator of this event based on the hacking log.
Report	If any hacking attempts are detected, the TOE creates the related logs into a report and sends it to the predefined email address.
Blocking and Terminating sessions	The TOE notifies KLSM of a pair of IP addresses to be blocked on the hacking packet detected based on the administrator configuration to block accesses for a certain period of time or forcefully terminates the session in case of TCP.

Table 22 Responses against illegal accesses and attacks

6.1.5. TOE User Identification and Authentication (UI&AD)

There are authorized administrators who manage the system itself. The authorized administrator shall be registered in the TOE and pass the user identification and authentication procedure through the TOE.

6.1.5.1. TOE User Registration (User_Register)

The TOE user registration is made by the authorized administrator through the administrator program.

- **Administrator registration (Admin_User_Register)**

For the administrator to use the TOE administrator program through the console port or to directly interoperate with the TOE through CLI, the administrator identification and authentication procedure is required after being registered as an administrator.

To register a main administrator, when the system starts, the main administrator account (cannot be deleted before uninstallation) and password should be set during the log-in step by directly interacting with the TOE through the TOE console port, and the IP address of the IT entity that is to run the administrator program.

The authorized administrator can add/modify/delete any administrator on the administrator list through the administrator program. When registering the administrator account and password, the access level, department, phone number, mobile phone number, and email address are input. And also the IP address of the external IT product to be managed by the administrator can be input. The administrator cannot register the administrator with a higher security level. And also the security level of an existing administrator cannot be modified to have a security level higher than the one of the registration subject. (For the items on the access level, refer to 6.2.2.4 Management of security roles.)

6.1.5.2. TOE User Identification and Authentication (User_InA)

The TOE user identification and authentication is used to identify and authenticate administrators. The administrator identification and authentication is enabled by interoperating with the TOE through the administrator program or the console port. The TOE users can perform TOE communication after the identification and authentication procedure and all the authentication connections are enabled by SSL communication.

The TOE user identification and authentication satisfies the strength of function—medium for the strength of the minimum security function of the statistics and permutation mechanism. The FIA_UAU.1 satisfies the strength—medium for the strength of the minimum security function of the statistics and permutation mechanism. The TOE user identification and authentication provides the administrator authentication function using ID and password, and the password convention rules restrict the length to 6 to 32 characters with mixed alpha-numeric and special characters. In addition, it provides the administrator authentication function using the

one-time password (S-Key) mechanism. For demonstration of the number of cases in the password and one-time password mechanisms, refer to the *6.2 section in Analysis on vulnerabilities and misuses*.

In order for the administrator to use the administrator program, the IP address of the predefined IT entity should be used. The TOE stores the IP address in the file system so that it can allow connection of registered IP addresses and refuse connection of not registered IP address. The input IP address can be modified during initial installation and by the authorized administrator using the administrator program.

Before authentication, the administrator can only download and install Welf File Converter, SNMP MIB FILE, JRE 1.4.2_05, and NXG Manager but cannot perform any security-related functions. In the middle of authentication procedure, the input password is displayed with dots. The displayed authentication failure message “ID/password is not valid” does not provide the information about which one is not valid in order for the attacker not to tell it.

All connections by the administrator are made through SSL communication and when the administrator accesses the TOE, the authentication key is downloaded to the administrator PC. Whenever the administrator accesses the TOE, a new authentication key value defined by the TOE is downloaded to the administrator PC.

The administrator information including the account and password is stored in the file system and this information is used to identify the administrator using the input account and password when the authorized administrator accesses the TOE administrator program. When the identified values are identical and the downloaded authentication key value is the same as the valued stored in the TOE, the TOE allows the access.

The administrator identification and authentication procedure is performed through SSL communication. The SSL communication is used as a route for identification and authentication and generates a new SSL’s encrypted key and a integrity testing secret key SSL for authentication so that the authentication data may not be reused.

The following are the authentication modes for the identification and authentication procedure.

- Local mode: When the administrator sends the authentication information (authentication request message, administrator ID and password) to TOE, the TOE performs the administrator identification and authentication procedure by checking the authentication information and TOE administrator account file.
- S/Key mode: The TOE sends the S/Key challenge value to the administrator who has requested for value identification and authentication and the administrator sends the S/Key response value to the TOE by using the S/Key challenge value (Interaction count, seed). Arbitrarily using the seed value transmitted from the TOE blocks the data reuse.

When the administrator authentication fails 3 times (default) in a row, the TOE notifies the authorized administrator of this event and terminates the browser of the administrator PC. To perform authentication again, the browser should be open to run the administrator program. However, when the system is installed,

The number of authentication trials can be changed by the authorized administrator (default=3) and the value ranges from 3 to 10. When the number of failed authentication trials reach or exceed the predefined number, the TOE notifies the administrator of this event on the alarm window, sends email to the predefined email address, blocks the authentication for the corresponding user, and records auditable events.

If the administrator could not maintain the connection for the predefined inactive period (Idle time, default=10 minutes) due to network disconnection and miscellaneous reasons, the administrator can use normal services or terminate the session by issuing authentication again.

The administrator interoperates with the TOE through the TOE console port and passes the administrator identification and authentication procedure. First of all, the administrator must input the system default ID and password. When this default identification and authentication procedure is a success, the prompt for the administrator identification and authentication appears on the terminal.

When the administrator ID and password are input, they are compared with the administrator ID and password stored in the TOE. The information about the administrator account, password, and access privilege is stored in the file system. If the identified values are the same as the information, the TOE allows the access of the administrator. When the administrator authentication fails 3 times (default) in a row, the TOE returns to the default identification and authentication procedure.

6.1.6. TSF Stability (TSF_SECURER)

6.1.6.1. Security Stability Self test (Secu_Self_Test)

- **TSF data integrity test and reaction**

The TSF data integrity function guarantees the integrity of the corresponding file, when the administrator successfully stores/deletes/modifies the following TSF data that is stored in the TOE. In addition, this function performs the integrity test periodically or by the administrator request when the system starts or is in normal operation.

The files that check the integrity are as follows:

- TSF data
  - Access control policy data
  - TOE user identification and authentication data
  - System configuration data that stores the configuration information in the TOE
- TSF execution file

Whenever the TSF data is changed, the system calculates the checksum value using SHA-256 and stores it in the TOE and the stored value is periodically checked through the checksum value check program. The testing is performed whenever the TSF data is changed by default and can be periodically performed according to the administrator's setting (weekly (every specific day) and daily (every specific time)).

