IceWall SSO

Version 10.0

User's Manual

August 2010

Printed in Japan

HP Part No. B1544-96000

Rev.111007A

Notice

The information contained in this document is subject to change without notice.

Meticulous care has been taken in the preparation of this document, however, if a questionable or erroneous item, or an omission of content is found, please contact us.

Hewlett-Packard Japan, Ltd. shall not be liable for errors contained herein or for incidental or consequential damage in connection with the furnishing, performance or use of this document.

This manual and media, such as the CD-ROM supplied as a part of the product's package, are only to be used with this product.

– Table of Contents –

1 Introduction 1
1.1 Version designations in the text 1
2 IceWall SSO Overview 2
2.1 Individual module functions 2
2.1.1 Forwarder 2
2.1.2 Authentication Module 2
2.2 Basic functions 3
2.2.1 Login function 3
2.2.2 Logout function 3
2.2.3 Access control function 3
2.2.4 Password change function 3
2.2.5 Session management function 3
2.2.6 Information inheritance function 4
2.2.7 URL conversion function 4
2.2.8 Keyword conversion function 5
2.2.9 Reverse proxy function 5
2.2.10 Access logging function 5
2.2.11 Cross site scripting prevention function 5
2.2.12 Automatic Form Authentication function 5
2.2.13 Simple portal function 5
2.2.14 SSL communication function (optional) 5
2.2.15 Failover function (optional) (10.0) 5
2.2.16 Authentication Server load balancing function (optional) [10.0] 6
2.2.17 Client authentication function (optional) 6
2.2.18 Windows integrated authentication function (support for Windows Kerberos) (optional) 6
2.2.19 Agent function (optional) 6
2.3 Basic transactions 7
2.3.1 Login function 7
2.3.2 Content display of Backend Web Servers 9
2.3.3 Logout function 11
2.3.4 Password change function (10.0) 13
3 Basic Operations 16
3.1 System startup 16
3.2 System shutdown 16
3.3 Editing configuration files 16
3.3.1 Forwarder configuration file 16
3.3.2 Authentication Module configuration file 17
3.3.3 Reloading/synchronizing each Authentication Module configuration file (10.0) ... 17
3.3.4 Updating the Authentication Module configuration file (10.0) 19
3.3.5 Notes on configuration file descriptions 20
3.4 Authentication Module command process exit status for the Authentication Module [10.0] 20

3.5 System configuration example 22
3.6 Editing the Forwarder configuration file 24
3.6.1 Editing the Forwarder configuration file (dfw.conf) 24
3.6.2 Editing the host configuration files 25
3.6.3 Editing the HTML configuration file (html.conf) 26
3.7 Editing the Authentication Module configuration file 26
3.7.1 Editing the Authentication Module configuration file (cert.conf) 26
3.7.2 Editing the group configuration file (cert.grp) 33
3.7.3 Editing the access control file (cert.acl) 34
3.7.4 Editing the Authentication DB column information file (dbattr.conf) 34
3.7.5 Editing the log column information file (logdbattr.conf) 34
3.7.6 Editing the forbidden password configuration file (pwdforbid.conf) 35
3.7.7 Restarting the Authentication Module 35
3.8 Log out all users function 10.0 35
3.9 Handling Authentication Module maintenance 10.0 35
3.9.1 Pre-blocking 36
3.9.2 Main blocking 36
4 Advanced Configuration 37
4.1 Access control 37
4.1.1 Concept of access control 37
4.1.2 Creating various groups 37
4.1.3 Various access control configurations 38
4.1.4 Notes on the sequential order of description for access control configuration 40
4.2 Header control function and cookie control function 43
4.2.1 Header control when sending to the Backend Web Server 43
4.2.2 Cookie control when sending to the Backend Web Server 43
4.2.3 Header control when sending to the browser 44
4.2.4 Header control for the response from the Backend Web Server 44
4.3 Basic authentication 44
4.3.1 Basic authentication configuration 44
4.3.2 Basic authentication configuration for reverse proxy mode 45
4.3.3 Notes when using ICP 2.0 communication 45
4.4 Information inheritance configuration 45
4.5 Request control based on the access path 46
4.5.1 Performing request control 46
4.5.2 Request control from a specific Forwarder or agent 47
4.5.3 Forbidding requests from a specific Forwarder or agent 47
4.5.4 Restricting the user information sent to specific Forwarders or agents 48
4.5.5 Restricting access paths based on whether a client certificate is used 48
4.6 Agent control based on the request source 49
4.6.1 Agent control based on the IP address 49
4.6.2 Agent control based on the host name 49
4.6.3 Agent control based on the domain name 50
4.7 Forbidding connections from clients by request type 10.0 50
4.8 Password configuration 50

4.8.1 Using the password change function 50
4.8.2 Changing the password policy 10.0 51
4.8.3 To disallow changing a password to a previously used password 53
4.8.4 Defining strings not to be used as passwords 10.0 53
4.9 UserExit routine 54
4.10 IceWall SSO multi-tier architecture 54
4.10.1 Prerequisite conditions for a multi-tier architecture 55
4.11 Changing the names of authentication cookies 55
4.11.1 Configuring keywords for handling authentication information 56
4.11.2 Restrictions regarding a multi-tier architecture 56
4.12 Handling HTTP-compressed content 57
4.12.1 Configuring the transfer of HTTP-compressed content 58
4.12.2 Restrictions on HTTP-compressed content 58
4.13 Unicode conversion of user data (for LDAP, OpenLDAP, NED, and MSAD editions only) 58
4.13.1 Configuring the character encoding conversion 58
4.13.2 Restrictions on character encoding conversion 59
4.14 Sending, after login, POST data that was sent before login 59
4.14.1 Using POST data inheritance 60
4.14.2 Restricting the size of the inherited POST data 60
4.14.3 Setting the parameter name to use when inheriting POST data 60
4.14.4 Encrypting inherited POST data 61
4.15 Using IceWall Cert Protocol (ICP) 2.0 61
4.15.1 Configuring the communication to ICP 2.0 61
4.15.2 Identifying the source of an ICP 2.0 request 61
4.16 Extending the ICP 2.0 communication message encryption library 10.0 62
4.17 ICP 2.0 HTTP support 10.0 62
4.17.1 HTTP requests 62
4.17.2 HTTP responses 63
4.17.3 Authentication Module HTTP send/receive settings 64
4.18 Specify the number of request threads that do not require DB connections (option) 10.0 64
4.19 Password encryption library 10.0 65
4.19.1 Specifying the password encryption method 65
4.19.2 Merging the password encryption libraries 66
4.19.3 SHA256 encryption method 66
4.20 Other useful functions 66
4.20.1 Operation in reverse proxy mode 66
4.20.2 Configuring authentication cookie attributes 66
4.20.3 Configuring a no-authentication URL/extension 67
4.20.4 URL-Cookie authentication method 67
4.20.5 Replacing error pages 68
4.20.6 Support for load balancing systems 68
4.20.7 Setting the maximum number of simultaneous user logins 68
4.20.8 Controlling response header URL conversion (for virtual hosts) 69

4.20.9 Function to configure the method of acquiring request path information 69
4.20.10 Changing the maximum number of allowable user logins 10.0 69
4.20.11 Performing an Authentication Module alive check 10.0 71
4.20.12 Connection target Authentication Module assignment function 10.0 72
4.20.13 Retrieving a running Authentication Module's configuration information 10.0 73
4.20.14 Linking IceWall SSO group and Active Directory group attributes 10.0 73
5 High Availability 74
5.1 IceWall server redundancy 74
5.2 Authentication Server and Backend Web Server redundancy 74
5.3 Authentication DB redundancy 75
5.4 Failover recovery function 10.0 75
5.4.1 Enabling the failback function 75
5.4.2 Enabling the failback function (Forwarder) 75
5.5 Authentication Module non-stop maintenance function 10.0 76
5.5.1 Changing a configuration that uses the Authentication Module's non-stop maintenance function 77
5.5.2 Updating a library by using the non-stop maintenance function for the Authentication Module. 78
5.5.3 Remarks on changing a configuration by using the Authentication Module's non-stop maintenance function 79
5.6 Transaction IDs 10.0 98
5.6.1 How to make the configuration 98
5.6.2 Logs 99
5.6.3 Headers 99
5.6.4 Special keywords 99
6 Security 100
6.1 Sniffing prevention 100
6.2 High-strength session IDs 10.0 100
6.3 Encryption of Authentication DB reference columns 10.0 101
6.4 Session ID encryption 101
6.5 Spoofing prevention 101
6.6 Cross site scripting prevention 101
6.6.1 Configuring a filter for requests sent to Backend Web Servers 102
6.6.2 Configuring a filter for content received from Backend Web Servers 102
6.6.3 Setting a filter by using a designated keyword 102
6.6.4 Filter dry-run function 10.0 102
6.7 Provisions against buffer overflows 103
6.8 Other Topics 103
6.9 Improving security 103
6.9.1 Common configuration for the user ID error and password ID error pages used at login 104
6.9.2 Prohibiting substitution of special keywords with user information 104
6.9.3 Changing templates 104
7 Performance Tuning 105
7.1 Forwarder performance 105

7.1.1 Frequent transactions exceeding 64 KB 105
7.1.2 If the non-text-based (binary) content in the browsing content is too large 105
7.1.3 Keep-Alive between Forwarder and a Backend Web Server 105
7.1.4 Adjusting the Forwarder disconnection waiting time 106
7.1.5 Adjusting response timeout between Forwarder and the Backend Web Server 106
7.1.6 Adjusting the connection timeout between Forwarder and the Backend Web Server 10.0 107
7.1.7 Adjusting connection timeout between Forwarder and the Authentication Module 10.0 108
7.1.8 Adjusting response timeout between Forwarder and the Authentication Module 108
7.2 Authentication Module performance 109
7.2.1 Adjusting the request receive timeout 112
7.2.2 Receive timeout from the Authentication DB 112
7.2.3 Adjusting the timeout for connections to the Authentication Module 10.0 113
7.3 Preventing Authentication Module performance deterioration 114
7.4 Parameter calculation method 114
7.4.1 Calculating MAXDBCONNECT 114
7.4.2 Calculating ACCTHREAD 115
7.4.3 Calculating MAXREQTHREAD 115
7.4.4 Calculating REQQUESIZE 115
7.4.5 Calculating MAXREPTHREAD 116
7.4.6 Calculating REPQUESIZE 116
7.4.7 Calculating THREADSTACKSIZE 116
7.4.8 Calculating kernel parameters 116
7.4.9 Authentication Module and Authentication Server default values 116
7.4.10 Calculating the maximum number of file descriptors 117
8 IPv6 Support 10.0 118
8.1 IPv6 specifications for IceWall SSO 118
8.1.1 Prerequisite conditions 118
8.1.2 IP address format in IPv6 118
8.1.3 Precedence of IP versions during communication 119
8.2 IPv6 support in Forwarder 120
8.2.1 IPv6 support in the Forwarder configuration file (dfw.conf) 120
8.2.2 IPv6 support for parameter values in the host configuration file (any file name) 122
8.2.3 IPv6 support for parameter values in the HTML configuration file (html.conf) 123
8.2.4 Restrictions on absolute URLs in Backend Web Server content 123
8.3 IPV6 support in the Authentication Module 124
8.3.1 Authentication Module reception of IPv6 requests 124
8.3.2 IPV6 support for parameter values in the Authentication Module configuration file (cert.conf) 124
8.3.3 IPV6 support for configuration parameters of the access control file (cert.acl) 125

8.3.4 IPv6 support for parameter values in the group configuration file (cert.grp) 126
8.3.5 IPV6 support for parameter values in the request control configuration file (request.acl) 126
8.3.6 IPv6 support for various commands in the Authentication Module 126
9 Restrictions on Various Authentication DBs 128
9.1 Common restrictions for Authentication DBs 128
9.2 Restrictions for an ORACLE Authentication DB 128
9.3 Restrictions for an LDAP, OpenLDAP, or NED Authentication DB 128
9.4 Restrictions for an MSAD Authentication DB 128
9.5 Restrictions on a MySQL Authentication DB 129
9.6 Restrictions for a CSV Authentication DB 129
9.6.1 CSV file specifications 129
10 Log Files 130
10.1 Access logs 130
10.1.1 Forwarder access log format (10.0) 130
10.1.2 Authentication Module access log format (10.0) 131
10.2 Error logs 133
10.2.1 Error log format 133
10.3 Audit log 133
10.4 Security log 134
10.4.1 Security log format 134
10.5 Information log 134
10.6 Performance log 134
10.7 Trace time log [10.0] 134

1 Introduction

This manual describes IceWall SSO functions and includes configuration examples to provide system administrators with the information necessary for installing, operating, and managing IceWall SSO.

Also refer to the “IceWall SSO Reference Manual” for information on content covered in this manual.
Note that contents of this manual have been based on the assumption that the operating environment has been verified according to instructions in the “IceWall SSO Sample Setup Guide” and other documents.

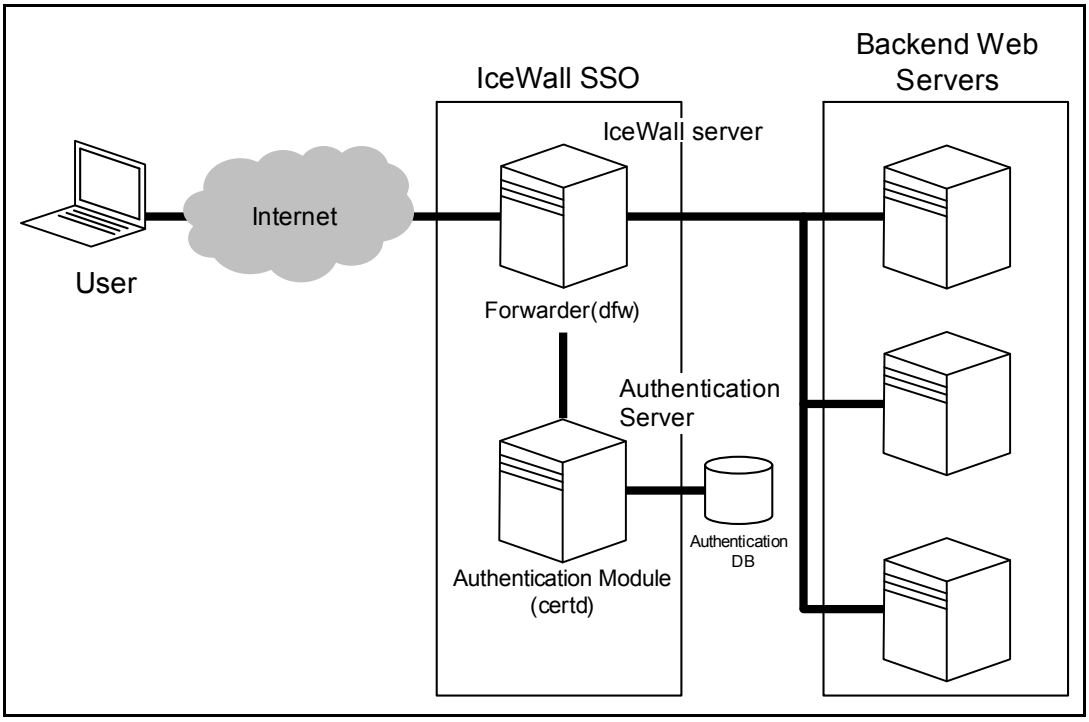
1.1 Version designations in the text

The table below gives the meanings of the version designations added to the text.

Designation	Meaning
10.0	An item added with the version enclosed in the square. In this case, the designation indicates the item was added with 10.0.
(10.0)	An item where the specification was changed or function added with the version enclosed in the oval. In this case, the designation indicates a specification change or added function with 10.0.

2 IceWall SSO Overview

IceWall SSO is middleware developed to provide single sign on authentication. It consists of two modules: Forwarder (dfw) and the Authentication Module (certd). This chapter describes the main functions and basic transactions of IceWall SSO.



IceWall SSO Diagram

2.1 Individual module functions

2.1.1 Forwarder

Forwarder is a CGI process that receives requests from users and forwards them to web servers. Forwarder performs URL conversion and keyword conversion to rewrite URLs included in web content to URLs that go through Forwarder. Forwarder then sends the web content to the user.
The server that runs Forwarder is called "IceWall server," and a web server that Forwarder sends requests to is called the "backend server."

2.1.2 Authentication Module

The Authentication Module receives requests from Forwarder and runs authentication queries against a database (or directory service). It is a daemon process that does authorization for access control and other functions.
The server that runs the Authentication Module is called the "Authentication Server" and the database used by the Authentication Module is called the "Authentication DB."

2.2 Basic functions

Basic IceWall SSO functions are described below.

2.2.1 Login function

This function is used when users begin to access content on a Backend Web Server. It performs authentication with user IDs and passwords in connection with the Authentication DB.

2.2.2 Logout function

This function allows the user to terminate access to Backend Web Servers. When this function is used, the user can no longer access the Backend Web Servers. (To access the Backend Web Servers again, the user must use the login function.)

2.2.3 Access control function

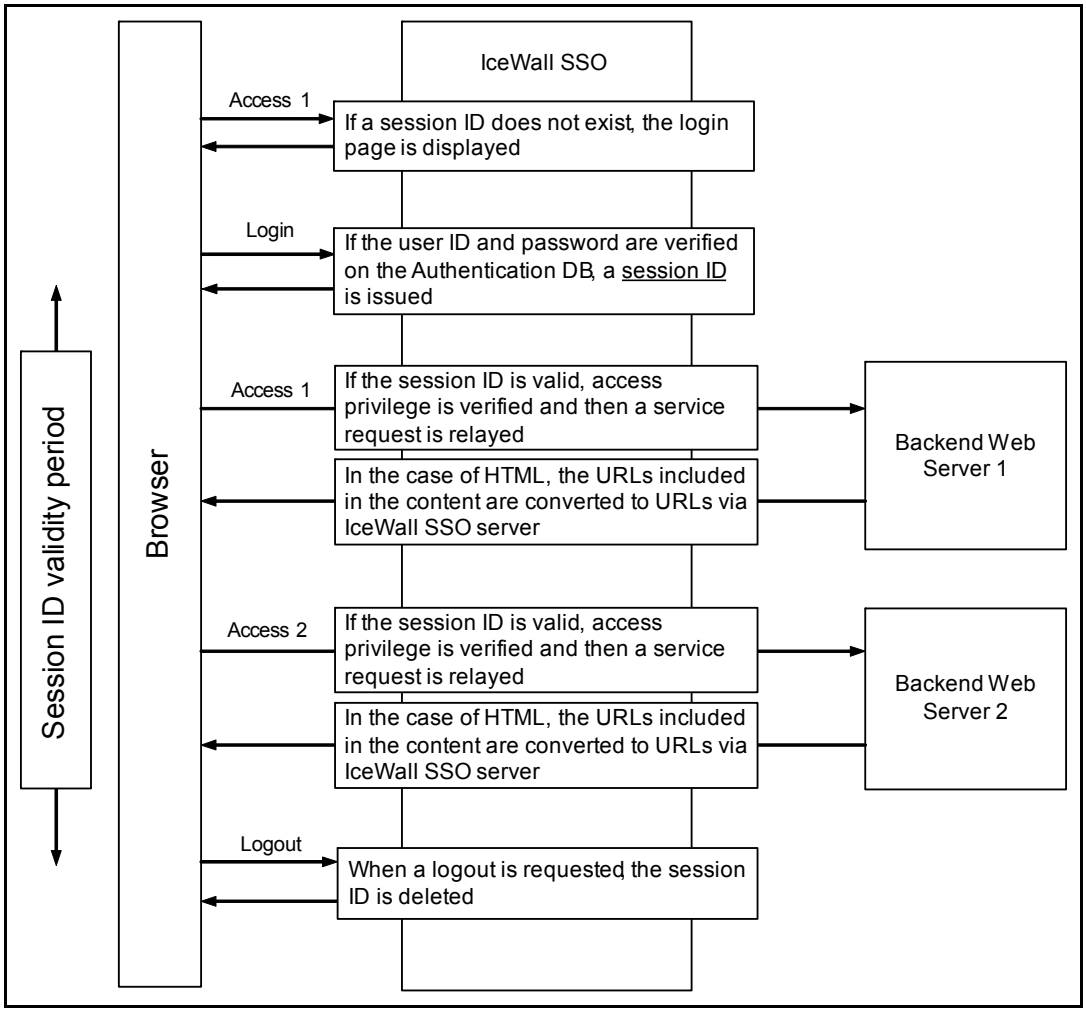
This function controls a user's access to content available on Backend Web Servers, based on the authorization information of the groups to which the authenticated user belongs. Access to certain directories or specified file names can be controlled.

2.2.4 Password change function

This function allows users to change their own passwords stored on the Authentication DB. Detailed password policy settings can be configured via the configuration file.

2.2.5 Session management function

IceWall SSO uses session IDs to manage single sign on authentication.
This function uses session IDs to verify authentication (login status) and perform access control of users (access authorization/denial for each URL destination).
A brief description of session management is shown in the following diagram.



IceWall SSO session management

Note: The session ID can be set as a browser HTTP Cookie or as a URL-Cookie that includes the session ID in the URL of the IceWall server. The name of an HTTP cookie can be arbitrarily specified (the default name is "IW_INFO").

2.2.6 Information inheritance function

This function passes user login information and information stored on the Authentication DB to Backend Web Servers. The Backend Web Server application receives information such as HTTP headers and environment variables.

2.2.7 URL conversion function

This function converts URLs included in the content received from Backend Web Servers to IceWall-style URLs. Since all URLs included in content are converted to subdirectories of IceWall server, it appears as if the users are accessing a single web server. As a result, the existence of the Backend Web Server is transparent to users.

2.2.8 Keyword conversion function

This function converts strings in the content received from Backend Web Servers to the specified strings. This function is mainly used to convert any URLs that cannot be converted by the URL conversion function described above.

2.2.9 Reverse proxy function

This function allows the IceWall server to substitute for Backend Web Servers and receive access requests from users. The IceWall server analyzes a given URL and relays the information to Backend Web Servers.
A very secure system can be configured by fortifying the IceWall server, as all user access goes through it.

2.2.10 Access logging function

This function logs records of user access via the IceWall server. An administrator can use these records to analyze usage frequency and the operation history of each user.

2.2.11 Cross site scripting prevention function

This function disables malicious scripts sent from users. It can also handle already embedded scripts.

2.2.12 Automatic Form Authentication function

This function implements single sign on for form authentication built into applications running on Backend Web Servers.

2.2.13 Simple portal function

This function displays a simple portal page for each accessing user, based on various conditions.

2.2.14 SSL communication function (optional)

Besides the HTTP protocol, the SSL (https) protocol can also be used for communications between IceWall server and Backend Web Servers. This enables secure single sign-on to Backend Web Servers across the Internet as well.

2.2.15 Failover function (optional) 10.0

A function for increasing system availability. The failover function allows Forwarder to use an alternative backend server or Authentication Server. It also allows the simple portal function and the Windows integrated authentication function (discussed below) to use alternative Authentication Servers.

Also, a failover recovery function is available from version 10.0. As a result, when the master Authentication Module is recovered after a failover, the connection sequence before failover can be automatically restored.

2.2.16 Authentication Server load balancing function (optional) 10.0

Forwarder allots the Authentication Server's connections to distribute the load among Authentication Servers.

2.2.17 Client authentication function (optional)

This function verifies a user's identification by using client certificates. Installing client certificates in a browser enables more advanced user verification.

2.2.18 Windows integrated authentication function (support for Windows Kerberos) (optional)

This function enables single sign on with Microsoft Windows domain authentication. By simply logging on to the Windows domain, the user can log in to the IceWall SSO server seamlessly.

2.2.19 Agent function (optional)

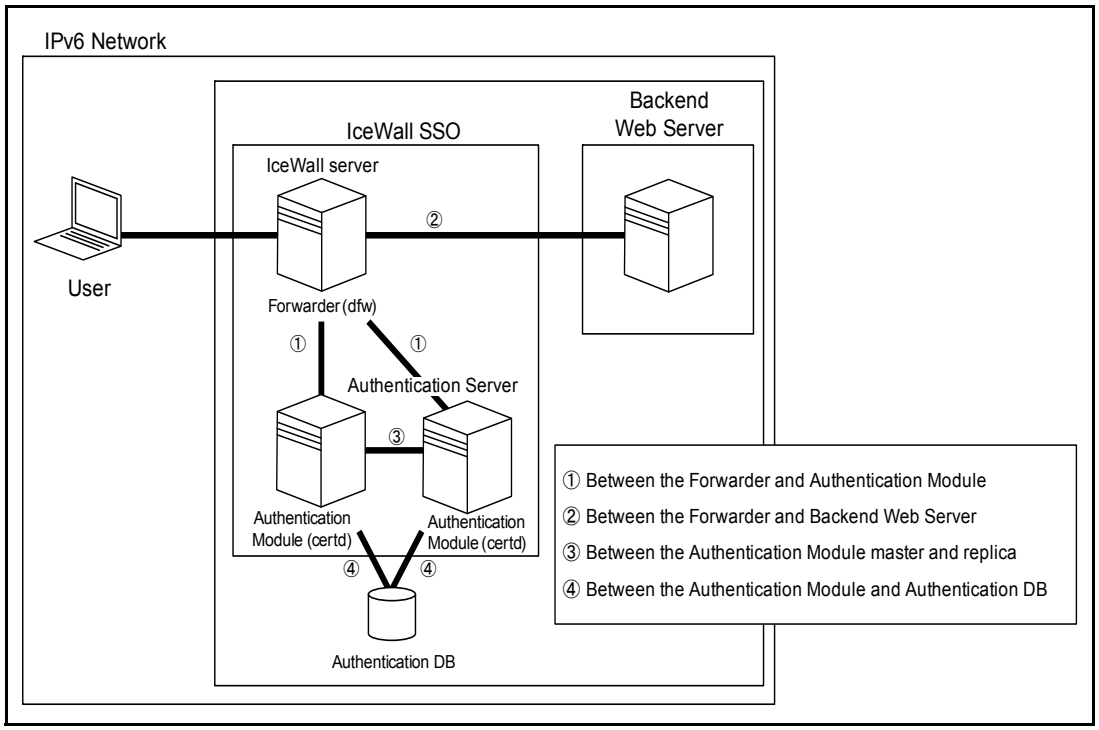
This function enables using agent single sign on in addition to traditional reverse proxy single sign on.

* This option is not included in the IceWall SSO installation media.

2.3 Basic transactions

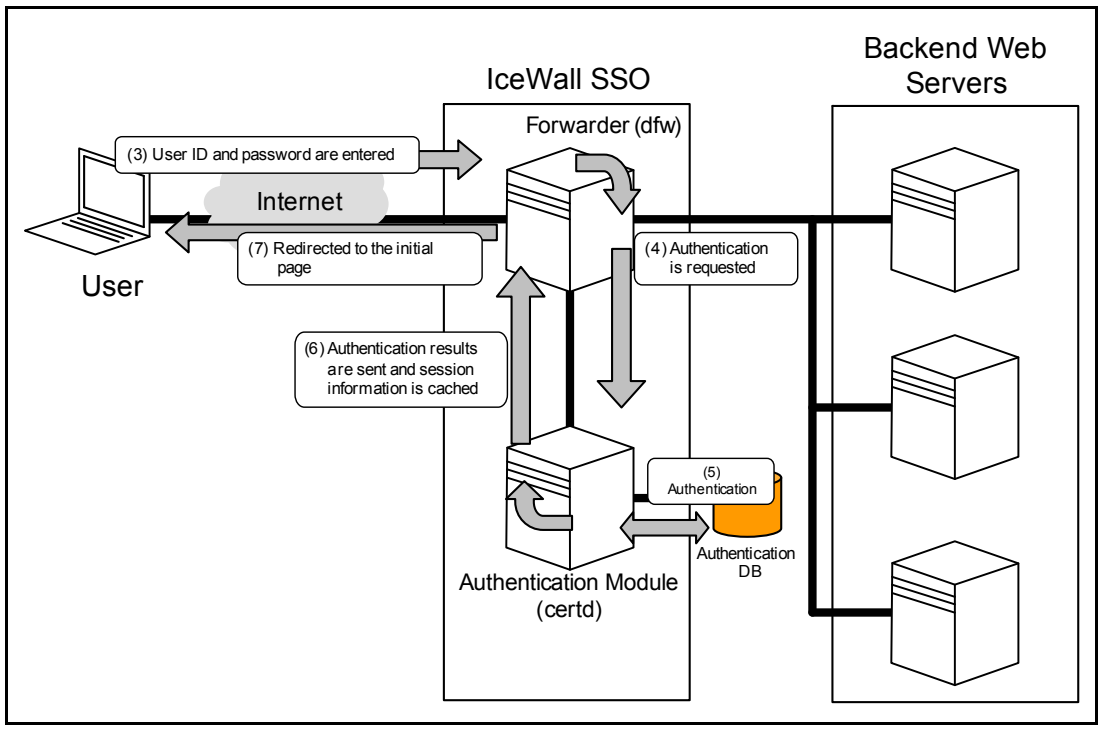
2.3.1 Login function

This function is used when users begin to access content on a Backend Web Server. It performs authentication with user IDs and passwords in connection with the Authentication DB.



(1) The user enters a URL for the initial page (the Backend Web Server page in this example).

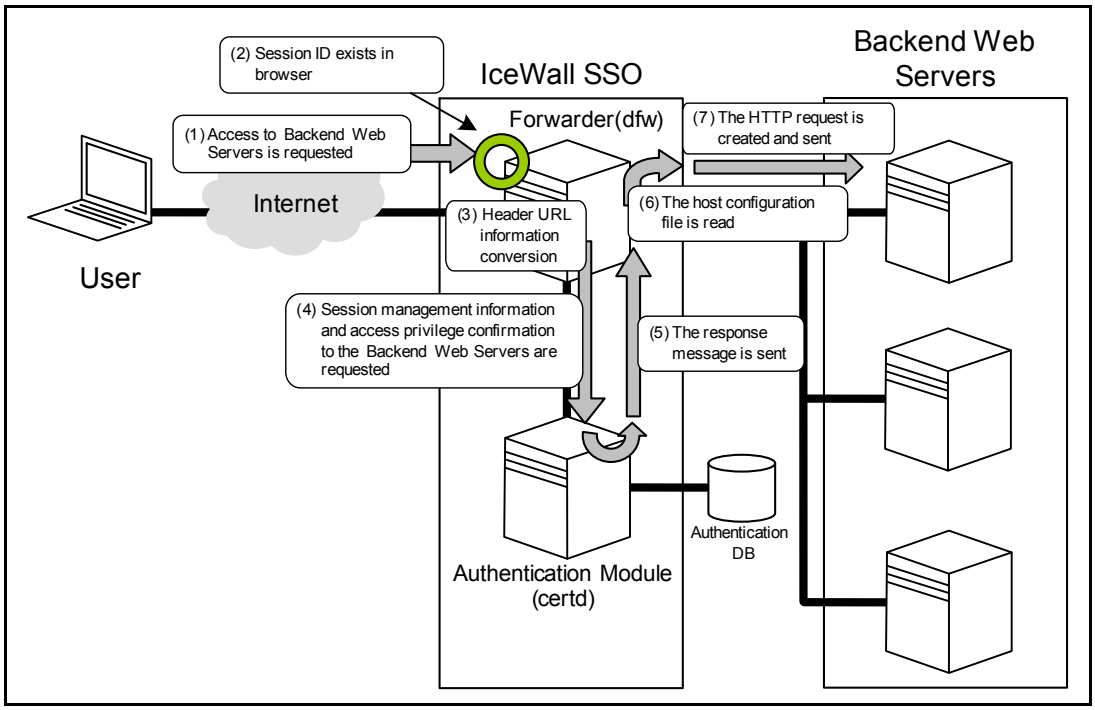
(2) In order to verify the login status of the user, Forwarder verifies the HTTP request header sent by the browser to see if a session ID exists. As no session ID exists in the header for initial access, Forwarder loads the login page and sends it to the user (if a session ID already exists, then the user is processed as a logged-in user and the process moves to the Backend Web Server page display).



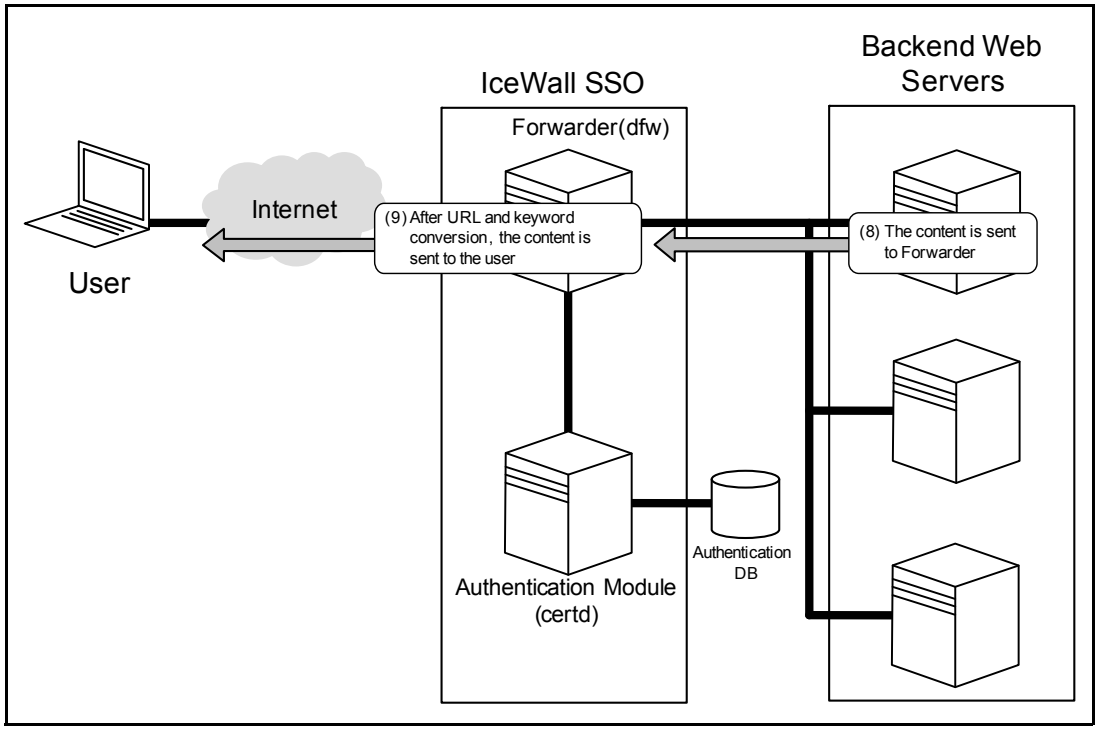
(3) The user enters a user ID and password in the login page, then a login request is sent to Forwarder.
(4) Forwarder uses the user ID and password received to request user authentication from the Authentication Module.
(5) The Authentication Module receives the request and performs authentication in connection with the Authentication DB.
(6) The Authentication Module then sends its results to Forwarder. If authentication is successful, the user session information will be created and cached
(7) If user authentication fails, Forwarder returns the login error page. If authentication succeeds, it returns a redirect response for the initial page and session information to the browser.
* A browser that has received a redirect response for the initial page will send an access request for the initial page to Forwarder.