The checksum value checking program calculates the checksum values for the TSF security data and execution files stored in the system to guarantee the security data integrity after comparing with the checksum value stored in the TOE.

If the integrity checking object is modified by an attacker without using the administrator program, the checksum value for the TSF data in the system gets different from the one stored in the TOE, so the TOE notifies the administrator of this event and makes the failed TSF impossible to be changed.

■ **Self test**

To show the TSF underlying abstract machine-related security assumptions are correctly operated, the TSF performs the following series of self tests when requested by the authorized administrator and when the TOE starts.

- Self test list
  - Process operation state test
  - Disk state test
  - Data integrity test

The module that performs the security stability self test, checks the status of the processes that operate in the TOE and checks the used capacity (in KB) of all disks that are connected to the TOE. The module checks the validity of the file list for integrity test and performs the integrity test for the TSF data files and execution files.

6.1.6.2. Inter-TSF Trusted Channel (TSF_Secure_Channel)

The administrator downloads the Java program-related compression files from the TOE through the Web browser for reliable IT products. This file is executed by the Java-typed administrator program using JRE (Java runtime environment). During the communication between the administrator program and the TOE, the SSL (protocol version SSL V3) is used to maintain the transmitted data integrity and confidentiality. The TSF opens the TCP port configured, waits in a server state, and links sessions when the administrator session link request is issued. For the key exchange, RSA (1024 bit) is used, for the data encryption, 3DES (168 bit, operating mode, CBC) is used, and for the message integrity, HMAC SHA-1 is used. When the communication between the administrator program and TSF is made, the secure route is guaranteed.

For reference, briefly explaining SSL, the connection between the administrator and the TOE is made through the SSL communication. For this purpose, the administrator computer creates a new SSL context after confirming normal TCP socket connection. This SSL context contains the cipher suite information for SSL connection. When the SSL_connect is successfully made for the TOE, the socket description on the SSL connection can be got. The TOE authentication key value is acquired by using the socket description as a parameter of the SSL_get_peer_certificate. When no failure is found after checking the information of the TOE subject and issuer, the administrator communication is made using the socket description. At this time, the TOE downloads a new authentication key to the administrator computer.

6.1.6.3. TSF Protection (TSF_Protection)

Because the TOE is installed on the end point of the network, all internal traffics can communicate with the external network through the TOE (the communication from outside to inside is the same way). When network traffic arrives at an internal network port, it passes via KLSM without exception. According to the packet content, the TSP is executed by the corresponding security function (the communication from outside to inside is the same way) to satisfy the non-bypassability of the TSP through the external network port.

In addition, the TOE physical attack is protected by the assumption A. Physical security and OE. Physical security. All security functions in the TOE can be changed only by CLI and the administrator program offered from the TOE and does not provide any other external interfaces that can change the TOE file system and memory. (e.g. FTP server, Telnet server, and programming interface) Because all the processes operating in the TOE are reliable processes to implement the TSF and cannot operate other processes, the security area is separated.

The TOE time is set by the RTC (real-time clock) that the hardware provides during the system booting. The RTC has its own battery to provide reliable time with no power. The system time is updated by the CPU clock

interrupt after system booting to provide reliable time stamps to be used for recording security audit data.

The TOE also controls the time through the NTP (network time protocol) server. The TOE adjusts the time by corresponding its own clock with the clock of the server. The TOE provides reliable time after synchronizing the time with the external NTP server to offer reliable time stamps by utilizing it during recording security audit data.

When any of other NXG products is converted to the TOE, the policy conversion tool among TOE functions converts the existing policy to the TOE policy. The policy conversion tool converts the policy of the existing NXG product to the TOE policy, when it is applied after specifying the object file, policy file, and internal/DMZ network and filling in the name input items.

6.1.7. Process Monitoring (Mon_Process)

The process guard function watches whether the application process for each security function operates correctly or not. This function supports continuous services so that the security function defined by the administrator cannot be stopped due to any arbitrary error.

When the administrator defines security policies, the corresponding information is stored in the configuration file. This function can support the security function only when the corresponding process against the security policy works properly. If the corresponding process is destroyed or does not operate, the corresponding security function cannot be provided.

The function checks required processes that correspond to the configuration file the administrator has defined. After this, it checks whether the corresponding process exists in the system and operates normally or not. If the process does not operate or abnormally operates, the function restarts the corresponding process and provides normal security functions.

On the contrary, it watches whether the watchdog process operates normally or not to guarantee normal operations. If the watchdog process does not operate or abnormally operates, it restarts the watchdog process to provide normal security functions.

In addition, the TOE has two hardware power units. If a hardware power unit has a failure, power is supplied to another redundant power unit to normally provide network services through the TOE.

6.2. Assurance Measures

The TOE in the ST follows the security requirements in *Part 3 of the common evaluation criteria*.

Through assurance measures shown in the following table, TOE security requirements are satisfied.

Assurance class	Assurance component		Assurance measures
Configuration management	ACM_AUT.1	Partial CM automation	CM documentation
	ACM_CAP.4	Creation support and acceptance procedures	
	ACM_SCP.2	Problem tracking CM coverage	
Delivery and operation	ADO_DEL.2	Detection of modification	Delivery and installation guidance
	ADO_IGS.1	Installation, generation, and start-up procedures	
Development	ADV_FSP.2	Fully defined external interfaces	Functional specification
	ADV_HLD.2	Security enforcing high-level design	High-level design specification
	ADV_IMP.1	Expression of implementation for partial TSF	Verification specification Source code Life-cycle support
	ADV_LLD.1	Descriptive low-level design	Design specification
	ADV_RCR.1	Informal correspondence demonstration	Verification specification
	ADV_SPM.1	Informal TOE security policy model	Security policy model
Guidance documents	AGD_ADM.1	Administrator guidance	Administrator guidance
	AGD_USR.1	User guidance	None
Life cycle support	ALC_DVS.1	Identification of security measures	Life-cycle support
	ALC_LCD.1	Life-cycle model defined by developers	
	ALC_TAT.1	Well-defined development tools	
Tests	ATE_COV.2	Analysis of coverage	Tests
	ATE_DPT.1	Basic design test	
	ATE_FUN.1	Functional testing	
	ATE_IND.2	Independent testing: sample	

Vulnerability assessment	AVA_MSU.2	Verification of administrator guidance analysis	Analysis of vulnerabilities and misuses
	AVA_SOF.1	Evaluation for strength of the TOE security functions	
	AVA_VLA.2	Independent vulnerability analysis	

Table 23 Assurance measures

7. Rationale

This chapter describes rationale for security objectives and security requirements that satisfy the security objectives that have been defined on the security environment (threats, assumptions, operational security policies) basis. It describes that the TOE provides efficient IT security measures in the TOE security environment.