2.3.2 Content display of Backend Web Servers

The IceWall server receives all access requests from users as a substitute for the Backend Web Servers and then relays the requests to the Backend Web Servers.



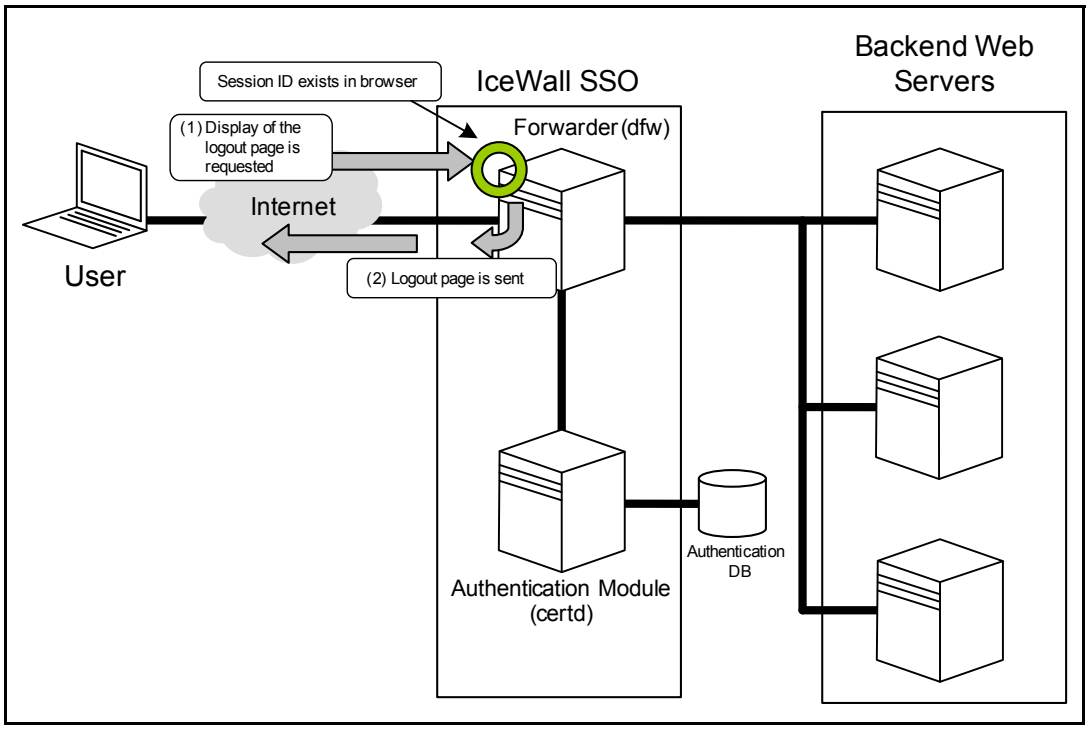
(1) The user issues an access request to a given Backend Web Server
(2) In order to verify the login status of the user, Forwarder verifies the HTTP request header sent by the browser to see if a session ID exists. If a session ID does not exist, then the user is requested to login again.
(3) If a session ID exists, Forwarder verifies the HTTP request header received from the user. If there are any URLs that are accessed via the IceWall SSO server, then these URLs are converted to URLs that are accessed without going through the IceWall SSO server.
(4) Forwarder requests the Authentication Module to provide session information and verify access privilege to the Backend Web Servers.
(5) The Authentication Module verifies the user's access privilege and sends a response message to Forwarder.
(6) If the session ID is invalid, then Forwarder sends a login page to the user; if there is an access privilege violation, it sends an access privilege error page to the user. If the session information and access privilege are valid, then it loads the host configuration file of the Backend Web Server that the user requested.
(7) Forwarder sends the request to a Backend Web Server.



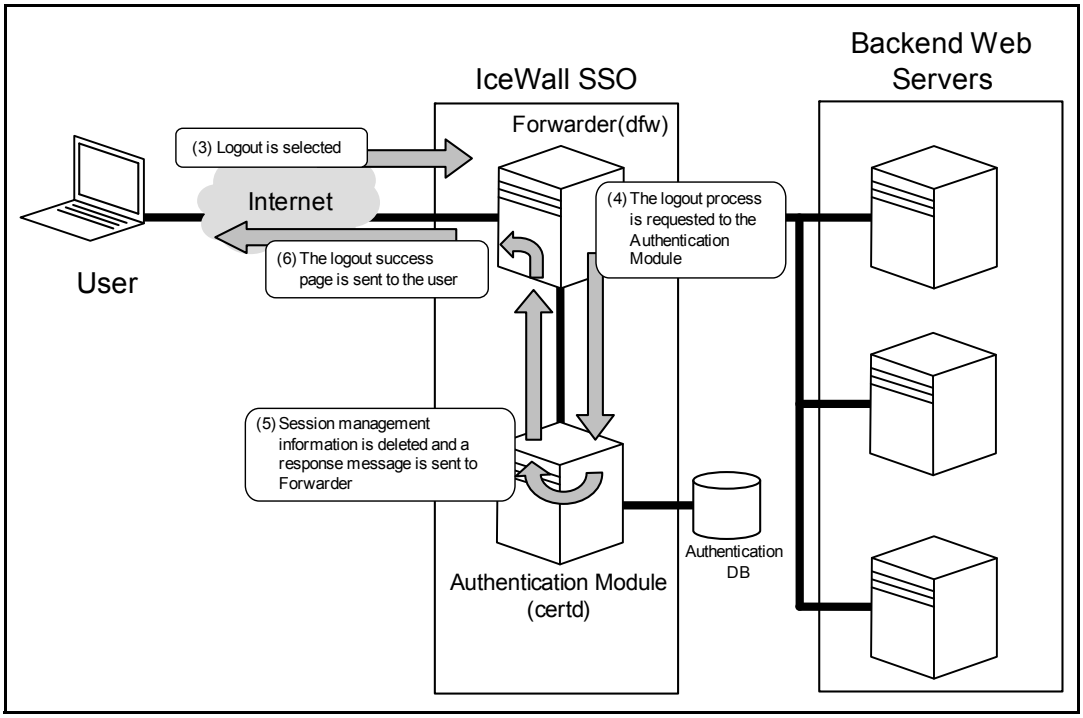
(8) The Backend Web Server sends the content to Forwarder.
(9) After Forwarder receives the content,
it converts URLs and keywords and sends the converted content to the user.

2.3.3 Logout function

This function terminates the user's access to Backend Web Servers. If the session management information has been deleted, a user cannot access a Backend Web Server unless the user logs in again.



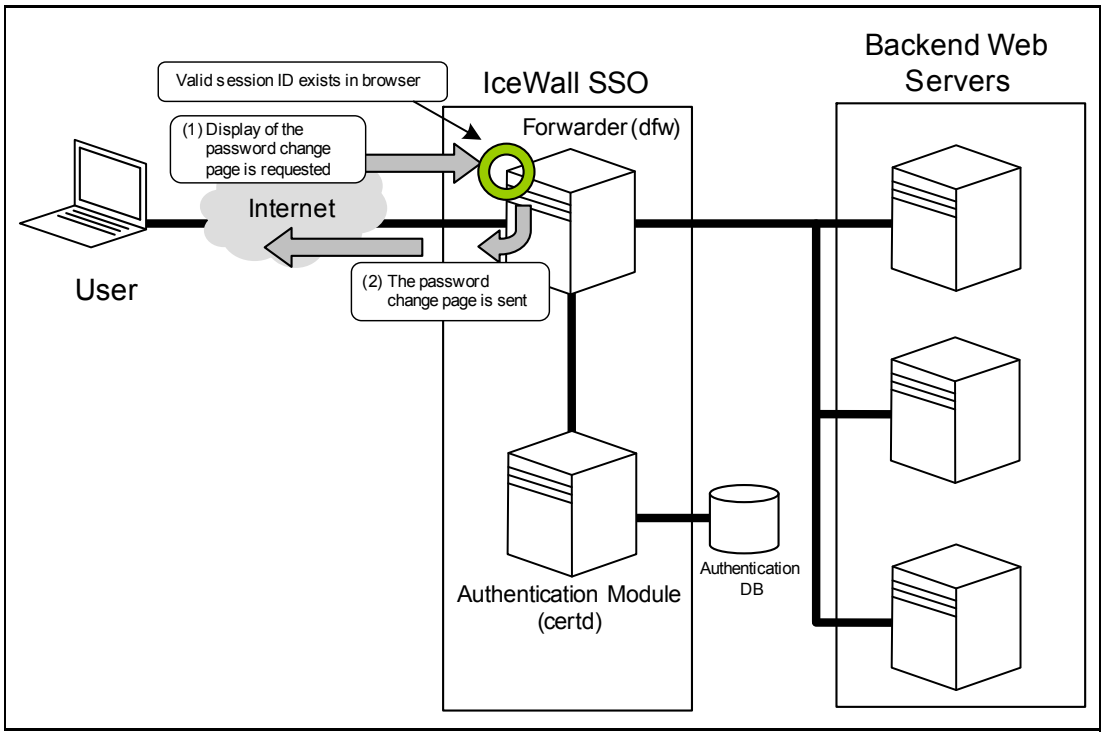
(1) A logged-in user accesses the IceWall SSO logout page.
(2) To verify the login status of the user, Forwarder verifies the presence of a session ID in the request from the browser, and then displays the logout page. (If a session ID does not exist, the login page is displayed.)



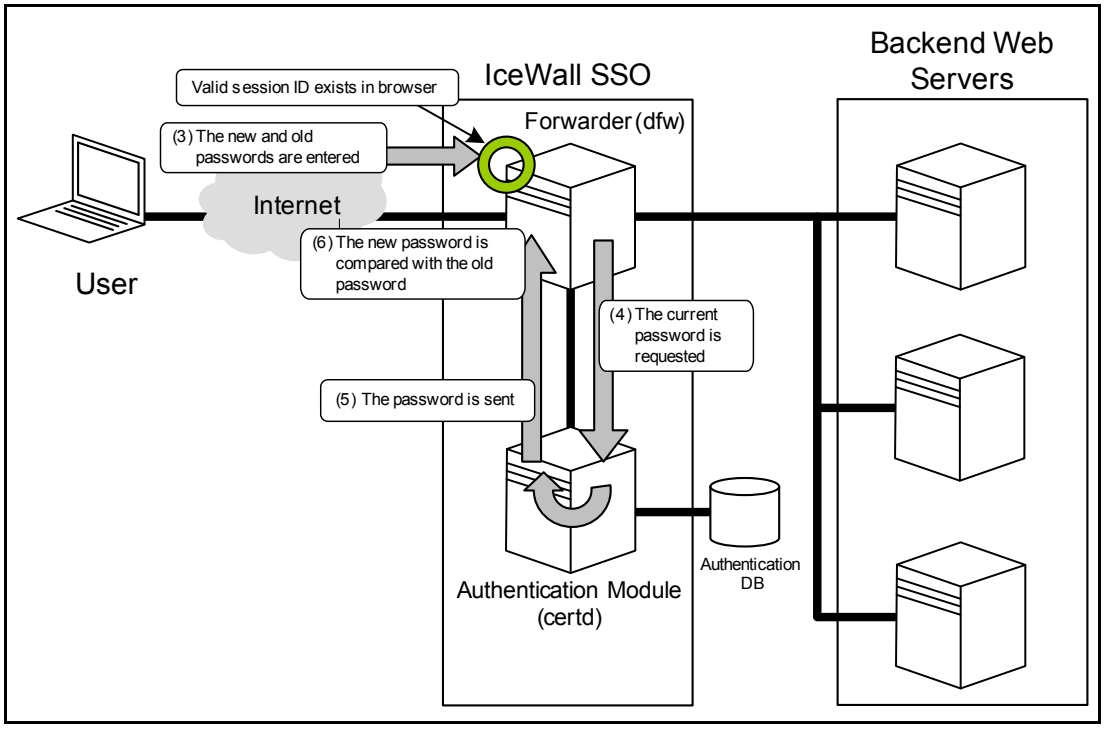
(3) The user selects to log out.
(4) Forwarder requests the Authentication Module to perform the logout process.
(5) The Authentication Module deletes the session information and sends a response message to Forwarder.
(6) Forwarder sends a logout success page to the user.

2.3.4 Password change function (10.0)

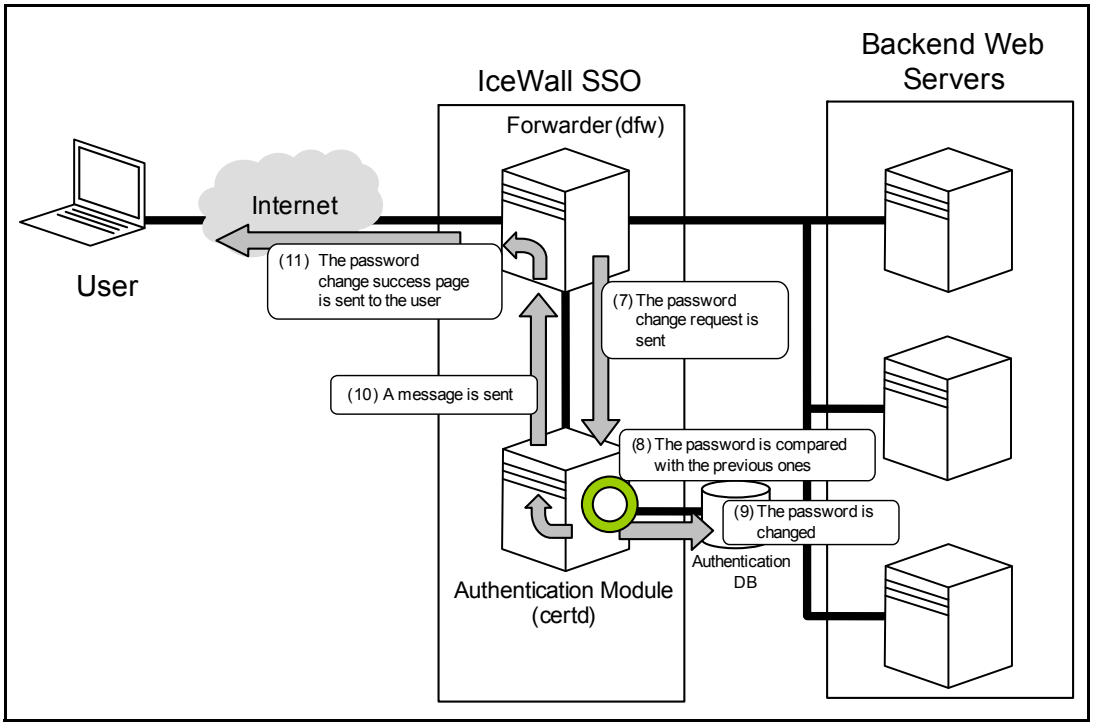
This function allows users to change their own passwords stored on the Authentication DB.



(1) A logged-in user accesses the password change page.
(2) To verify the login status of the user, Forwarder verifies the presence of a session ID in the request from the browser, and then it displays the password change page. (If a session ID does not exist, the login page is displayed.)



(3) The user enters both the new and old passwords on the password change page.
(4) Forwarder receives both the new and old passwords and requests the current password from the Authentication Module.
(5) The password is sent to Forwarder.
(6) The password received from the Authentication Module is compared with the old password to confirm whether it matches, and the new password entered is verified to determine whether it complies with the password policy.



(7) Forwarder checks whether the new password matches the confirmation password.
(8) If the new and confirmation passwords match, Forwarder sends the password change request to the Authentication Module.
(9) The Authentication Module checks whether the new password complies with the password policy. (If it does not comply, then an error page will be displayed.)
(10) If the password complies with the policy, the Authentication Module checks whether the new password matches any past password.
(11) If the password does not match any past password, the Authentication Module communicates with the Authentication DB and sets the new password. (If a match is found, a password policy error page will be displayed.)
(12) If the password change is successful, the Authentication Module sends a password change success message to Forwarder.
(13) Forwarder sends the password change success page to the user.

3 Basic Operations

Basic IceWall SSO operations include system startup and shutdown and configuration file editing.

3.1 System startup

(1) Starting Forwarder (dfw)

Forwarder is invoked by the web server as a CGI process. There is no need to start up Forwarder separately.

(2) Starting the Authentication Module (certd)

The Authentication Module is started by the iwadmin user with the following command:

```
$ /opt/icewall-sso/certd/bin/start-cert
```

When the system structure includes duplicate Authentication Modules (replication), the Primary Authentication Module is started first and then the Secondary Authentication Module, the replication target, is started.

3.2 System shutdown

(1) Shutting down Forwarder (dfw)

Forwarder is invoked by the web server as a CGI and shuts down after its execution. Therefore, there is no need to shut down Forwarder separately.

(2) Shutting down the Authentication Module (certd)

The Authentication Module is shut down by the iwadmin user with the following command:

* We recommend you check the Authentication Module status with the operating information output command (info-cert) before stopping the Authentication Module.

```
$ /opt/icewall-sso/certd/bin/end-cert
```

When the system structure includes duplicate Authentication Modules (replication), the Secondary Authentication Module is stopped first and then the Primary Authentication Module is stopped.

3.3 Editing configuration files

The configuration of each module can be edited to match the operating environment.

3.3.1 Forwarder configuration file

All of Forwarder configuration files are edited with a text editor. The following are the default storage locations for the configuration files:

/opt/icewall-sso/dfw/cgi-bin/dfw.conf (Forwarder configuration file)
/opt/icewall-sso/dfw/cgi-bin/html.conf (HTML configuration file)
/opt/icewall-sso/dfw/cgi-bin/sample.conf (host configuration file)
/opt/icewall-sso/dfw/cgi-bin/form.conf (form authentication configuration file)

When a configuration file is edited, the changes are reflected immediately after editing. If there are multiple Forwarders, it is necessary to edit the configuration files of all Forwarders.

3.3.2 Authentication Module configuration file

All configuration files for the Authentication Modules are edited with a text editor. The following are the default storage locations for the configuration files. To reflect the changes, the Authentication Module must be shut down and restarted.

<Configuration files for the ORACLE edition>
/opt/icewall-sso/certd/config/cert.conf (Authentication Module configuration file)
/opt/icewall-sso/certd/config/cert.acl (access control file)
/opt/icewall-sso/certd/config/cert.grp (group configuration file)
/opt/icewall-sso/certd/config/dbattr.conf (Authentication DB column information file)
/opt/icewall-sso/certd/config/request.acl (request control configuration file)
/opt/icewall-sso/certd/config/logdbattr.conf (log column information file)
/opt/icewall-sso/certd/config/pwdforbid.conf (forbidden password strings configuration file)

<Configuration files for the LDAP, OpenLDAP, NED, CSV, MySQL and MSAD editions>
/opt/icewall-sso/certd/config/cert.conf (Authentication Module configuration file)
/opt/icewall-sso/certd/config/cert.acl (access control file)
/opt/icewall-sso/certd/config/cert.grp (group configuration file)
/opt/icewall-sso/certd/config/dbattr.conf (Authentication DB column information file)
/opt/icewall-sso/certd/config/request.acl (request control configuration file)
/opt/icewall-sso/certd/config/pwdforbid.conf (forbidden password strings configuration file)

3.3.3 Reloading/synchronizing each Authentication Module configuration file 10.0

You can reload and synchronize the configuration files by executing the reload command (reload-cert) while the Authentication Module is running, which makes it possible to change the operation of the Authentication Module while it is running.

- Reload
  The content and configured values in each configuration file are loaded into the Authentication Module where the reload command was executed.

- Synchronization
  After the configuration files are reloaded by executing the reload command, synchronization processing is performed with the replication target.

The configuration files that can be reloaded and synchronized with the reload command (reload-cert) are listed below.

cert.acl (access control file)
cert.grp (group configuration file)
request.acl (request control configuration file)
pwdforbid.conf (forbidden password strings configuration file)

For the following configuration files, only certain parts of parameters are reloaded and synchronized.

cert.conf (Authentication Module configuration file)

Reloadable parameters	ALEVEL
	ELEVEL
	LOGPERF
	LOGINFO
	ACCTHREAD
	TRANSID
	MAXLOGINUSER
	FAILBACK
	HTTPECHOHEADER
	PWDHISCHK
	PWDHISCNT
	PWDMINLEN
	PWDMAXLEN
	PWDALPHANUM
	PWDEXPIRE
	PWDSAMEPASS
Synchronizable parameters	TRANSID
	MAXLOGINUSER
	HTTPECHOHEADER
	PWDHISCHK
	PWDHISCNT
	PWDMINLEN
	PWDMAXLEN
	PWDALPHANUM
	PWDEXPIRE
	PWDSAMEPASS

The Authentication Module reloads and synchronizes its configuration by the command below that the iwadmin user executes.

```
$ /opt/icewall-sso/certd/bin/reload-cert
```

When the reload command is executed, the authentication/authorization processes are temporarily held. For this reason, if this command is used when there are many requests, a failover may occur at Forwarder or connections to the Authentication Module may fail. Using this command repeatedly within a short period of time may cause the processing performance to drop.

Note the following items when using this function:

- Use this function when there are only a few requests (recommended to use when there are no requests at all).
- Allow at least one minute between uses of this function.
- Comments in the configuration files may be deleted by performing synchronization.

* The user information for logged-in users is not deleted even if the Authentication Module is reloaded and synchronized.

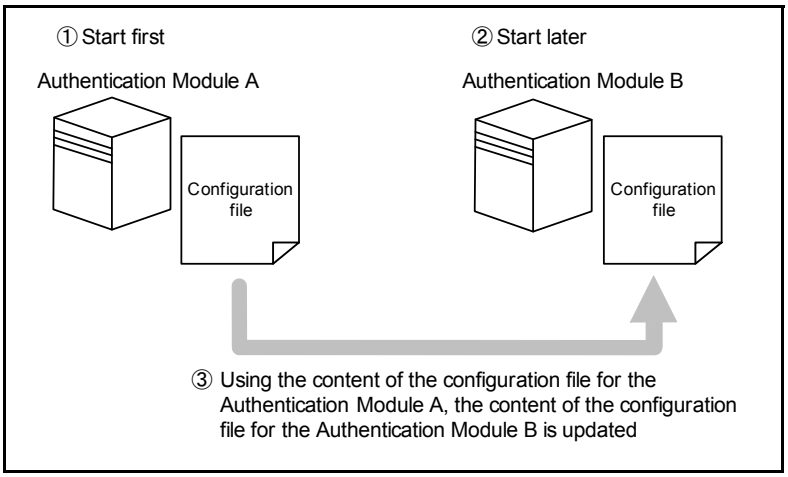
3.3.4 Updating the Authentication Module configuration file (10.0)

The Authentication Module's configuration file is updated in the situations below.

- After the Authentication Module is started, if the DBUID parameter and the DBPWD parameter in the Authentication Module configuration file are ClearText, the configuration file is updated to encrypt and save those parameters.

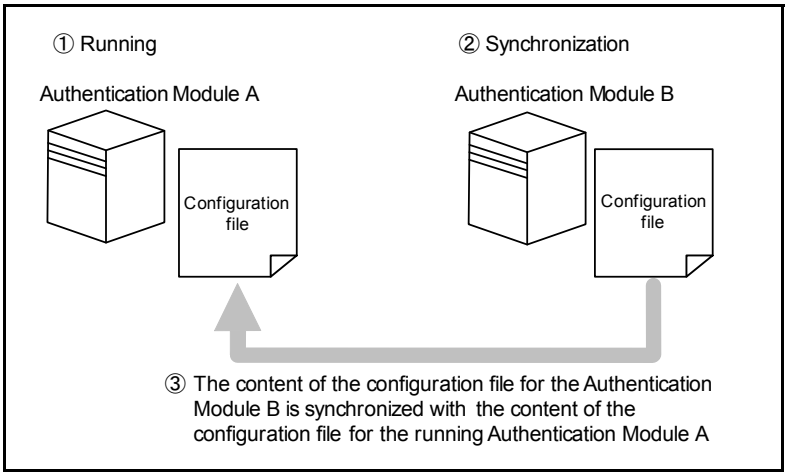
- When the connected Authentication DB changes because of a failover, the configuration file contents are updated.

- When the Authentication Modules start with a redundant configuration applied, the Authentication Module that was started later, regardless of Primary or Secondary, receives the configuration files and configuration parameter settings to be downloaded from the running Authentication Module, and it updates its own configuration file.



Configuration file download process at the time of start up

- When the reload command (reload-cert) is executed, the configuration file is updated in a configuration where the Authentication Modules are configured redundantly. The synchronization target Authentication Module updates its own configuration file when it receives configuration files and configuration parameters that are subject to synchronization.



- When the Authentication DB connected with the Authentication Module is switched by Authentication DB failover:
  The Authentication DB, which the Authentication Module connects to, fails over on some Authentication DB failure or something similar to that. The connection information to the Authentication DB will be updated on such failover.

3.3.5 Notes on configuration file descriptions

The configuration files used by the modules are defined as text files.
For more details, see "1.1 Configuration file format" in the "IceWall SSO Reference Manual."

3.4 Authentication Module command process exit status for the Authentication Module 10.0

From version 10.0, if there is an error when an Authentication Module command (start-cert, end-cert, info-cert, reload-cert, logout-cert, cdump-cert) is executed, the output message is now output to standard output. An error exit status (other than 0) is now returned too.

(1) Output messages and exit status codes
The output messages and exit status codes when an Authentication Module command (start-cert, end-cert, info-cert, reload-cert, logout-cert, cdump-cert) is executed are listed below.

Output message	Exit status	Description
certification module terminated.	0	The stop command (end-cert) was successfully executed.

Output message	Exit status	Description
config file reloaded.	0	The reload command (reload-cert) was successfully executed.
cert information output.	0	The operating information output command (info-cert) was successfully executed.
config file dump.	0	The configuration file information output command (cdump-cert) was successfully executed.
config file cannot dump.	5	The configuration file information output command (cdump-cert) failed.
all user logout success.	0	The log out all users command (logout-cert) was successfully executed.
all user logout failed. errmsg=REQPERMERR	3	You do not possess log out all users command (logout-cert) execution privileges. Execute the command on the server where the Authentication Module is running.
all user logout failed. errmsg=LOGOUTERR	5	The log out all users command (logout-cert) failed to log out one or more users.
all user logout failed. errmsg=DBBUSY	5	The log out all users command (logout-cert) was stopped because of a DB busy error.
all user logout failed. errmsg=SYSTEMERR	5	An unexpected error occurred while executing the log out all users command (logout-cert). The log out all users operation was canceled.
ERROR: Unknown parameter.	2	The arguments are wrong.
ERROR: cannot parse response.	4	Could not accurately evaluate the Authentication Module command response.
ERROR: deny request message.	3	You do not possess request execution privileges. Execute the command on the server where the Authentication Module is running.
ERROR:send request message.	4	Failed to send the request for the Authentication Module command.
ERROR: internal server error.	5	An internal server error has occurred. Could not receive the response.

(2) Suppressing messages

The output message can be suppressed by specifying the "--silent" option in the Authentication Module command.
The script below shows how to suppress messages with the reload command (reload-cert).

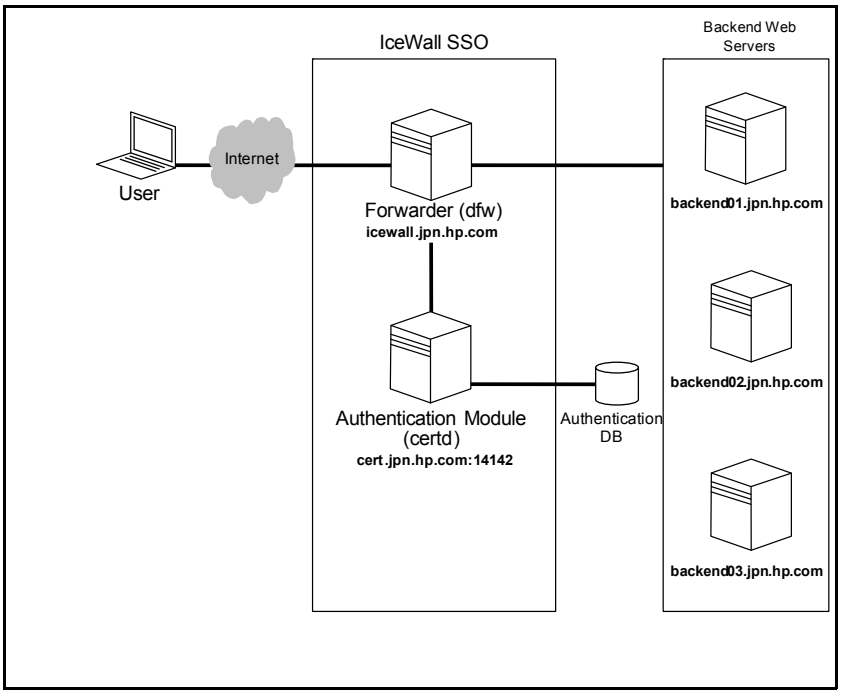
```
#!/bin/sh

export IW_HOME=/opt/icewall-sso
export SHLIB_PATH=$IW_HOME/lib/certd
export LD_LIBRARY_PATH=$SHLIB_PATH

$IW_HOME/certd/bin/certd -F -R --silent -P 14142
```

3.5 System configuration example

Below is a system configuration example that will be referenced in the descriptions below.



Extract the parameters required to implement IceWall SSO when using a server that runs a web application from an existing system as a Backend Web Server in IceWall SSO. An example is described below:

Parameter name	Description
Server running Forwarder (IceWall server)	icewall.jpn.hp.com

Parameter name	Description
User ID	user01, user02, user03, mobile01, mobile02, mobile03
Backend Web Server	backend01.jpn.hp.com Alias name : BK1 backend02.jpn.hp.com Alias name : BK2 backend03.jpn.hp.com Alias name : BK3
Authentication DB server For a server on which ORACLE, LDAP, OpenLDAP, NED, MSAD, or MySQL is running	ORACLE : SID=ORCL (cert.jpn.hp.com) LDAP : cert.jpn.hp.com:389 OpenLDAP : cert.jpn.hp.com:389 NED : cert.jpn.hp.com:389 MSAD : msad.jpn.hp.com:389 MySQL : icewalldb:cert.jpn.hp.com:3306:MySQL ODBC 5.1.6 Driver
Authentication table (ORACLE)	Table name : icewalltest User name : scott Password : tiger
Authentication directory (LDAP, OpenLDAP)	Directory name : o=hp.com User name : uid=admin,ou=Administrator,ou=TopologyManagement,o=NetscapeRoot Password : passwd
Authentication directory (NED)	Directory name : o=hp.com User name : cn=admin Password : passwd
Authentication directory (MSAD)	Directory name : CN=Users,DC=hp,DC=com User name : CN=Administrator,CN=Users,DC=hp,DC=com Password : passwd
Authentication table (MySQL)	Table name : icewalltest User name : root Password : new_password
Authentication file (CSV)	File name : /opt/icewall-sso/certd/config/sample.csv
A server on which the Authentication Module is running (Authentication Server)	cert.jpn.hp.com:14142
Groups to which individual users belong and their access privilege to web servers	user01, group01 : backend03.jpn.hp.com user02, group02 : backend03.jpn.hp.com, backend01.jpn.hp.com user03, group03 : all mobile01, group01 : backend03.jpn.hp.com mobile02, group02 : backend03.jpn.hp.com, backend01.jpn.hp.com mobile03, group03 : all
Authentication method	Authentication via user ID and password
Password policy	Minimum of five characters, maximum of ten characters, an expiration term of 72 days; user ID and password may be identical and must contain both alphabetic and numeric characters

Parameter name	Description
Keyword conversion	The content on the web server at backend03.jpn.hp.com is displayed with the keyword “aaaa” converted into “bbbb.”

3.6 Editing the Forwarder configuration file

The Forwarder configuration file is edited according to the above configuration example.

3.6.1 Editing the Forwarder configuration file (dfw.conf)

Edit the Forwarder configuration file (/opt/icewall-sso/dfw/cgi-bin/dfw.conf) of the IceWall SSO server by following the configuration details shown in “3.5 System configuration example.” The configurations of the Authentication Module and the Backend Web Server to which IceWall SSO is connected are all performed via the configuration file of Forwarder.

(1) Set the Authentication Module server and port number to “cert.jpn.hp.com:14142.”