7.1. Rationale for Security Objectives

This section describes the rationale for security objectives and security requirements that satisfy the security objectives that have been defined on the security environment (threats, assumptions, operational security policies) basis.

The threat actors are computer users or IT entities that access internal computers from outside. It is assumed that the threat actors have low level of expert knowledge, resources, and motivation, and have low potential to find vulnerabilities that can be used for evil purpose.

The following are rationale of the TOE.

Security objectives / Security environment	TOE security objectives									Environmental security objectives									
	O.Availability	O.Audit	O.Management	O.TSF data protection	O.Blocking abnormal packets	O.Blocking denial of service attack	O.Identification	O.Authentication	O.Information flow control	OE.Physical security	OE.Security maintenance	OE.Authorized administrator	OE.Secure management	OE.Operating system reinforcement	OE.Unique connection point	OE.Vulnerability list renewal	OE.Operating system reinforcement	OE.SSL certificate	OE.Secure TOE external server
A.Physical security										X			X						
A.Maintaining Scurity											X								
A.Authorized administrator												X							
A.Operating system reinforcement														X					
A.Unique connection point															X				
A.Operating system time																	X		
A.SSL certificate																		X	
A.Secure TOE external server																			X
T.Spoofing		X					X	X											
T.Failure	X			X									X	X					
T.Writing errors	X	X																	
T.Inflow of illegal information			X						X										
T.Illegal access to services			X						X										
T.Abnormal packet transfer		X			X		X												
T.Attack on new Vulnerabilities			X								X		X	X		X			
T.Denial of service attack		X				X	X												
T.Continuous authentication trials		X					X	X											
T.Bypass access	X								X	X					X				
T.Address munging		X			X	X	X												
T.TSF data modification without permission	X	X		X			X												
TE.Poor management			X									X	X						
TE.Delivery and installation												X	X						
P.Audit		X					X												
P.Secure management			X									X	X						

Table 24 TOE security environment and reactions for security objectives

7.1.1. Rationale for Security Objectives

The rationale for security objectives proves that the specified security objectives are valid, efficient to handle security problems, not excessive, and indispensable.

The rationale for security objectives proves the following:

- Each assumption, threat, and organizational security objective are handled at least by one security objective.
- Each security objective handles at least an assumption, threat, and organizational security objective.

The following are rationale for TOE security objectives.

O.Availability

When a TOE failure occurs or the TOE is in the overload state by any attack, this TOE security target provides the availability of the TOE to provide the minimum network services.

Accordingly this security target provides the availability of the TOE to cope with the threats such as T.Failures, T.TSF data modification without permission, T.Bypass access-initiated threat, and T. Writing errors by the storage saturation while recording TOE audit data.

O.Audit

When users are using the security functions, with this security objective, the TOE records auditable events for each user according to the auditing policies. And the TOE guarantees the recorded auditable events are securely maintained and reviewed. It means that the TOE provides the reaction function when the audit data is saturated. The audit log generation function enables to detect the identity of the attacker using the recorded audit logs when the attacker attempts continuous authentication trials. And in case of address spoofing attacks, service decline attacks, and sending abnormal packets, the attack can be tracked using the recorded audit logs.

Accordingly this security target provides reactions against attacks such as T.Spoofing, T.Writing errors, T.Sending abnormal packets, T.Service decline attacks, T.Continuous authentication trials, T.Address munging, and T.TSF data modification without permission, and supports organizational security policy P.Audit.

O.Management

In order to carry out security policies, the TOE controls illegal access to the internal network by defining the information flow control rules. For this purpose, the TOE provides the means to securely manage the TOE and the TSF data such as configuration data generation and management, recent vulnerability signature management.

Accordingly this security target provides reactions against threats such as T.Inflow of illegal information, T.Illegal service access, T.New vulnerability attacks, and TE.Poor management, and supports organizational security policy P.Secure management.

O.TSF data protection

The security policies can be improperly executed due to the TSF data modification by unexpected attacks or TOE failures. For this purpose, the TOE guarantees that the TSF normally operates by checking any TSF data modification to support TSF data integrity.

Accordingly this security objective reacts against T.Failures and T.TSF data modification without permission.

O.Blocking abnormal packets

This security objective guarantees that the packets like TCP/IP standards-rebellious packets, packets from outside with an internal address, broadcasting packets, and looping packets, are not flowed into the network.

Accordingly this TOE security objective reacts against the threats like T.Sending abnormal packets and T.Address munging.

O.Blocking denial of service attack

An attacker can perform the network service decline attack through the TOE to internal computers. The network service decline attack exhausts the computer resources by requesting abnormally flooding services by a remote user. At this time, the internal computer allocates too many resources to the attacker so normal users cannot use the computer. In preparation of this case, the TOE guarantees that normal users use computers by blocking a certain user exclusively using computer resources.

Accordingly this security objective reacts against the threats like T.Service decline attack and T.Address munging.

O.Identification

The TOE users include administrators who manage the TOE by authorized connection and external users (IT entities) who pass the TOE without authentication to use internal computers. To process the abovementioned 2 security events, the identification function is required. The administrator identification function grants the administrator's actions responsibilities and the external IT entity identification function is necessary to generate audit data when sending abnormal packets, blocking service decline attacks, blocking address munging attacks, and external IT access trial.

Accordingly the TOE security objectives react against the threats like T.Spoofing, T.Service decline attacks, T.Address munging, T.Sending abnormal packets, T.Continuous authentication trials, and T.TSF data modification without permission, and supports P.Audit.

O.Authentication

The user who wants to access the TOE must acquire authentication. However, the authentication required to access the TOE can be vulnerable to the consecutive authentication trials by an external attacker. Accordingly the TOE must guarantee the authentication mechanism to protect the system from consecutive authentication trials. This security objective reacts against T.Spoofing and T.Continuous authentication trials.