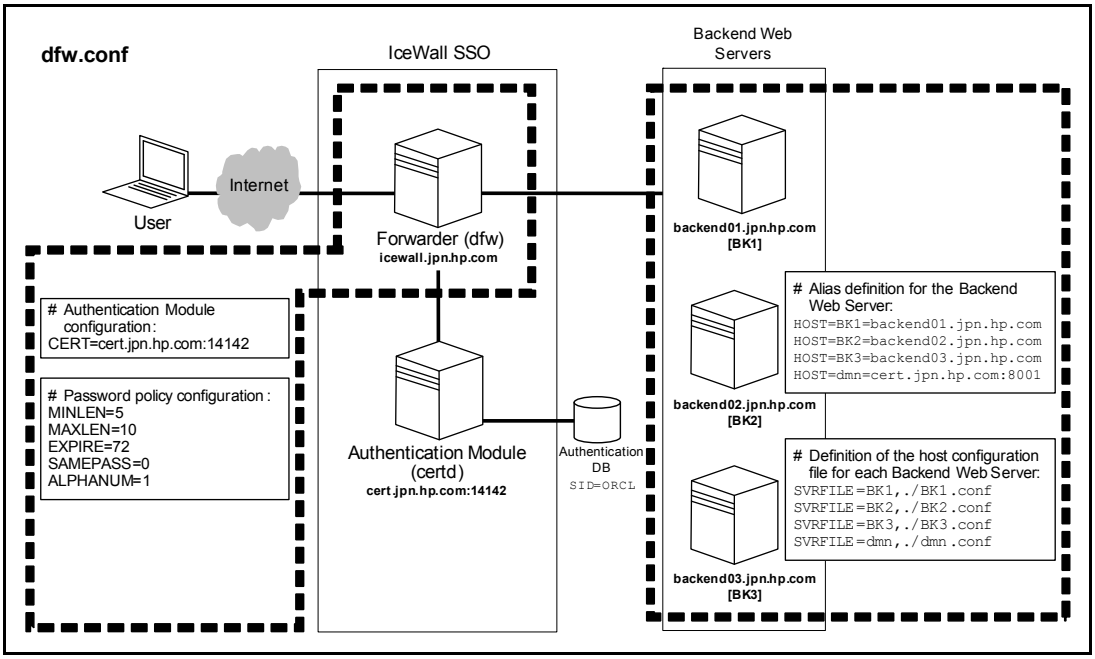
Parameter name	Description
CERT	CERT=cert.jpn.hp.com:14142

(2) Set the aliases for the Backend Web Servers to “BK1” for backend01.jpn.hp.com, “BK2” for backend02.jpn.hp.com, and “BK3” for backend03.jpn.hp.com.

Parameter name	Description
HOST	HOST=BK1=backend01.jpn.hp.com HOST=BK2=backend02.jpn.hp.com HOST=BK3=backend03.jpn.hp.com

(3) Set the host configuration files of the Backend Web Servers as “BK1.conf” for backend01.jpn.hp.com, “BK2.conf” for backend02.jpn.hp.com, “BK3.conf” for backend03.jpn.hp.com, and “dmn.conf” for cert.jpn.hp.com:8001.
The host configuration files are created in the same directory as the Forwarder configuration files.

Parameter name	Description
SVRFILE	SVRFILE=BK1,./BK1.conf SVRFILE=BK2,./BK2.conf SVRFILE=BK3,./BK3.conf SVRFILE=dmn,./dmn.conf



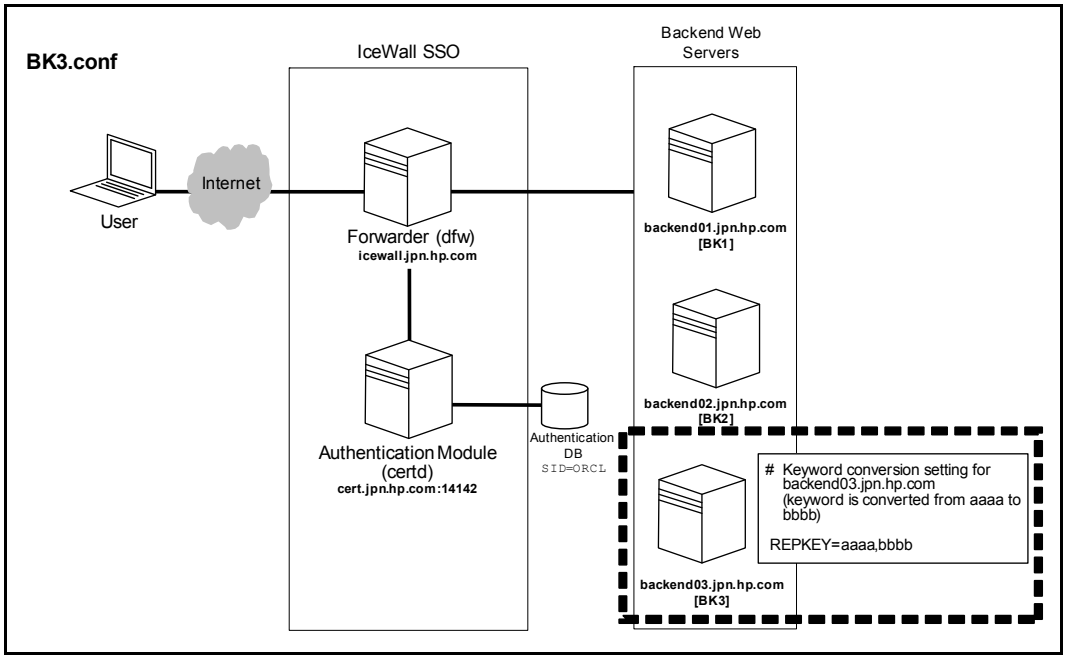
3.6.2 Editing the host configuration files

Edit the host configuration file for each Backend Web Server. The host configuration file (BK3.conf) for backend03.jpn.hp.com is used in the example here to describe how to configure keyword conversion settings. This configuration is performed based on the host configuration sample file: (/opt/icewall-sso/dfw/cgi-bin/sample.conf).

<BK3.conf>

(1) The configuration is performed in such a way that a page is displayed only after the keyword “aaaa” is replaced by “bbbb.”

Parameter name	Description
REPKEY	REPKEY=aaaa,bbbb



3.6.3 Editing the HTML configuration file (html.conf)

Edit the HTML configuration file to set the web page that is output to browsers by Forwarder. For details on the configuration method, see the “IceWall SSO Reference Manual.”
(When the HTML template is configured on an external web server, the HTTP headers, which are sent by the browser, are transferred to the server without conversion.)

3.7 Editing the Authentication Module configuration file

Edit the Authentication Module configuration file according to the configuration example above. For details on individual parameters, see “3.1.5 Authentication DB related parameters” in the “IceWall SSO Reference Manual.”

3.7.1 Editing the Authentication Module configuration file (cert.conf)

The Authentication Module configuration file settings may be slightly different depending on the Authentication DB used.

<Common settings>

(1) Set the password policy for password changes to “a minimum of five characters, a maximum of ten characters, an expiration term of 72 days; user ID and password may be identical and must contain both alphabetic and numeric characters.”

Parameter name	Description
PWDMINLEN	PWDMINLEN=5
PWDMAXLEN	PWDMAXLEN=10
PWDALPHANUM	PWDALPHANUM=1
PWDEXPIRE	PWDEXPIRE=72
PWDSAMEPASS	PWDSAMEPASS=0

(2) Set the number of the port connecting to Forwarder to “14142.”

Parameter name	Description
PORT	PORT=14142

(3) Configure a value for the maximum number of users that can use IceWall SSO simultaneously. In the example below, the maximum number of simultaneously logged in users is set to “100.”

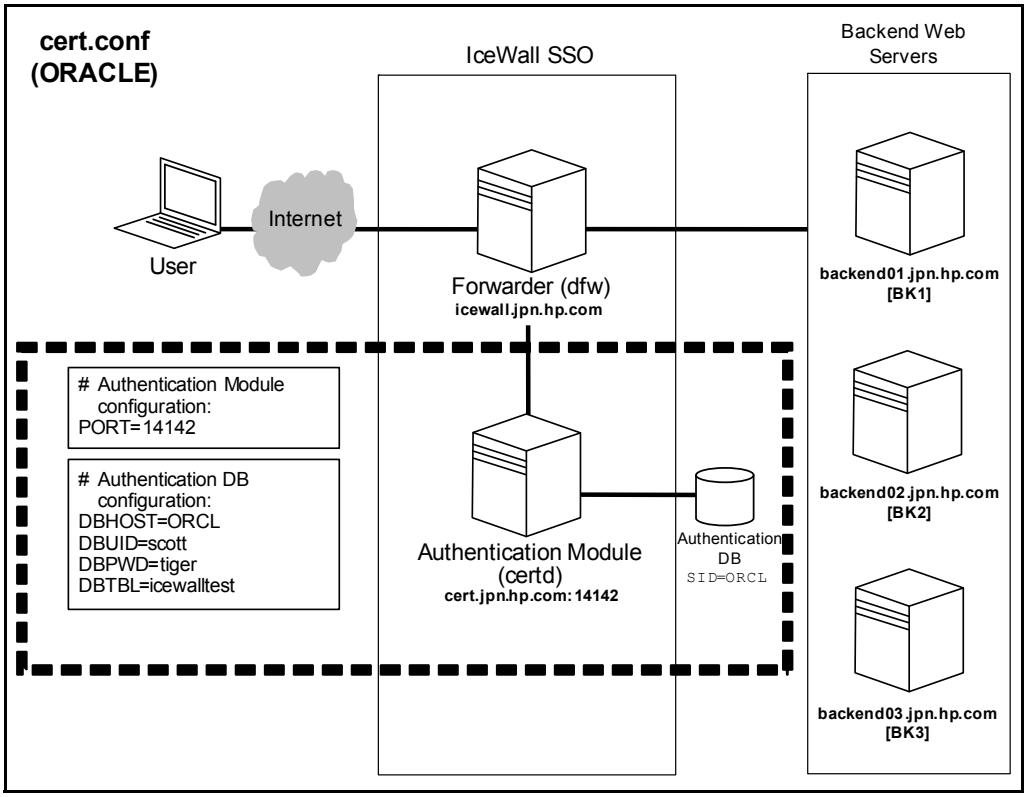
Parameter name	Description
CACHE	CACHE=100

(4) Configure the Authentication DB.

<ORACLE>

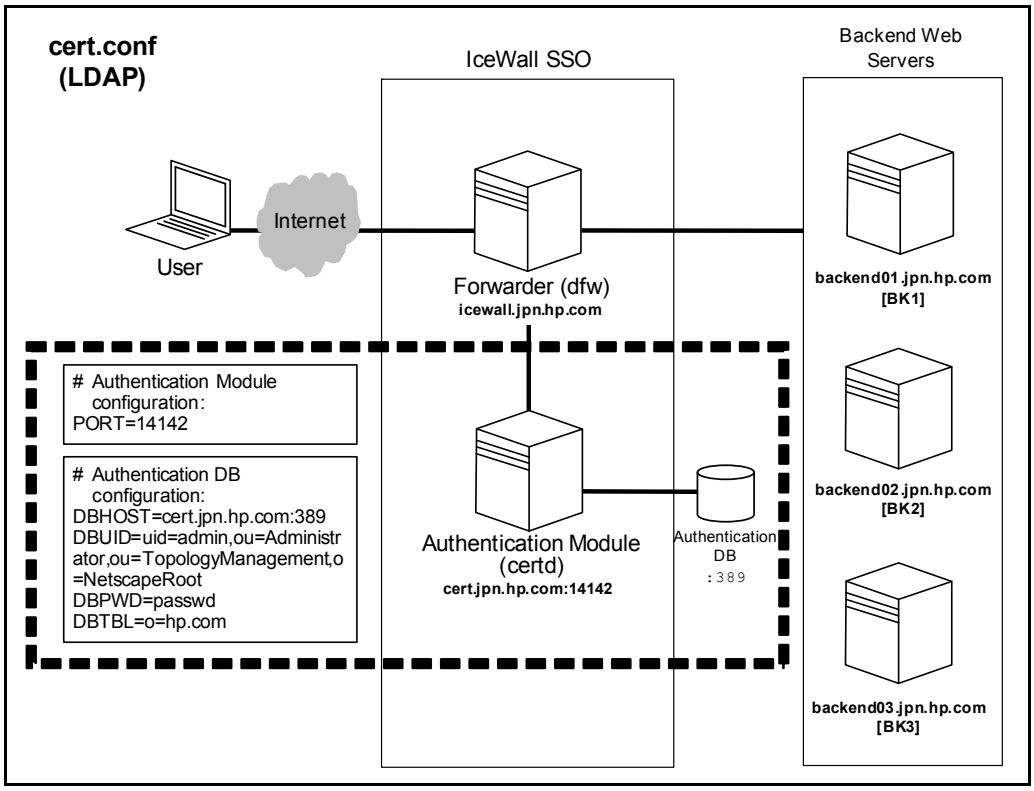
When performing user authentication, configure the database so that the Authentication Module connects to the database with the “ORCL” SID, logs in with the user ID “scott” and password “tiger,” and references the table “icewalltest.” (For details on the icewalltest table, see the “IceWall SSO Sample Setup Guide.”)

Parameter name	Description
DBHOST	DBHOST=ORCL
DBUID	DBUID=scott
DBPWD	DBPWD=tiger
DBTBL	DBTBL=icewalltest



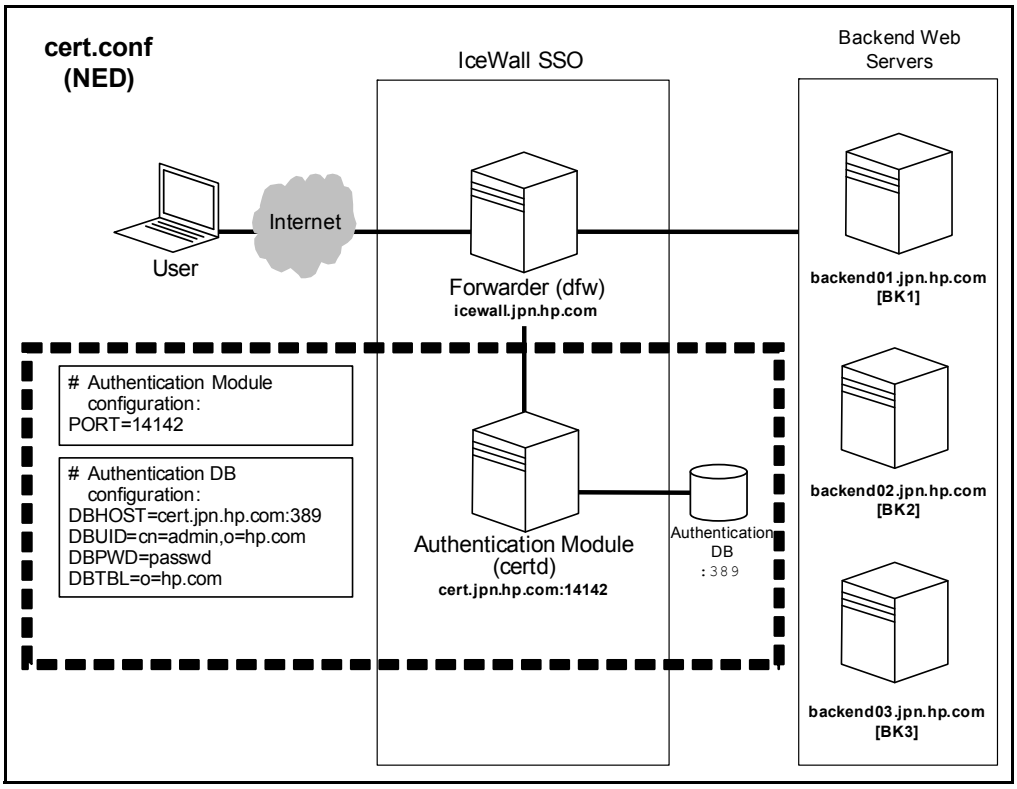
<LDAP, OpenLDAP>
When performing user authentication, configure the database so that the Authentication Module connects to the LDAP, OpenLDAP server with the host name “cert.jpn.hp.com” on port 389, logs in with the user ID “uid=admin, ou=Administrator,ou=TopologyManagement,o=NetscapeRoot” and password “passwd,” and references the tree “o=hp.com.”

Parameter name	Description
DBHOST	DBHOST=cert.jpn.hp.com:389
DBUID	DBUID=uid=admin,ou=Administrator,ou=TopologyManagement,o=NetscapeRoot
DBPWD	DBPWD=passwd
DBTBL	DBTBL=o=hp.com



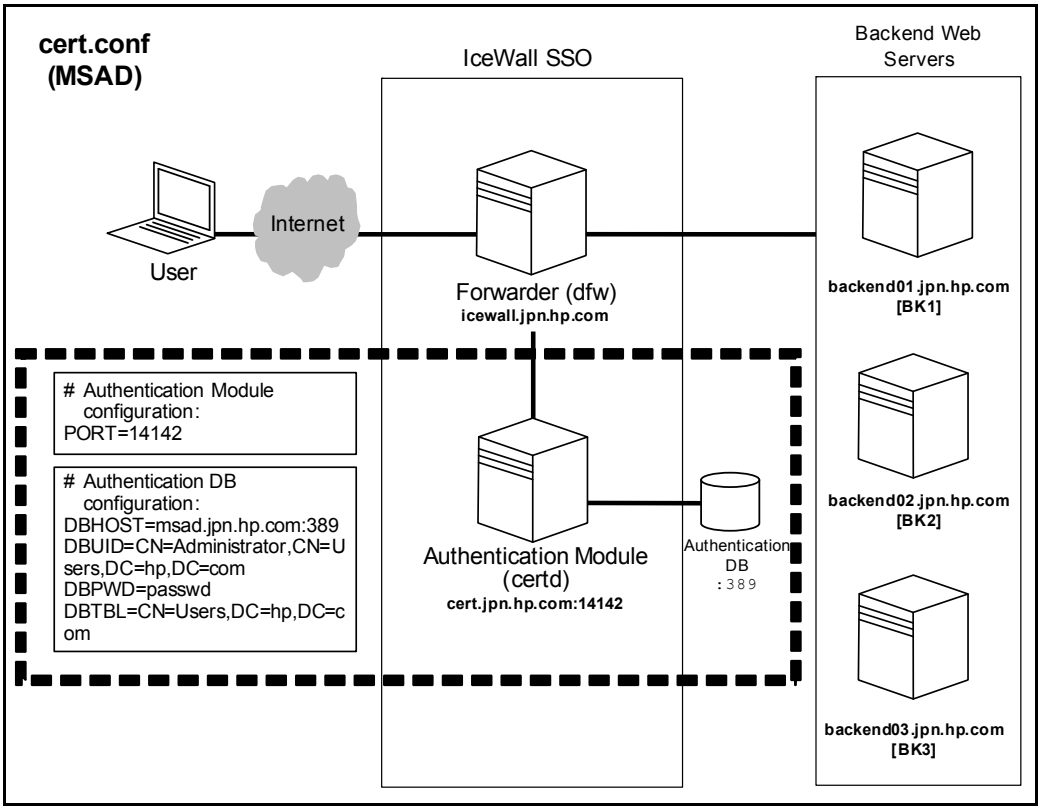
<NED>
When performing user authentication, configure the database so that the Authentication Module connects to the eDirectory of host name “cert.jpn.hp.com” on port 389, logs in with the user ID “cn=admin,o=hp.com” and password “passwd,” and references the tree “o=hp.com.”