O.Information flow control

The TOE is installed on the point that separates internal and external networks and controls the information flow according to the security policy. This security policy guarantees that the TOE identifies and blocks various attacks that can be made on the network according to the decline and allowance policies. Various attacks on the network include virus attacks, email and Web services with illegal information, and accesses to now allowed services. The TOE guarantees the secure internal network by controlling them according to the predefined rules. Accordingly, this security target reacts against T.Inflow of illegal information, T.Accesses to illegal services, and T.Bypass access.

7.1.2. Rationale for Environmental Security Objectives

It proves rationale for environmental security objectives.

The following are rationale for TOE environmental security objectives.

OE.Physical security

This environmental security policy guarantees that the TOE is installed and operated in a secure place and blocks external physical intrusion attacks and TOE modification trials to support A.Physical security assumptions and reacts against T.Bypass access.

OE.Security maintenance

This environmental security policy guarantees to maintain the same level of security as the previous by reflecting the changed environment and security policy due to internal network configuration change, increase or decrease in host range, and increase or decrease in services to support the assumption A.Security maintenance and react against the threat T.New vulnerability attacks.

OE.Authorized administrator

This environmental security objective guarantees to rely on the TOE authorized administrator, support the assumption A.Authorized administrator, and security measure P.Secure management, and react against TE.Poor management and TE.Delivery and installation.

OE.Secure management

This environmental security objective guarantees that the TOE is securely deployed and installed, and securely configured, managed, and used by the authorized administrator to react against threats T.Failures, T.New vulnerability attacks, TE.Poor management, and TE.Delivery and installation and support the assumptions A.Physical security, Organizational security policy, and P.Secure management.

OE.Operating system reinforcement

This environmental security objective guarantees that the operating system is secure and reliable by removing unnecessary services and means, and reinforcing the vulnerabilities to support the assumption A.Operating system reinforcement and react against threats T.Failures and T.New vulnerability attacks.

OE.Unique connection point

This environmental security policy guarantees that all communications between internal and external networks are made by the TOE to react against the threat T.Bypass access and support he assumption A.Unique connection point.

OE.Vulnerability list renewal

This environmental security objective guarantees that the vulnerability database is updated and maintained to protect the system from external attacks using the internal network vulnerabilities and react the threat T.New vulnerability attacks.

OE.Operating system time—ST writer's additional items

This environmental security objective guarantees that the secure time source for audit is provided to support the assumption A.Operating system time.

OE.SSL certificate—ST writer's additional items

This environmental security objective guarantees that the private certificate for TOE's connection to SSL is provided in the TOE itself to support the assumption A.SSL certificate.

OE.Secure TOE external server—ST writer's additional items

This environmental security objective guarantees that the external server interoperating with the TOE is secure to support the assumption A.Secure TOE external server.

7.2. Rationale for Security Requirements

The rationale for security requirements prove that the described IT security requirements are eligible for satisfying security objectives and accordingly eligible to handle security problems.

The properties that the TOE must protect are computer resources and services in the operation, and general resources stored on computers. The property value is medium and the threat actors have low expert knowledge, resources, and motivation.

The security requirements described in the ST support each other and one or more security requirements satisfy security objectives.

7.2.1. Rationale for TOE Security Objectives

The following are rationale for TOE security requirements.

Security objectives / Security function / Requirement	O.Availability	O.Audit	O.Management	O.TSF data protection	O.Blocking abnormal packets	O.Blocking service decline attacks	O.Identification	O.Authentication	O.Information flow control
FAU_ARP.1		x							
FAU_GEN.1		x							
FAU_GEN.2		x							
FAU_SAA.1		x							
FAU_SAR.1		x							
FAU_SAR.3		x							
FAU_SEL.1		x							
FAU_STG.1		x							
FAU_STG.3		x							
FAU_STG.4		x							
FDP_IFC.1									x
FDP_IFF.1					x				x
FIA_AFL.1								x	
FIA_ATD.1 (1)		x			x	x	x		x
FIA_ATD.1 (2)		x					x		
FIA_UAU.1			x	x				x	
FIA_UAU.7								x	
FIA_UID.2 (1)		x			x	x	x		x
FIA_UID.2 (2)		x	x	x			x		
FMT_MOF.1	x		x						
FMT_MSA.1			x	x					x
FMT_MSA.3			x	x					x
FMT_MTD.1 (1)			x	x					
FMT_MTD.1 (2)			x	x					
FMT_MTD.1 (3)			x	x					
FMT_MTD.1 (4)			x	x					

FMT_MTD.1 (5)			X	X					
FMT_MTD.1 (6)			X	X					
FMT_MTD.2	X		X						
FMT_SMF.1			X						
FMT_SMR.1			X				X	X	
FPT_AMT.1	X			X					
FPT_FLS.1	X								X
FPT_RVM.1									X
FPT_SEP.1				X					X
FPT_STM.1		X							
FPT_TST.1	X			X					
FRU_FLT.1	X								X
FRU_RSA.1						X			
FTA_SSL.1				X					
FTA_SSL.3						X			
FTP_ITC.1			X					X	X

Table 25 TOE security objectives and security requirements

7.2.1.1. FAU (Security Audit)

FAU_ARP.1 Security alarms

This component satisfies the TOE security objectives O.Audit because it guarantees the capability to take reactions when any security violation is detected.

FAU_GEN.1 Audit data generation

This component satisfies the TOE security objectives O.Audit because it defines auditable events and guarantees the capability to generate audit records.

FAU_GEN.2 User identity association

This component defines auditable events and requests the user identification to track between audit records and user association so that it can support the TOE security target O.Audit.

FAU_SAA.1 Potential violation analysis

This component satisfies the TOE security objectives O.Audit because it guarantees the capability to locate

security violations by checking audit events.

FAU_SAR.1 Audit review

This component satisfies the TOE security objectives O.Audit because it guarantees the capability for the authorized administrator to review audit records.

FAU_SAR.3 Selectable audit review

This component satisfies the TOE security objectives O.Audit because it guarantees the capability to search and sort audit data based on the logical relations.

FAU_SEL.1 Selective audit

This component satisfies the TOE security objectives O.Audit because it guarantees the capability to include or exclude auditable events based on the attributes.

FAU_STG.1 Protected audit trails storage

This component satisfies the TOE security objectives O.Audit because it guarantees the capability to protect audit records from unauthorized modification and deletion.

FAU_STG.3 Reaction in case of possible audit data loss

This component satisfies the TOE security objectives O.Audit because it guarantees the reaction when audit trails exceed the predefined limit.

FAU_STG.4 Prevention of audit data loss

This component satisfies the TOE security objectives O.Audit because it guarantees the capability to take reactions in case of the audit storage saturation.

7.2.1.2. FDP (User Data Protection)

FDP_IFC.1 Subset information flow control

This component satisfies the TOE security objectives O.Information flow control because it guarantees that the security policies for TOE information flow control and the scope of security policies are defined.

FDP_IFF.1 Simple security attributes

This component satisfies the security objectives O.Blocking abnormal packets because it describes the reactions against implicit attacks.

7.2.1.3. FIA (Identification and Authentication)

FIA_AFL.1 Authentication failure processing

The component satisfies the TOE security objectives O.Audit because it guarantees the capability to take reaction when the predefined number is reached or exceeded.

FIA_ATD.1(1) User attribute definition(1)

This component requests to identify the identifier of the external IT entity as a computer IP address. The IP address satisfies O.Audit, O.Blocking abnormal packets, O.Blocking service decline attacks, O.Identification, and O.Information flow control because it becomes a ground to identify external IT entities for generating audit data, to determine whether the address is fake, to identify the service decline attack, and to control information flow.

FIA_ATD.1(2) User attribute definition(2)

This component satisfies O.Audit and O.Identification because it requests to identify the administrator.

FIA_UAU.1 Authentication

Because this component guarantees the capability to successfully identify the administrator, It satisfies the TOE security objectives O.Management, O.TSF data protection (TOE management and TSF data protection are possible after administrator authentication), and O.Audit.

FIA_UAU.7 Protected authentication feedback

This component satisfies the TOE security objectives O.Authentication because it guarantees that the predefined authentication feedback is provided to the administrator while authentication is proceeding.

FIA_UID.2(1) User identification before any action(1)

This component requests to identify the identifier of the external IT entity as a computer IP address. The IP address satisfies O.Audit, O.Blocking abnormal packets, O.Blocking service decline attacks, O.Identification, and O. Information flow control because it becomes a ground to identify external IT entities for generating audit data, to determine whether the address is fake, to identify the service decline attack, and to control information flow.

FIA_UID.2(2) User identification before any action(2)

This component satisfies O.Audit, O.Management, O.TSF data protection, and O.Identification because it requests to identify the administrator.

7.2.1.4. FMT (Security Management)

FMT_MOF.1 Management of security functions behavior

This component satisfies the TOE security objectives O.Availability and O.Management because it guarantees the capability for the administrator to manage security functions, and the availability in case of TOE failures.

FMT_MSA.1 Management of security attributes

This component satisfies the TOE security objectives O.Management, O.TSF data protection, and O.Information flow control because while performing TOE security functions, it guarantees that only the authorized can access the security attributes data (TSF data necessary for executing a TOE security function).

FMT_MSA.3 Static attribute initialization

This component satisfies the TOE security objectives O.Management, O.TSF data protection, and O.Information flow control because while performing TOE security functions, it guarantees that only the authorized can access the security attributes data (TSF data necessary for executing a TOE security function) for initialization.

FMT_MTD.1(1) Management of TSF data(1)

This component satisfies the TOE security objectives O.Management and O.TSF data protection because it guarantees that the authorized administrator can control the TSF data.

FMT_MTD.1(2) Management of TSF data(2)

This component satisfies the TOE security objectives O.Management and O.TSF data protection because it guarantees that the authorized administrator can control the TSF data.

FMT_MTD.1(3) Management of TSF data(3)

This component satisfies the TOE security objectives O.Management and O.TSF data protection because it guarantees that the authorized administrator can control the TSF data.

FMT_MTD.1(4) Management of TSF data(4)

This component satisfies the TOE security objectives O.Management and O.TSF data protection because it guarantees that the authorized administrator can control the TSF data.

FMT_MTD.1(5) Management of TSF data(5)

This component satisfies the TOE security objectives O.Management and O.TSF data protection because it guarantees that the authorized administrator can control the TSF data.

FMT_MTD.1(6) Management of TSF data(6)

This component satisfies the TOE security objectives O.Management and O.TSF data protection because it guarantees that the authorized administrator can control the TSF data.

FMT_MTD.2 Management of TSF data limits

This component satisfies the TOE security objectives O.Availability and O.Management because it guarantees the TOE availability that is enabled by taking reaction when the administrator manages the TSF data limits and the predefined limit is reached or exceeded.

FMT_SMF.1 Specification of management functions

This component satisfies O.Management because it requests to specify the management functions such as security attributes that the TSF must provide, TSF data, and security function.

FMT_SMR.1 Security roles

This component satisfies the TOE security objectives O.Management, O.Identification, and O.Authentication because it requests to restrict TOE security administrator roles to administrator roles.

7.2.1.5. FPT (TSF Protection)

FPT_AMT.1 Abstract machine test

This component satisfies the TOE security objectives O.Availability and O.TSF data protection because it performs a series of tests to show correct operation of TSF underlying abstract machine.

FPT_FLS.1 Failure with preservation of secure state

This component satisfies the TOE security objectives O.Availability and O.Information flow control because it guarantees to perform the information flow control function by maintaining the main security function's secure state even in TOE failures.

FPT_RVM.1 Non-bypassability of the TSP

This component satisfies the TOE security target O.Information flow control because it blocks information bypass by guaranteeing that the function that performs the TSP is called and works correctly.

FPT_SEP.1 TSF domain separation

This component satisfies the TOE security objectives O.TSF data protection and O.Information flow control because it guarantees to maintain the security area for TSF self execution separated from the unauthorized subject.

FPT_STM.1 Reliable time stamps

This component provides reliable time stamps that the TSF can use. The generated time satisfies the TOE security target O. Audit because it guarantees to record sequential audit events when generating audit data.

FPT_TST.1 TSF self test

This component satisfies the TOE security objectives O.Availability and O.TSF data protection because it requests to promptly prevent or detect TOE failures by self testing and verifying the integrity of TSF data and TSF execution codes.

7.2.1.6. FRU (Resource Utilization)

FRU_FLT.1 Fault tolerance: Partial application

This component satisfies the TOE security objectives O.Availability and O.Information flow control because it guarantees to perform the information protection control function by requesting main security function even in TOE failures.

FRU_RSA.1 Maximum quotas

This component satisfies the TOE security objectives O.Blocking service decline attacks because it blocks service decline attacks by requesting to restrict the resource utilization quotas for user-initiated TOE protection resources.