Parameter name	Description
DBHOST	DBHOST=cert.jpn.hp.com:389
DBUID	DBUID=cn=admin,o=hp.com
DBPWD	DBPWD=passwd
DBTBL	DBTBL=o=hp.com



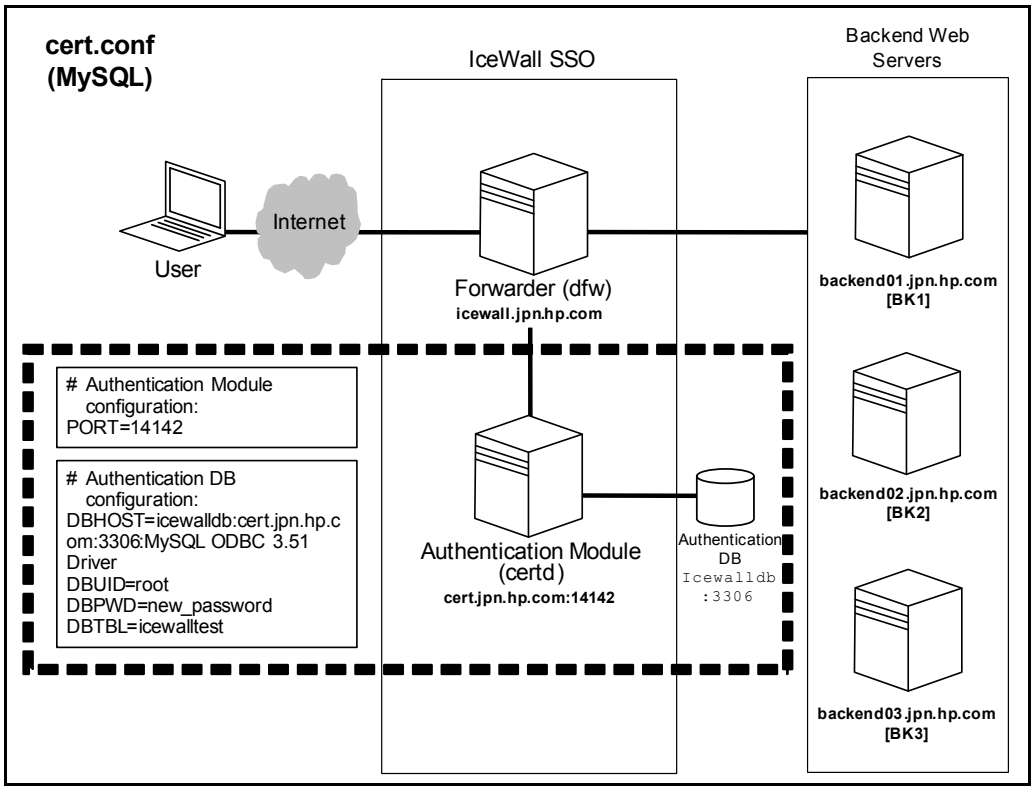
<MSAD>
When performing user authentication, configure the database so that the Authentication Module connects to the Active Directory of host name "msad.jpn.hp.com" on port 389, logs in with the user ID "CN=Administrator, CN=Users,DC=hp,DC=com" and password "passwd," and references the tree "CN=Users,DC=hp,DC=com."

Parameter name	Description
DBHOST	DBHOST=msad.jpn.hp.com:389
DBUID	DBUID=CN=Administrator,CN=Users,DC=hp,DC=com
DBPWD	DBPWD=passwd
DBTBL	DBTBL=CN=Users,DC=hp,DC=com



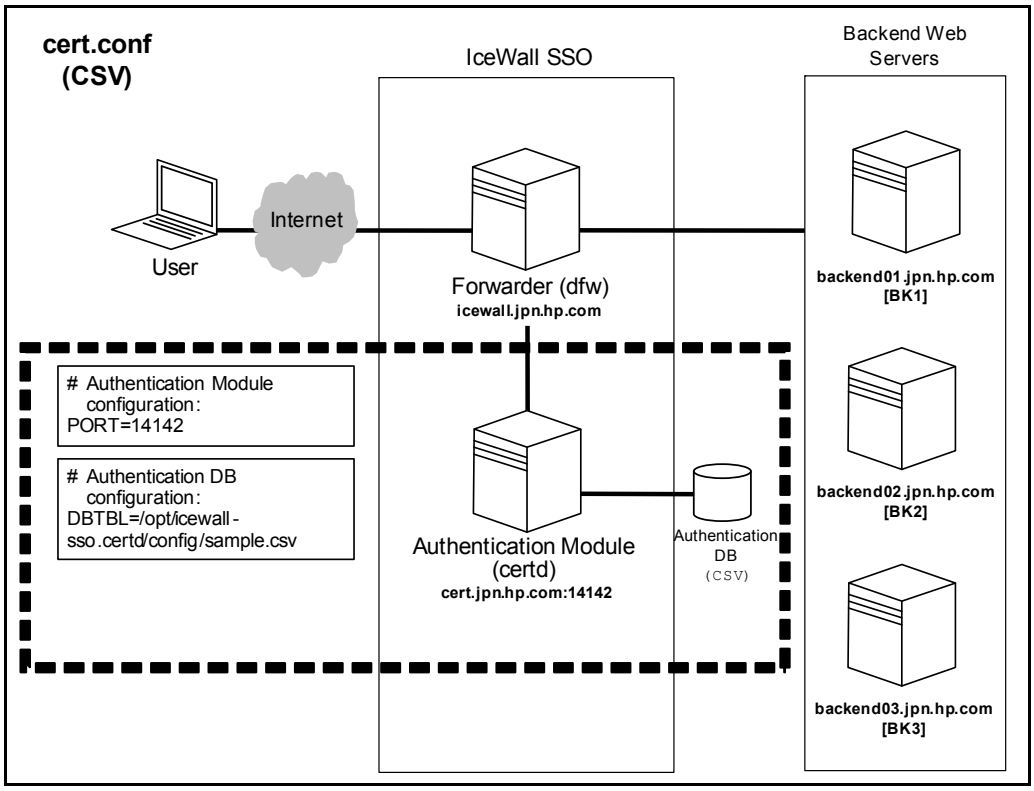
<MySQL>
When performing user authentication, configure the database so that the Authentication Module connects to the database “icewalldb” on host name “cert.jpn.hp.com” on port 3306, logs in with the user ID “root” and password “new_password,” and references the table “icewalltest.” (For details on the icewalltest table, refer to the “IceWall SSO Sample Setup Guide.”)

Parameter name	Description
DBHOST	DBHOST=icewalldb:cert.jpn.hp.com:3306:MySQL ODBC 3.51 Driver
DBUID	DBUID=root
DBPWD	DBPWD=new_password
DBTBL	DBTBL=icewalltest



<CSV>
When performing user authentication, configure the database so that the Authentication Module references the local file "/opt/icewall-sso/certd/config/sample.csv."

Parameter name	Description
DBHOST	This does not need to be configured.
DBUID	This does not need to be configured.
DBPWD	This does not need to be configured.
DBTBL	DBTBL=/opt/icewall-sso/certd/config/sample.csv



3.7.2 Editing the group configuration file (cert.grp)

Configuration of the group configuration file varies depending on the Authentication DB used.

<Groups created>

- Configure user01 and mobile01 as comprising members of "group01," user02 and mobile02 as members of "group02," and user03 and mobile03 as members of "group03."

<ORACLE, MySQL, CSV>

Description
group01,USERID=user01|USERID=mobile01 group02,USERID=user02|USERID=mobile02 group03,USERID=user03|USERID=mobile03

<LDAP, OpenLDAP>

Description
group01,uid=user01|uid=mobile01 group02,uid=user02|uid=mobile02 group03,uid=user03|uid=mobile03

<NED, MSAD>

Description
group01,CN=user01|CN=mobile01 group02,CN=user02|CN=mobile02 group03,CN=user03|CN=mobile03

3.7.3 Editing the access control file (cert.acl)

Edit the Authentication Module access control file (/opt/icewall-sso/certd/config/cert.acl).

Configure the file to grant access privileges for BK1 to group02 and group03, access privilege for BK2 to group03, and access privilege for BK3 to all groups.

Description
http://backend01.jpn.hp.com/=group02|group03 http://backend02.jpn.hp.com/=group03 http://backend03.jpn.hp.com/=group01|group02|group03

3.7.4 Editing the Authentication DB column information file (dbattr.conf)

Edit the Authentication DB column information file to configure the relationship between the parameter names and actual column names.

3.7.5 Editing the log column information file (logdbattr.conf)

Note: Only when Oracle is used as the Authentication DB.
Edit the log column configuration file to configure the relationship between the parameter names and actual column names.

3.7.6 Editing the forbidden password configuration file (pwdforbid.conf)

Edit the forbidden password configuration file to configure the strings that are forbidden for use as passwords.

3.7.7 Restarting the Authentication Module

The Authentication Module is a daemon process. If changes are made to the Authentication Module's configuration files, they will not be effective until the module is restarted.

3.8 Log out all users function 10.0

From version 10.0, an Authentication Module command (logout-cert) to forcibly log out all logged-in users has been added.

To forcibly log out all logged-in users with the iwadmin user.

```
$ /opt/icewall-sso/certd/bin/logout-cert
```

When the above log out all users command (logout-cert) is executed, the log below is output in the access log or the error log.

Output message	Description	Remarks
User Logout. (All User Logout) UserID=xxxxxxx nnn:nn:nn	Users were logged out by the log out all users operation.	The user IDs that were logged out and the time they were logged in are output in UserID.
All User Logout Error. UserID=xxxxxxx	Failed in the logout operation for the given users during the logout operation for all users.	The user IDs that could not be logged out are output in UserID.
Database Update Error. (All User Logout) UserID=xxxxxxx	Failed in the DB update operation for the given users during the logout operation for all users.	The user IDs that could not be updated in the DB are output in UserID.

3.9 Handling Authentication Module maintenance 10.0

When you must stop the Authentication Module for maintenance or other reasons, you are now able to block off the Authentication Module in stages. The blocking stages are pre-blocking (stop accepting new login requests) and main blocking (stop access to the Authentication Module).

3.9.1 Pre-blocking

Pre-blocking stops accepting new login requests. The already logged-in users can continue to process their tasks. The login limit error page is displayed when a new login request is handled in this stopped state.

(1) Set the value of the login limit configuration parameter (MAXLOGINUSER) in the Authentication Module configuration file (cert.conf) to 0.

Authentication Module configuration file (cert.conf)

```
MAXLOGINUSER=0
```

(2) Execute the reload command (reload-cert) to reflect the new setting.

```
$ /opt/icewall-sso/certd/bin/reload-cert
```

3.9.2 Main blocking

Stop requesting to the Authentication Module and forcibly log out all logged-in users. Requests from the Authentication Module commands (end-cert, reload-cert, others) can be accepted without stopping.

(1) Configure the request control configuration file (request.alc) to stop accepting requests.

Request control configuration file (request.acl)

```
TARGET=!(SOURCE_ADDR=127.0.0.1)
{
REJECT=ALL
}
```

(2) Execute the reload command (reload-cert) to reflect the new setting.

```
$ /opt/icewall-sso/certd/bin/reload-cert
```

(3) Execute the log out all users command (logout-cert) to forcibly log out all logged-in users.

```
$ /opt/icewall-sso/certd/bin/logout-cert
```

4 Advanced Configuration

Detailed configuration settings for access and authentication are described below.
For details on individual parameters, see the “IceWall SSO Reference Manual.”

4.1 Access control

IceWall SSO access control is described below.

4.1.1 Concept of access control

IceWall SSO access control uses the “group level access permission method” that grants access to a requested URL only to permitted groups.

The IceWall SSO administrator configures each user to belong to one or more groups, according to the access privileges to be granted to each user.
When configuring access control, note that access control is sequentially checked starting from the first line in the access control list. See “4.1.4 Notes on the sequential order of description for access control configuration” for details.

4.1.2 Creating various groups

Examples for the creation of a typical group are shown here. For more details on the configuration settings, see “Group configuration file (cert.grp)” in the “IceWall SSO Reference Manual.”
Values are described with regular expressions. For more information on how to write regular expressions, see the manual of the operating system.

(1) Creating a group of anonymous users
A group comprised of all users can be created in the following manner (In this example, “ANONYMOUS” is the group name and “UID” is the Authentication DB user ID column name.):

Group configuration file (cert.grp)

```
ANONYMOUS,UID=.*
```

(2) Creating a group of specific users
A group comprised of specific users can be created in the following manner (in this example, the group “group01,” comprised of users “user01” and “user02,” is created. "UID" is the Authentication DB user ID column name.):

Group configuration file (cert.grp)

```
group01,UID=^user01|UID=^user02
```

(3) Creating a group based on IP addresses
A group comprised of users accessing from particular IP addresses is created in the following manner:

Group configuration file (cert.grp)

```
To specify a particular IP address:
group02,REMOTE_ADDR=192.168.1.30|REMOTE_ADDR=192.168.1.50

To specify a range of IP addresses:
group03,REMOTE_ADDR=192.168.1.1-192.168.1.120
```

(4) Creating a group using user information
A group creation from user information saved in the Authentication DB is performed in the following manner:

Group configuration file (cert.grp)

```
To target users with 0 set in the APENABLE column
group04,APENABLE=^0
```

4.1.3 Various access control configurations

Typical access control configuration examples are shown here.

(1) Configuration of general access control
Example:
The URL that can be referenced by user10 and user11, belonging to group04, as well as user12 and user13, belonging to group05, is limited to “http://backend01.jpn.hp.com/secure/.”

① These users are configured to belong to group04 and group05 in the group configuration file. ("UID" is the Authentication DB user ID column name.)

Group configuration file (cert.grp)

```
group04,UID=user10|UID=user11
group05,UID=user12|UID=user13
```

② URLs and groups are configured in the access control file.

Access control file (cert.acl)

```
http://backend01.jpn.hp.com/secure/=group04|group05
```

(2) Configuring to allow only particular users to reference all URLs
Example:
Users “johnson” and “williams”
are permitted to access all URLs that start with “http://.”

① A group, which includes particular users only, is configured in the group configuration file (“ADMIN” is the group name and “UID” is the Authentication DB user ID column name here.).

Group configuration file (cert.grp)

```
ADMIN,UID=johnson|UID=williams
```

② URLs and groups are configured in the access control file.

Access control file (cert.acl)

```
http://=ADMIN
```

(3) Methods for simplifying the description of individual user directories
For example, let us assume that the following directories exist:
http://backend01.jpn.hp.com/~johnson/
http://backend01.jpn.hp.com/~williams/
http://backend01.jpn.hp.com/~brown/
A group can be configured in the root directory of the server to reference the above directories. In order to grant reference permission for these directories only, however, the access control file must be configured in the following manner:
(“UID” is the Authentication DB user ID column name.)

Group configuration file (cert.grp)

```
GRP01,UID=johnson|UID=williams|UID=brown
```

Access control file (cert.acl)

```
http://backend01.jpn.hp.com/~johnson/=GRP01
http://backend01.jpn.hp.com/~williams/=GRP01
http://backend01.jpn.hp.com/~brown/=GRP01
```

In this configuration method, the number of access control file definitions increases in proportion to an increase in the number of users.
In order to simplify these definitions, special keywords for the access control file are used. These special keywords are described as “%[column name of authentication DB]%.”

To configure the above URLs with special keywords, configure them as follows (the user ID column name is “UID”):

```
http://backend01.jpn.hp.com/~%UID%/=GRP01
```

4.1.4 Notes on the sequential order of description for access control configuration

The access control function searches for the requested URL in the access control file (cert.acl) starting from the beginning of the file, and grants access to the group that is included in the line of the first matching URL (prefix search). Once a line is found with a matching URL, no further search is performed on the remaining lines. For this reason, the order for specifying URLs in the configuration file is important when configuring access privileges for URLs.