7.2.1.7. FTA (TOE Access)

FTA_SSL.1 TSF-initiated session locking

This component satisfies O.TSF data protection because the TOE requests to lock the authorized session after the inactive period of the authorized administrator.

FTA_SSL.3 TSF-initiated termination

This component satisfies O.Blocking service decline attacks because it is used to acquire the network service availability due to the request from the external IT entities for the session termination of internal computers after a certain period of time.

7.2.1.8. FTP (Secure Route/Channel)

FTP_ITC.1 Inter-TSF trusted channel

This component satisfies O.Management, O.Authentication, and O.TSF data protection because it requests to establish a secure channel during communication between vulnerability data servers when the administrator locally or remotely manages the TOE.

7.2.2. Rationale for TOE Assurance Requirements

This ST is written on the EAL4 (assurance level of the network intrusion protection system's protection profile)

basis.

7.2.3. Rationale for IT Environmental Security Requirements

Security objectives / Security requirements	OE.Operating system time	OE.SSLCertificate
FPT_STM.1	X	
FTP_ITC.1		X

FPT_STM.1 Reliable time stamps

The TOE provides reliable time stamps through the reliable external time stamp synchronization call function (NTP). Accordingly, it satisfies the requirements that correspond to the TOE security target OE.Operating system time.

FTP_ITC.1 Inter-TSF trusted channel

The TOE provides the secure security channel management function while the administrator is connecting the TOE to the management environment by the administrator program. Accordingly, it satisfies the requirements that correspond to the TOE security target OE. SSL certificate.

7.3. Rationale for Dependencies

7.3.1. Dependencies for TOE Security Requirements

The following table shows the dependencies of functional components.

FAU_GEN.2, FIA_UAU.1, and FMT_SMR.1 has dependencies with FIA_UID.1 but it is satisfied by FIA_UID.2 that is hierarchical to FIA_UID.1.

No.	Functional component	Dependency	Reference No.
1	FAU_ARP.1	FAU_SAA.1	4
2	FAU_GEN.1	FPT_STM.1	29
3	FAU_GEN.2	FAU_GEN.1	2
		FIA_UID.1	17
4	FAU_SAA.1	FAU_GEN.1	2
5	FAU_SAR.1	FAU_GEN.1	2
6	FAU_SAR.3	FAU_SAR.1	5
7	FAU_SEL.1	FAU_GEN.1	2
		FMT_MTD.1	21
8	FAU_STG.1	FAU_GEN.1	2
9	FAU_STG.3	FAU_STG.1	8
10	FAU_STG.4	FAU_STG.1	8
11	FDP_IFC.1	FDP_IFF.1	12
12	FDP_IFF.1	FDP_IFC.1	11
		FMT_MSA.3	20
13	FIA_AFL.1	FIA_UAU.1	15
14	FIA_ATD.1	-	-
15	FIA_UAU.1	FIA_UID.1	17
16	FIA_UAU.7	FIA_UAU.1	15
17	FIA_UID.2	-	-
18	FMT_MOF.1	FMT_SMF.1	23
		FMT_SMR.1	24
19	FMT_MSA.1	[FDP_ACC.1 or FDP_IFC.1]	11
		FMT_SMF.1	23
		FMT_SMR.1	24
20	FMT_MSA.3	FMT_MSA.1	19
		FMT_SMR.1	24
21	FMT_MTD.1	FMT_SMF.1	23
		FMT_SMR.1	24
22	FMT_MTD.2	FMT_MTD.1	21
		FMT_SMR.1	24
23	FMT_SMF.1	-	-
24	FMT_SMR.1	FIA_UID.1	17

25	FPT_AMT.1	-	-
26	FPT_FLS.1	ADV_SPM.1	Assurance requirements
27	FPT_RVM.1	-	-
28	FPT_SEP.1	-	-
29	FPT_STM.1	-	-
30	FPT_TST.1	FPT_AMT.1	25
31	FRU_FLT.1	FPT_FLS.1	26
32	FRU_RSA.1	-	-
33	FTA_SSL.1	FIA_UAU.1	15
34	FTA_SSL.3	-	-
35	FTP_ITC.1	-	-

Table 26 Dependencies for functional components

7.3.2. Dependencies for TOE Assurance Requirements

Each assurance package dependency provided by the common evaluation criteria of information protection system is already satisfied.

7.4. Rationale for the Strength of Security Functions

The information that the TOE in the ST must protect is a general material, has a medium property value, and the threat actor has low level of expert knowledge, resources, and motivation. The common evaluation methodology recommends that the security function should be set with at least a minimum strength of function—medium. This ST sets the strength of function—medium because the calculation result by the method in CEM Appendix is 24. (Refer to the vulnerability analysis for calculation method)

The security function that the minimum security functional strength is declared in this ST conforms to the corresponding functional components through Table 28. The specified functional strength declaration and the minimum functional strength satisfy the TOE security objectives through Table 27.

Functional component	Functional component name	Security function
FIA_UAU.1	Authentication	TOE user identification and authentication (User_InA) - ID/Password mechanism - One-time password mechanism

Table 27 Functional components that correspond to the declared security strengths.

Strength of function	Security function	Security target
Medium	TOE user identification and authentication (User_InA)	Identification and authentication

Table 28 TOE security objectives that correspond to the declared security functional strength (medium)

7.5. Rationale for TOE Summary Specification

The rationale of the TOE summary specification describe that the TOE security functions satisfy the IT security requirements.

7.5.1. Security Functions for IT Security Requirements

The following table shows that the TOE-enabled security functions satisfy all IT security functions that the PP (accommodate TOE security functions) requires.

IT security requirements	Security function
FAU_ARP.1	Security auditing, and audit data generation and protection (Audit) Detection of illegal accesses and attacks (IPS)
FAU_GEN.1	
FAU_GEN.2	
FAU_SAA.1	
FAU_SAR.1	Security auditing, and audit data generation and protection (Audit)
FAU_SAR.3	
FAU_SEL.1	Security auditing, and audit data generation and protection (Audit) Detection of illegal accesses and attacks (IPS)
FAU_STG.1	
FAU_STG.3	
FAU_STG.4	
FDP_IFC.1	Protection of internal property and information from illegal accesses (Firewall) Detection of illegal accesses and attacks (IPS)
FDP_IFF.1	
FIA_AFL.1	TOE user identification and authentication (UI&AD)
FIA_ATD.1 (1)	
FIA_ATD.1 (2)	
FIA_UAU.1	
FIA_UAU.7	
FIA_UID.2(1)	
FIA_UID.2(2)	
FMT_MOF.1	Security management (SEC_MAN)
FMT_MSA.1	
FMT_MSA.3	

FMT_MTD.1 (1)	
FMT_MTD.1 (2)	
FMT_MTD.1 (3)	
FMT_MTD.1 (4)	
FMT_MTD.1 (5)	
FMT_MTD.1 (6)	
FMT_MTD.2	
FMT_SMF.1	
FMT_SMR.1	
FPT_AMT.1	TSF stability (TSF_SECURER)
FPT_FLS.1	Process monitoring (Mon_Process)
FPT_RVM.1	TSF stability (TSF_SECURER)
FPT_SEP.1	
FPT_STM.1	Security auditing, and audit data generation and protection (Audit) Detection of illegal accesses and attacks (IPS)
FPT_TST.1	TSF stability (TSF_SECURER)
FRU_FLT.1	Process monitoring (Mon_Process)
FRU_RSA.1	Detection of illegal accesses and attacks (IPS)
FTA_SSL.1	TOE user identification and authentication (UI&AD)
FTA_SSL.3	
FTP_ITC.1	TSF stability (TSF_SECURER)

Table 29 Security functions that correspond to security requirements

The following IT security functions satisfy the TOE security functional requirements.

An audit event is generated using the predefined potential security violation rules after associating the identity of the user who has issued an audit event. FAU_ARP.1, FAU_GEN.1, FAU_GEN.2, and FAU_SAA.1 are satisfied using the functions like the security audit and audit data generation and protection (Audit) that detects audit events and potential security violations, and the detection of illegal accesses and attacks (IPS).

The selective review of audit data is enabled by the security audit and audit data generation and protection (Audit) function and FAU_SAR.1 and FAU_SAR.3 are satisfied because only the authorized administrator can read all audit data.

FAU_SEL.1, FAU_STG.1, FAU_STG.3, and FAU_STG.4 are satisfied because reactions in case of selective audit, audit trails protection, possible data loss, and audit data loss protection are enabled through protection of

internal properties and information (Firewall) and the detection of illegal accesses and attacks (IPS) functions.

FDP_IFC.1 and FDP_IFF.1 are satisfied because the subset information flow control and NXG IPS security policies are enabled through protection of internal properties and information (Firewall) and the detection of illegal accesses and attacks (IPS) functions.

FIA_AFL.1 is satisfied because the authentication failure handling is enabled through the TOE user identification and authentication (UI&AD) function. FIA_ATD.1 (1) and FIA_ATD.1 (2) are satisfied because the user attributes can be defined. FIA_UAU.1, FIA_UAU.7, FIA_UID.2 (1), FIA_UID.2 (2) are satisfied because the users are authenticated as IT entities and administrators.

FMT_MOF.1 is satisfied because the authorized administrator can assign restricted functions on the functional list to users through the security management (SEC_MAN).

FMT_MSA.1 and FMT_MSA.3 are satisfied because the authorized administrator can assign the arithmetic capability, restricted default values, and selective initial values through the NXG IPS policy of the security management (SEC_MAN).

FMT_MTD.1 (1), FMT_MTD.1 (2), FMT_MTD.1 (3), FMT_MTD.1 (4), FMT_MTD.1 (5), FMT_MTD.1 (6), and FMT_MTD.2 are satisfied because the TSF data management and TSF data limits definition are possible through the security management (SEC_MAN). .

FMT_SMR.1 is satisfied because the security roles are maintained through the administrator authorized by the security management (SEC_MAN) and the group-initiated access.

FPT_AMT.1 is satisfied because the underlying abstract machine test is performed through the TSF stability (TSF_SECURER).

FPT_FLS.1 is satisfied because the secure state is maintained when the TOE failures happen through the process monitoring (Mon_Process).

FPT_RVM.1 and FPT_SEP.1 are satisfied because the TSP is constricted by the TSF stability (TSF_SECURER) and the area for security function is separated.

FPT_STM.1 is satisfied because reliable time stamps to be used by TSF are provided through the audit data generation and protection (Audit_Gen_Protect) and the reactions against illegal accesses and attacks (IPS_React).

FPT_TST.1 is satisfied because a self test is performed to demonstrate the TSF normal operation through the security function stability self test (Secui_Self_Test).

FRU_FLT.1 is satisfied because the users can use the network services through the process monitoring (Mon_Process) even in TOE failures.

FRU_RSA.1 is satisfied because the maximum value for resource utilization for each network traffic is set through Detection of illegal accesses and attacks (IPS).

FTA_SSL.1 and FTA_SSL.3 are satisfied because the session can be locked or terminated when the authorized administrator does not use the TSF in the inactive period through the TOE user identification and authentication (UI&AD).

FTP_ITC.1 is satisfied because the SSL session is linked for the administrator to connect the administrator program through the TSF stability (TSF_SECURER).

7.6. Rationale for TOE Assurance Requirements

The assurance means for each assurance component are satisfied through the following table.

Assurance class	Assurance component		Assurance measures
Configuration management	ACM_AUT.1	Partial CM automation	CM documentation
	ACM_CAP.4	Creation support and acceptance procedure	
	ACM_SCP.2	Problem tracking CM coverage	
Delivery and operation	ADO_DEL.2	Detection of modification	Delivery and installation guidance
	ADO_IGS.1	Installation, generation, and start-up procedures	
Development	ADV_FSP.2	Fully defined external interfaces	Functional specification
	ADV_HLD.2	Security enforcing high-level design	High-level design specification
	ADV_IMP.1	Expression of implementation for partial TSF	Verification specification, Source code Life-cycle support
	ADV_LLD.1	Descriptive design specification	Design specification
	ADV_RCR.1	Informal correspondence demonstration	Verification specification
	ADV_SPM.1	Informal TOE security policy model	Security policy model
Guidance documents	AGD_ADM.1	Administrator guidance	Administrator guidance
	AGD_USR.1	User guidance	None
Life cycle support	ALC_DVS.1	Identification of security measures	Life-cycle support
	ALC_LCD.1	Life-cycle model defined by developers	
	ALC_TAT.1	Well-defined development tools	
Tests	ATE_COV.2	Analysis of coverage	Tests
	ATE_DPT.1	Basic design test	
	ATE_FUN.1	Functional testing	
	ATE_IND.2	Independent testing: sample	
Vulnerability assessment	AVA_MSU.2	Verification of administrator guidance analysis	Analysis of vulnerabilities and misuses
	AVA_SOF.1	Evaluation for strength of the TOE security functions	
	AVA_VLA.2	Independent vulnerability analysis	

Table 30 Rationale for TOE assurance requirements

The TOE assurance requirements are satisfied using the following assurance measures.