Example 1

Access control file (cert.acl)

```
http://icewall-sso.co.jp/guest/=GUEST
http://icewall-sso.co.jp/=MEMBER-JP
http://icewall-sso.com/=MEMBER-US
```

If the above configuration is performed:
According to the first line: GUEST can access all URLs that start with “http://icewall-sso.co.jp/guest/.”
According to the second line: MEMBER-JP can access URLs that start with “http://icewall-sso.co.jp/.” Due to the configuration of the first line, however, MEMBER-JP cannot access URLs that start with “http://icewall-sso.co.jp/guest/.”
According to the third line: MEMBER-US can access all URLs that start with “http://icewall-sso.com/.”

The access control configuration of Example 1 is represented by the following table:

No	URL	GUEST	MEMBER-JP	MEMBER-US
1	http://icewall-sso.co.jp/	Forbidden	Permitted	Forbidden
2	http://icewall-sso.co.jp/guest/	Permitted	Forbidden	Forbidden
3	http://icewall-sso.com/	Forbidden	Forbidden	Permitted

Example 2

Access control file (cert.acl)

```
http://icewall-sso.com/=MEMBER-US
http://icewall-sso.co.jp/=MEMBER-JP
http://icewall-sso.co.jp/guest/=GUEST
```

If the above configuration is performed:

According to the first line: MEMBER-US can access all URLs that start with “http://icewall-sso.com/.”
According to the second line: MEMBER-JP can access all URLs that start with “http://icewall-sso.co.jp/.”
According to the third line: Configured so that GUEST is able to access all URLs that start with “http://icewall-sso.co.jp/guest/.” Due to the configuration of the second line, however, this becomes meaningless.

The access control configuration of Example 2 is represented by the following table:

No	URL	GUEST	MEMBER-JP	MEMBER-US
1	http://icewall-sso.co.jp/	Forbidden	Permitted	Forbidden
2	http://icewall-sso.co.jp/guest/	Forbidden	Permitted	Forbidden
3	http://icewall-sso.com/	Forbidden	Forbidden	Permitted

In Example 1, GUEST is able to access “http://icewall-sso.co.jp/guest/,” however, in Example 2, GUEST is unable to access the same URL.
Further, in Example 1, MEMBER-JP is unable to access “http://icewall-sso.co.jp/guest/,” however, in Example 2, MEMBER-JP is able to access the same URL.

Example 3

Access control file (cert.acl)

```
http://icewall-sso.co.jp/guest/=GUEST
http://icewall-sso.co.jp/member/=MEMBER-JP
http://icewall-sso.co.jp/=ADMIN
```

If the above configuration is performed:
According to the first line: GUEST can access all URLs that start with “http://icewall-sso.co.jp/guest/.”
According to the second line: MEMBER-JP can access all URLs that start with “http://icewall-sso.co.jp/member/.”
According to the third line: ADMIN can access URLs that start with “http://icewall-sso.co.jp/.” Due to the configuration of the first and second lines, however, ADMIN cannot access URLs that start with “http://icewall-sso.co.jp/guest/” or “http://icewall-sso.co.jp/member/.”

The access control configuration of Example 3 is represented by the following table:

No	URL	GUEST	MEMBER-JP	ADMIN
1	http://icewall-sso.co.jp/guest/	Permitted	Forbidden	Forbidden
2	http://icewall-sso.co.jp/member/	Forbidden	Permitted	Forbidden
3	http://icewall-sso.co.jp/	Forbidden	Forbidden	Permitted

Example 4

Access control file (cert.acl)

```
http://icewall-sso.co.jp/=ADMIN
http://icewall-sso.co.jp/guest/=GUEST
http://icewall-sso.co.jp/member/=MEMBER-JP
```

If the above configuration is performed:
According to the first line: ADMIN can access all URLs that start with “http://icewall-sso.co.jp/.”
According to the second line: Configured so that GUEST is able to access all URLs that start with “http://icewall-sso.co.jp/guest/.” Cannot be accessed because of the first line setting.
According to the third line: Configured so that MEMBER-JP is able to access all URLs that start with “http://icewall-sso.co.jp/member/.” Cannot be accessed because of the first line setting.

The access control configuration of Example 4 is represented by the following table:

No	URL	GUEST	MEMBER-JP	ADMIN
1	http://icewall-sso.co.jp/member/	Forbidden	Forbidden	Permitted
2	http://icewall-sso.co.jp/guest/	Forbidden	Forbidden	Permitted
3	http://icewall-sso.co.jp/	Forbidden	Forbidden	Permitted

In Example 3, ADMIN can access everything except “http://icewall-sso.co.jp/guest/,” which is accessible by GUEST, and “http://icewall-sso.co.jp/member/,” which is accessible by MEMBER-JP, but in Example 4, everything under “http://icewall-sso.co.jp/” is accessible by ADMIN.
In Example 4, GUEST and MEMBER-JP cannot access any of the URLs which they could respectively access in Example 3.

Example 5

Access control file (cert.acl)

```
http://=MEMBER-JP
http://=MEMBER-US
```

If the above configuration is performed:
According to the first line: MEMBER-JP can access all URLs that start with “http://” in Backend Web Servers.
Since the URLs listed in the first and second lines are identical, the configuration of the second line is meaningless.

4.2 Header control function and cookie control function

IceWall SSO header and cookie control are described below.

4.2.1 Header control when sending to the Backend Web Server

The HTTP headers sent from the client are all forwarded to the Backend Web Server, but it is possible to define headers not to be sent or to change the values of headers.

Host configuration file (sample.conf)

To configure the User-Agent header not to be sent to the backend web server:
HEADER=USER_AGENT,NOTSEND

To change the value of the User-Agent header and then send it to the backend web server:
HEADER=USER_AGENT,MODVALUE,Browser

To send the User-Agent header to the backend web server with a different name:
HEADER=USER_AGENT,MODNAME,CLIENT_BROWSER

To set the name of the user ID sent to the backend web server to IW_UID:
HEADER_NAME_UID=IW_UID

To set the name of the session ID sent to the backend web server to IW_SESSION:
HEADER_NAME_SID=IW_SESSION

It is possible to define certain user IDs and session IDs among those individually sent by the IceWall SSO not to be sent to the Backend Web Server.

Host configuration file (sample.conf)

To prevent sending user IDs to a backend web server:
HEADER_FILTER=UID

To prevent sending user IDs and session IDs to a backend web server:
HEADER_FILTER=UID
HEADER_FILTER=SESSION

4.2.2 Cookie control when sending to the Backend Web Server

It is possible to set some cookies among those sent by a client not to be sent to a Backend Web Server.

Host configuration file (sample.conf)

To configure “CORP” cookies not to be sent to a backend web server:
COOKIE_FILTER=CORP

To configure “DATA” and “FLAG” cookies not to be sent to a backend web server:
COOKIE_FILTER=DATA
COOKIE_FILTER=FLAG

4.2.3 Header control when sending to the browser

It is possible to set whether the Last-Modified header received from a Backend Web Server should be sent to the browser.

Host configuration file (sample.conf)

To prevent sending the Last-Modified header to the browser:
LASTMOD_HEADER=1

4.2.4 Header control for the response from the Backend Web Server

This setting configures how headers included in the response from the Backend Web Server are controlled.

Host configuration file (sample.conf)

To add “Sample-Header: xxxxx” to the response header:
RES_HEADER=SAMPLE_HEADER,ADD,xxxxx

To delete “Sample-Header: xxxxx” from the response header:
RES_HEADER=SAMPLE_HEADER,NOTSEND

To change header name “Sample-Header” to “Modified-Header”:
RES_HEADER=SAMPLE_HEADER,MODNAME,MODIFIED_HEADER

4.3 Basic authentication

IceWall SSO can connect to a Backend Web Server by using basic authentication.

4.3.1 Basic authentication configuration

Basic authentication settings are configured via the following parameters in the host configuration file.

Parameter name	Description
BASICAUTH	BASICAUTH=1
BA_UID	BA_UID=DEFAULT

Parameter name	Description
BA_PWD	BA_PWD=DEFAULT

The configuration above uses the user ID and password entered at login for the basic authentication user ID and password.
When specifying the Authentication DB columns as the user ID and password, set the column names configured in the column configuration file (dbattr.conf) to BA_UID and BA_PWD. When using columns other than those descripted above in the Authentication DB, you must configure them with the DBEXATTR parameter in the Authentication Module configuration file (cert.conf).

4.3.2 Basic authentication configuration for reverse proxy mode

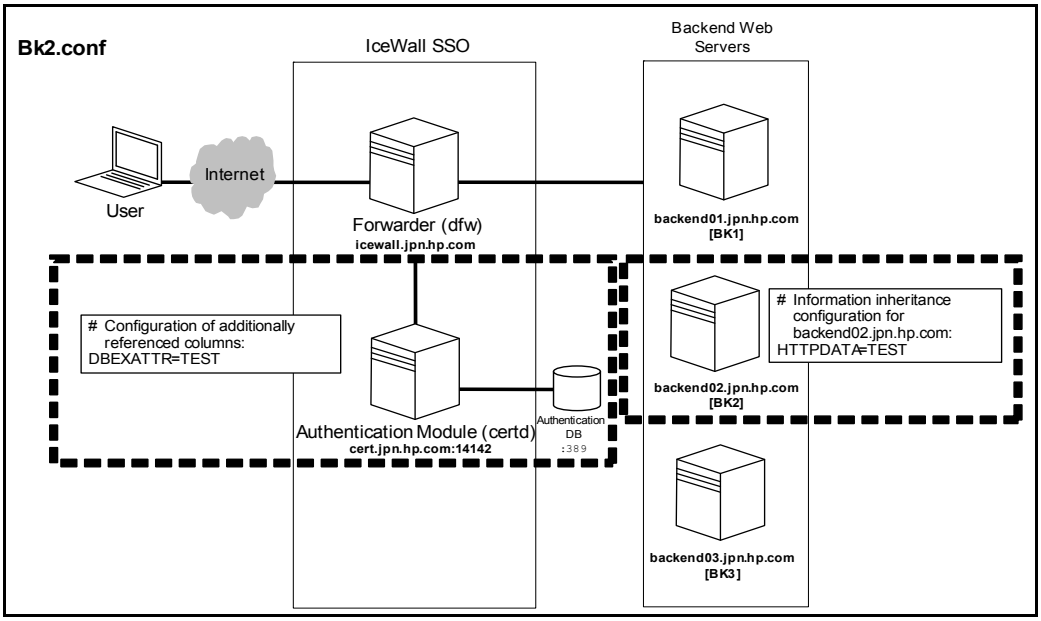
Because user authentication does not occur with IceWall SSO when operating in reverse proxy mode, basic authentication is not performed for Backend Web Servers. Similarly, basic authentication is not performed for URLs that are set not to require authentication.

4.3.3 Notes when using ICP 2.0 communication

When using ICP 2.0 to communicate with the Authentication Module, the password entered at login is not sent to the Backend Web Server, even if DEFAULT is set for the BA_PWD configuration parameter.
To send the password at login to the Backend Web Server, configure the Authentication Module's request control configuration file (request.acl) to include the password at login in the access control response message.

4.4 Information inheritance configuration

The information inheritance function can be used to configure the transfer of the data in specified Authentication DB columns to Backend Web Servers as HTTP headers. Through this method user IDs and passwords can be transferred to authentication processes performed via CGI scripts, and information in an Authentication DB can be transferred to applications running on Backend Web Servers. The entry added to the HTTP header is “authentication DB column name.”



The following configuration example represents a configuration for sending information in the TEST column of an Authentication DB.

Host configuration file (BK2.conf)

Parameter name	Description
HTTPDATA	HTTPDATA=TEST

When specifying a column name that is not configured in the Authentication DB column information file (dbattr.conf), the column information must be defined in the Authentication Module configuration file (cert.conf).

Authentication Module configuration file (cert.conf)

Parameter name	Description
DBEXATTR	DBEXATTR=TEST

4.5 Request control based on the access path

It is possible to restrict the user's access not only via the group configuration file and the access control file, but also via the request control configuration file.

4.5.1 Performing request control

Configure the Authentication Module configuration file to use the request control configuration file.

Authentication Module configuration file (cert.conf)

```
ACLREQUEST=/opt/icewall-sso/certd/config/request.acl
```

4.5.2 Request control from a specific Forwarder or agent

Using the request control configuration file, it is possible to block requests from specific Forwarders or agents.

(1) Specifying the server to block
To block Forwarder or an agent, set its IP address as follows:

Request control configuration file (request.acl)

```
TARGET=SOURCE_ADDR=[IP address]
{
}
```

(2) Specifying an IP range to block
To restrict the range of the request source, set the IP address of the request source server as follows:

Request control configuration file (request.acl)

```
TARGET=SOURCE_ADDR=[Starting IP address]-[Ending IP address]
{
}
```

4.5.3 Forbidding requests from a specific Forwarder or agent

Using the request control configuration file, it is possible to control the execution of requests sent from Forwarders or agents.

(1) To forbid password change requests from a specific Forwarder:

Request control configuration file (request.acl)

```
TARGET=SOURCE_ADDR=[IP address]
{
REJECT=PWDCHG
}
```

(2) To forbid forced login requests from a specific Forwarder:

Request control configuration file (request.acl)

```
TARGET=SOURCE_ADDR=[IP address]
{
REJECT=FLOGINUID
}
```

4.5.4 Restricting the user information sent to specific Forwarders or agents

Using the request control configuration file, it is possible to restrict the user information held by Authentication Modules that is normally sent entirely.

(1) To set the limited user information to be sent to a specific Forwarder:

Request control configuration file (request.acl)

```
TARGET=SOURCE_ADDR=[IP address]
{
SEND=LOGONDATE,PWDEXPDATE
}
```

(2) To specify user information not to be sent to a specific agent:

Request control configuration file (request.acl)

```
TARGET=SOURCE_ADDR=[IP address]
{
NOTSEND=LOGONDATE
}
```

4.5.5 Restricting access paths based on whether a client certificate is used

Using the request control configuration file, it is possible to restrict access to access paths that use/do not use a client certificate, regardless of the login method.

(1) To restrict access to access paths that do not use a client certificate:

Request control configuration file (request.acl)

```
TARGET=SOURCE_ADDR=[IP address]
{
ACCCTRL=uid
}
```

(2) To restrict access to access paths that use a client certificate:

Request control configuration file (request.acl)

```
TARGET=SOURCE_ADDR=[IP address]
{
ACCCTRL=cert
}
```

(3) To restrict access to access paths that use a client certificate and not to perform user ID verification:

Request control configuration file (request.acl)

```
TARGET=SOURCE_ADDR=[IP address]
{
ACCCTRL=certnouid
}
```

4.6 Agent control based on the request source

By configuring Forwarder, it is possible to control connections by specifying the agent sending the request. This allows for blocking requests from unknown sites, providing increased security.

4.6.1 Agent control based on the IP address

The agent sending the request is controlled based on its IP address. Even if the agent is specified as a host name (including full domain name), matching is performed after converting it to an IP address. A range can be specified, but regular expressions cannot be used.

Forwarder configuration file (dfw.conf)

```
AGENT_KEY=AGENT_KEY1,AGENT1
AGENT_PERMIT=AGENT1,IP,xxx.xxx.xxx.xxx
```

4.6.2 Agent control based on the host name

The agent sending the request is controlled based on its host name. In this case, a complete matching is performed on the host name.

Forwarder configuration file (dfw.conf)

```
AGENT_KEY=AGENT_KEY2,AGENT2
AGENT_PERMIT=AGENT2,HOST,agenthost
* Matching is performed only if only the host name is specified.

AGENT_PERMIT=AGENT2,HOST,agenthost.co.jp
* Matching is performed only if the full domain name is specified.
```

4.6.3 Agent control based on the domain name

The agent sending the request is controlled based on its domain name. In this case, matching is performed as a backwards match on only the domain name. However, regular expressions cannot be used.

Forwarder configuration file (dfw.conf)

```
AGENT_KEY=AGENT_KEY3,AGENT3
AGENT_PERMIT=AGENT3,DOMAIN,agentdomain.co.jp
```

4.7 Forbidding connections from clients by request type 10.0

From version 10.0, accesses can be forbidden by the type of connection request from a client. In this way, an error page can be displayed when a client connects with a certain type of request.
Use this when you wish to temporarily stop access for maintenance or other reasons and when service is not provided for a type of request.

•To temporarily forbid access for requests related to logging in:
Forwarder configuration file (dfw.conf)

```
REQUESTFILTER=LOGIN
```

•To forbid access for unsupported agent linking requests:
Forwarder configuration file (dfw.conf)

```
REQUESTFILTER=AGTLOGIN,AGTPWDCHG
```

4.8 Password configuration

IceWall SSO password configuration is described below.

4.8.1 Using the password change function

The password change function can be used by configuring an alias for the function and then accessing that alias.

(1) Configuring an alias for the password change function
When using the password change function, the following format is used to configure an alias to call the password change page:

Forwarder configuration file (dfw.conf)

```
PWDALIAS=IW-PWDCHG
```

(2) Displaying the password change page
Access http://icewall.jpn.hp.com/fw/dfw/IW-PWDCHG/.
* When you change PWDALIAS, also change the link to the password change page.

4.8.2 Changing the password policy 10.0

The password policy can be modified using the Authentication Module configuration file (cert.conf).

* From version 10.0, the settings related to the password policy have been migrated from Forwarder to the Authentication Module. The settings in Forwarder are valid as they have been in previous versions, but we recommend migrating the settings to the Authentication Module side.