ACM_AUT.1 Partial CM automation

Assurance measures satisfy this because the CM documentation explains about automated measures required for generating TOE.

ACM_CAP.4 Generation support and acceptance procedures

The assurance measures satisfy this because the CM documentation explains about the generation support and acceptance procedure in the CM procedure.

ACM_SCP.2 Problem tracking CM coverage

The assurance measures satisfy this because the CM documentation provides the scope of problem tracking configuration management.

ADO_DEL.2 Detection of modification

The assurance measures satisfy this because the delivery and installation guidance provides the method for confirming the modification detection.

ADO_IGS.1 Installation, generation, and start-up

The assurance measures satisfy this because the delivery and installation guidance provides the installation, generation, and start-up procedure.

ADV_FSP.2 Fully defined external interfaces

The assurance measures satisfy this because the functional specification provides the fully-defined external interfaces for TSF and TSFI.

ADV_HLD.2 High-level design that has the security functions and non-security functions

The assurance measures satisfy this because the high-level design specification provides the design that has security functions and non-security functions for each sub-system of the TSF.

ADV_IMP.1 Subset of the implementation of the TSF

The assurance measures satisfy this because the verification specification and source codes provide partial

expression of the TSF implementation that generates the TOE.

ADV_LLD.1 Descriptive design specification

The assurance measures satisfy this because the design specification provides the descriptive design specification for each TSF module.

ADV_RCR.1 Informal correspondence demonstration

The assurance measures satisfy this because the verification specification provides the analysis of correspondence between ST-TSS and FSP, FSP and HLD, HLD and LLD, LLD and IMP.

ADV_SPM.1 Informal TOE security policy model

The assurance measures satisfy this because the security policy model provides the informal security policy model.

AGD_ADM.1 Administrator guidance

The assurance measures satisfy this because the administrator guidance provides the guide for administrators.

AGD_USR.1 User guidance

This TOE has no general users so there is no user guidance, which has to be substituted for the administrator guidance.

ALC_DVS.1 Identification of security measures

The assurance measures satisfy this because the life-cycle support document provides the development security information and security measures identification.

ALC_LCD.1 Developer defined life-cycle model

The assurance measures satisfy this because the life-cycle model provides the developer-defined life-cycle model.

ALC_TAT.1 Well-defined development tools

The assurance measures satisfy this because the life-cycle support document explains about well-defined development tools.

ATE_COV.2 Analysis of coverage

The assurance measures satisfy this because the test book provides the analysis of coverage.

ATE_DPT.1 Testing: high-level design

The assurance measures satisfy this because the test book provides the test content on the high-level design.

ATE_FUN.1 Functional testing

The assurance measures satisfy this because the test book provides the test content on the functional test.

ATE_IND.2 Independent testing: sample

The assurance measures satisfy this because the independent testing can be done based on this test book.

AVA_MSU.2 Validation of analysis

The assurance measures satisfy this because the analysis of vulnerabilities and misuses provides the misuse analysis on the administrator guidance.

AVA_SOF.1 Strength of the TOE security function evaluation

The assurance measures satisfy this because the analysis of vulnerabilities and misuses provides the analysis on the strength of TOE security functions.

AVA_VLA.2 Independent vulnerability analysis

The assurance measures satisfy this because the evaluator can analyze the independent vulnerabilities based on the analysis on vulnerabilities and misuses.

8. Protection Profile (PP) Claims

8.1. Protection Profile Identification

Network Intrusion Protection System Protection Profile V1.1

8.2. Protection Profile Reestablishment

This ST adds separate security objectives and IT security requirements in the protection profile provided in the 8.1 section and tailors the operation to the following IT security requirements.

Functional component	Name	Operation
FAU_ARP.1	Security alarms	Assignment
FAU_GEN.1	Audit data generation	Assignment
FAU_SAA.1	Potential violation analysis	Assignment
FAU_SAR.1	Audit review	Assignment
FAU_SAR.3	Selectable audit review	Assignment, selection
FAU_SEL.1	Selective audit	Assignment, selection
FAU_STG.1	Protected audit trails storage	Selection
FAU_STG.3	Reaction in case of possible audit data loss	Assignment
FAU_STG.4	Prevention of audit data loss	Selection
FDP_IFC.1	Subset information flow control	Assignment
FDP_IFF.1	Simple security attributes	Assignment
FIA_AFL.1	Authentication failure processing	Assignment, selection
FIA_UAU.1	Authentication	Assignment
FIA_UAU.7	Authentication feedback protection	Assignment
FMT_MOF.1	Security function management	Assignment, selection
FMT_MSA.1	Management of security attributes	Assignment
FMT_MSA.3	Static attribute initialization	Assignment, selection
FMT_MTD.1	Management of TSF data	Assignment, selection, repetition
FMT_MTD.2	Management of TSF data limits	Assignment
FMT_SMF.1	Specification of management functions	Assignment
FPT_AMT.1	Abstract machine test	Selection

FPT_FLS.1	Keeping the secure state during system failure	Assignment
FPT_TST.1	TSF self test	Assignment, selection
FRU_FLT.1	Resistance to failure: Partial application	Assignment
FRU_RSA.1	Maximum assigned value	Selection
FTA_SSL.1	TSF-initiated session locking	Assignment
FTA_SSL.3	TSF-initiated session termination	Assignment
FTP_ITC.1	Inter-TSF trusted channel	Selection, assignment

FDP_IFC.1 (1) and FDP_IFC.1 (2) Integration of subset information flow control into FDP_IFC.1

The user guidance is not required because the item on the assurance requirement AGD_USR.1 has no security functions that are used by general users.

8.3. Rationale

This ST provides the description on the security environment, security objectives, rationale for IT security requirements, and protection profile identification in each section.

	Item	Remark
Assumtions	A.Operating system time	Addition
	A.SSL certificate	
	A.Secure TOE external server	
Security objectives for the environment	OE.Operating system time	Addition
	OE.SSL certificate	
	OE.Secure TOE external server	