(1) Changing the maximum password length
This parameter configures the maximum number of characters that may be used when updating a password.
In the following example, the maximum number of characters is set to 10.

Authentication Module configuration file (cert.conf)

```
PWDMAXLEN=10
```

(2) Changing the minimum password length
This parameter configures the minimum number of characters that may be used when updating a password.
In the following example, the minimum number of characters is set to 5.

Authentication Module configuration file (cert.conf)

```
PWDMINLEN=5
```

(3) Changing the password expiration date
The password expiration date is set as follows for an expiration time of 72 days:

Authentication Module configuration file (cert.conf)

```
PWDEXPIRE=72
```

(4) Changing the warning period for password expiration

This parameter configures whether a warning prompting the user to change their password is displayed before the password expires. If the warning is set, the warning page is displayed at the time of login only if login occurs within the warning period leading up to the password expiration date.
In the following example the password expiration warning period is set to start 14 days before the expiration date.

Authentication Module configuration file (cert.conf)

```
PWDEXPWARN=14
```

(5) Checking for password expiration

This parameter sets whether to check for password expiration.
If the check is set, the Authentication Module checks for password expiration at login, and displays the password change page if the password has expired.
In the following example, the password expiration check is enabled.

Authentication Module configuration file (cert.conf)

```
PWDEXPCHK=1
```

(6) Allowing or disallowing of passwords identical to the user ID

This parameter configures whether to allow or disallow passwords identical to the user ID.
For security purposes, it is recommended that the use of passwords identical to user IDs be disallowed.
The configuration in the following example disallows the use of a password identical to the user ID.

Authentication Module configuration file (cert.conf)

```
PWDSAMEPASS=1
```

(7) Configuring the types of characters that may be used in passwords

This parameter configures the combination of alphabetic, numeric, and special characters that can be used for passwords. For security purposes, it is recommended to disallow passwords that contain only alphabetic, or only numeric characters. (This means that requiring passwords to contain both alphabetic and numeric characters is recommended.)

The configuration in the following example disallows passwords that do not contain both alphabetic and numeric characters.

Authentication Module configuration file (cert.conf)

```
PWDALPHANUM=1
```

Special characters that can be used in passwords are as follows, excluding blank spaces:
!”#$%&( )*+,-./:;<=>?@[\]^_`{|}~
* Single quotes cannot be used starting with version 8.0 R2.

4.8.3 To disallow changing a password to a previously used password

Password changes can be configured to disallow changing the password to passwords that have been used in the past. The following setting disallows the use of prior passwords. (The current password cannot be used, regardless of this setting.)

(1) Specify the Authentication DB column that stores previously used passwords.

Authentication DB column information file (dbattr.conf)

```
PWDHISTORY=PASSWDHIS
```

(2) The parameter below sets whether to check for previously used passwords.

Authentication Module configuration file (cert.conf)

```
PWDHISCHK=1
```

(3) The parameter below configures the number of prior passwords to archive. The configuration in the following example is set to store the past three passwords, excluding the currently used password. The configuration can be set to store a maximum of 20 prior passwords.

Authentication Module configuration file (cert.conf)

```
PWDHISCNT=3
```

4.8.4 Defining strings not to be used as passwords 10.0

Setting strings that are difficult for others to guess as passwords improves security. The configuration in the following example disallows the use of particular strings as passwords.

* From version 10.0, you can use regular expressions. Because of this change, the

operation of the existing settings changes from complete matches to partial matches. Use caution when transitioning the settings from older versions.

(1) Specify the forbidden password configuration file.

Authentication Module configuration file (cert.conf)

```
PWDFORBID=/opt/icewall-sso/certd/config/pwdforbid.conf
```

(2) Use the forbidden password configuration file (pwdforbid.conf) to specify the strings that are disallowed for use as passwords.

To forbid a complete match of the password "0000":
^0000$

To forbid a password that contains "USER01":
USER01

To forbid a character string that does not contain one or more numbers as the password:
^[^0-9]*$

The error log below is output when the regular expression that was set contains an error. However, even if there is an error, only the corresponding line is ignored and processing continues.

```
Cannot compile regular expression. Value=[configuration value] ErrMsg=[error message]
```

4.9 UserExit routine

The UserExit routine is a function that enables users to add arbitrary processes to particular segments of processes, such as analyzing, sending, and receiving requests as performed by the Forwarder, and authentication/authorization of the Authentication Module. For example, an authentication process can be implemented with other security products. For details, see the “IceWall SSO UserExit Routine Developer's Manual.”

4.10 IceWall SSO multi-tier architecture

A site that already has IceWall SSO installed can be operated as a Backend Web Server of another IceWall SSO site. This section describes the necessary configurations and restrictions for configuring a multi-tier architecture.
For convenience, IceWall SSO accessed directly by the browser is referred to as “front-end IceWall,” while IceWall acting as a Backend Web Server is referred to as

"backend IceWall."

4.10.1 Prerequisite conditions for a multi-tier architecture

If Forwarder is operated with the same CGI prefix for both the front-end IceWall and the backend IceWall, then the URL of the backend IceWall is output without being converted by the front-end IceWall.
This is due to the specification of Forwarder, which "does not reconvert any URL that has already been converted to a URL via an IceWall SSO server." For this reason, configure the CGI prefix differently for the front-end IceWall and the backend IceWall when configuring a multi-tier architecture.

Example: the CGI prefix of both front-end and backend IceWall is "/fw"

This is for the case when the following URL is included in the content output via backend IceWall.
SVR1

When front-end IceWall receives this URL to convert it to a URL via front-end IceWall, it recognizes this as an "already converted URL," as the CGI prefix is the same for both the front-end and backend IceWall. For this reason, the front-end IceWall does not convert this URL. Thus, the URL output by the front-end IceWall is as follows:

SVR1

When the CGI prefix is different for the front-end and the backend, the URL is as following (in this example, the CGI prefix for the backend is "/icewall," and the alias for backend IceWall at the front end is "IWSVR"):

The URL at backend IceWall:
SVR1
The URL at front-end IceWall:
SVR1

4.11 Changing the names of authentication cookies

The names of authentication cookies for front-end and backend IceWall are to be configured to be different from each other in the Forwarder configuration file (dfw.conf).

Example: configuring the names of authentication cookies

Front-end IceWall
COOKIENAME=IW_FRONT

Backend IceWall
COOKIENAME=IW_BACK

Note: this configuration setting is required for either front-end or backend IceWall.

4.11.1 Configuring keywords for handling authentication information

Keywords can be configured for handling authentication information sent to Forwarder from the browser for login, logout, and password updates. In a multi-tier architecture, different keywords must be configured for both front-end and backend IceWall. These keywords are configured in the Forwarder configuration file (dfw.conf).

Parameter name	Description
POSTKEY_LOGIN	Keywords for identifying login information are configured.
POSTKEY_LOGOUT	Keywords for identifying logout information are configured.
POSTKEY_PWDCHG	Keywords for identifying password change information are configured.

Note: this configuration setting is required for either front-end or backend IceWall.

Example: configuration of the authentication data handling keywords (for configuring keywords for both the front and back ends):

Front-end IceWall
POSTKEY_LOGIN=**FRONT_LOGIN**
POSTKEY_LOGOUT=**FRONT_LOGOUT**
POSTKEY_PWDCHG=**FRONT_PWDCHG**

Backend IceWall
POSTKEY_LOGIN=**BACK_LOGIN**
POSTKEY_LOGOUT=**BACK_LOGOUT**
POSTKEY_PWDCHG=**BACK_PWDCHG**

4.11.2 Restrictions regarding a multi-tier architecture

The restrictions for operating IceWall SSO in a multi-tier architecture are as follows:

- The values of the Uid and Session HTTP headers configured by both front-end and backend IceWall are sent to the Backend Web Servers of backend IceWall.

Example: front-end IceWall user ID: user01, session ID: xxx to xxx
backend IceWall user ID: user02, session ID: zzz to zzz

The values of the HTTP header sent from front-end IceWall to the backend web servers
Uid: user01
Session: xxx to xxx

The values of the HTTP header sent from backend IceWall to the backend web servers
Uid: user01,user02
Session: xxx to xxx,zzz to zzz
* The order of these values is not fixed.

If the front-end IceWall HTTP headers, described above, are not required by the backend IceWall, they can be prevented from being sent by using the following methods:

(1) by not sending the applicable HTTP headers from the front-end IceWall.
(2) by not sending the applicable front-end IceWall HTTP headers from the backend IceWall.

In the case of method (1), the following configurations are added to the front-end host configuration file:

Front-end IceWall host configuration file (sample.conf)

```
HEADER_FILTER=UID
HEADER_FILTER=SESSION
```

In the case of method (2), the following configurations are added to the backend host configuration file:

Backend IceWall host configuration file (sample.conf)

```
HEADER=HTTP_UID,NOTSEND
HEADER=HTTP_SESSION,NOTSEND
```

4.12 Handling HTTP-compressed content

When compressed content is sent by Backend Web Servers that support HTTP compression, a system function excludes such content from URL conversion. This is effective when a mix of content compressed by HTTP compression and conventional uncompressed content exist on the same Backend Web Server.

4.12.1 Configuring the transfer of HTTP-compressed content

The following settings are specified in the host configuration file of the Backend Web Server that is handling HTTP compression in Forwarder:

Host configuration file (sample.conf)

```
CENCODE=1
```

Example: configuration for HTTP compression handling

This setting removes HTTP-compressed HTML files, sent by the backend web servers, from a list of files subject to conversion.
CTYPE=text/html
CENCODE=1
In this case, content conversion is performed on HTML files that have not undergone HTTP compression.

4.12.2 Restrictions on HTTP-compressed content

When HTTP-compressed content is transferred, none of the URLs contained in the data are converted, because the data is removed from the list of content subject to conversion.
However, the location and the Set-Cookie HTTP headers are subject to conversion.

4.13 Unicode conversion of user data (for LDAP, OpenLDAP, NED, and MSAD editions only)

When using LDAP, OpenLDAP, NED, or MSAD as the Authentication DB, the character encoding of the user data stored in Unicode (UTF-8) in the Authentication Module is converted to Shift_JIS, however, it is possible to specify the type of character encoding to convert to here. This function is only available for the LDAP, OpenLDAP, NED, and MSAD editions.

4.13.1 Configuring the character encoding conversion

The character encoding conversion is specified in the Authentication Module configuration file and in the startup command (start-cert) of the Authentication Module. The method for configuration is described below:

(1) Change to the directory where the configuration file is located.

```
# cd /opt/icewall-sso/certd/config
```

(2) Open the Authentication Module configuration file (cert.conf) with an editor.

```
# vi cert.conf
```

(3) Perform the following setting.

```
LDAPLANG=1
```

(4) Save the configuration file and close the editor.

(5) Change to the directory where the startup command script is located.

```
# cd /opt/icewall-sso/certd/bin
```

(6) Open the startup command script with an editor.

```
# vi start-cert
```

(7) Add the following line. Add this line before the line that contains the command to execute the Authentication Module.

```
IW_LANG=[keyword for specifying the character encoding]
```

Specify one of the following character encoding types:

Character encoding name	Character encoding specification keyword
Shift_JIS	sjis
Big5	big5

(8) Save the startup command script and close the editor.

Restart the Authentication Module when the character encoding conversion specification has been added or changed.

If a replication architecture is used with the failover option installed, the setting above must be performed on both the master and the slave.

4.13.2 Restrictions on character encoding conversion

Only the following character encoding types support conversion:

- Shift_JIS
- Big5 (traditional Chinese characters)

4.14 Sending, after login, POST data that was sent before login

Even if the IceWall SSO session expires by the time the data that was typed in using

an input form is being sent to an application, it is possible to resend the data to the application after login.

However, the following restrictions apply to the use of this function:
- The Content-type of the POST data must be “application/x-www-form-urlencoded”
- The maximum data size is 1KB
- A browser supporting JavaScript must be used

4.14.1 Using POST data inheritance

To use POST data inheritance, apply the following settings:

Forwarder configuration file (dfw.conf)

```
POST_INHERIT=1
POST_HTML=/opt/icewall-sso/dfw/cgi-bin/iw_postdata.html
```

The HTML file specified in the POST_HTML parameter is the HTML template file that includes the JavaScript for automatically sending the inherited POST data from the client.

4.14.2 Restricting the size of the inherited POST data

It is possible to limit the maximum size of the inherited POST data. If the size of the data for inheritance exceeds the limit, a specific error page can be displayed.
This function is configured as follows:

Forwarder configuration file (dfw.conf)

```
MAXPOST=1024
```

HTML configuration file (html.conf)

```
MAXPOST_ERR=file:///opt/icewall-sso/dfw/html/max_postsize_err.html
```

4.14.3 Setting the parameter name to use when inheriting POST data

Since the POST data inheritance function uses a Query String as a temporary intermediary, it is necessary to use a parameter name to determine whether the data is inherited POST data. This name can be chosen arbitrarily.
This function is configured as follows:

Forwarder configuration file (dfw.conf)

```
POSTNAME=iwpost
```

The value set here must be different than the value set in the AGENT_KEY parameter or the QUERY_ENC_NAME parameter, and it cannot be used as a parameter of the QueryString of the Web application.

4.14.4 Encrypting inherited POST data

Since the POST data inheritance function uses a Query String as a temporary intermediary, it is possible to encrypt the POST data when it is inherited.
This function is configured as follows:

Forwarder configuration file (dfw.conf)

```
POST_ENC= abc123ABC-_!
```

4.15 Using IceWall Cert Protocol (ICP) 2.0

The communication between Forwarder and the Authentication Module can be configured to use IceWall Cert Protocol (ICP) 2.0. The traditionally used communication protocol is called ICP 1.0.

4.15.1 Configuring the communication to ICP 2.0

Use the following configuration to communicate via ICP 2.0:

Forwarder configuration file (dfw.conf)

```
ICP_VERSION=2.0
```

There is no need to set anything in the Authentication Module.
Whether ICP 1.0 or ICP 2.0 is used is analyzed based on the request received.

4.15.2 Identifying the source of an ICP 2.0 request

When using ICP 2.0, it is possible to set a unique identifier at Forwarder, and since the Authentication Module outputs that identifier into the access log, the source of that request can be identified.
This function is configured as follows:

Forwarder configuration file (dfw.conf)

```
ICP_AGENTSTR=DFW
```

Authentication Module configuration file (cert.conf)

```
LOGINFO=2
```

4.16 Extending the ICP 2.0 communication message encryption library 10.0

You can use both communication messages encrypted by the ICP 2.0 communication message encryption library provided with the product and communication messages encrypted by an ICP 2.0 communication message encryption library developed for yourself in a single Authentication Module at the same time.
To encrypt ICP 2.0 communication messages with your own encryption method, you must develop a customized communication message encryption library. For details, see the “IceWall Cert Protocol 2.0 Developer's Manual.”

4.17 ICP 2.0 HTTP support 10.0

ICP 2.0 requests and responses can be handled with the HTTP interface.

4.17.1 HTTP requests

HTTP requests to the Authentication Module must be sent in the format below.

Request line [CRLF]
Header(s) [CRLF]
[CRLF]
Body

The HTTP request below is a sample.

```
POST / HTTP/1.0
Content-Length: 79

MSG_ID: ReqLoginUID
IW_UID: user01
IW_PWD: user01pass
REMOTE_ADDR: 127.0.0.1
```

* For reference the body section is plain text, but it would actually be encrypted with the communication message encryption library.

(1) Request line

The request line contains “Request method,” “Request URI,” and “HTTP version.” The request line is shown below.

[Request method] △ [Request URI] △ [HTTP version][CRLF]

Notes when specifying the request line:

- The request method can be POST only.
- Define the request URI as an ASCII character string. The maximum length is 256 bytes. (However, “/healthcheck” cannot be used, because this is reserved for an alive check of the Authentication Module described later.)

- The HTTP version can be specified as either HTTP/1.0 or HTTP/1.1.

(2) Request headers

Notes when specifying the request headers:

- The headers must conform to the ICP 2.0 header specification*.
  (if HTTP specification headers are sent, they are not processed.)
- Set the header name and header value each within 1,024 bytes.
- Up to 512 headers can be specified.
- The "Content-Length" header is required.
- Not case sensitive.
- Define the header name and header value as ASCII character strings.
  (Other character encodings must be URL encoded.)

(3) Request body

Notes when specifying the body:

- The body must conform to the ICP 2.0 body specification*.
- The body must be encrypted because it is decrypted by the Authentication Module's communication message encryption library.

* For details, see “3.2 Message format” in the “IceWall Cert Protocol 2.0 Developer's Manual.”

4.17.2 HTTP responses

The response format after processing requests is shown below.

Status line [CRLF]
Header(s) [CRLF]
[CRLF]
Body

The HTTP response below is a sample.

```
HTTP/1.0 200 OK
Content-Length: 63

MSG_ID: ResAccessUID
ERR_NO: 0
IW_UID: user01
USERID: user01
```

* For reference the body section is plain text, but it would actually be encrypted with the communication message encryption library.

(1) Status line

The status line contains “HTTP version,” “Status,” and “Status description.”

The status line is shown below.

[HTTP version] △ [Status] △ [Status description][CRLF]

Notes when receiving the request line:
- The HTTP version can be HTTP/1.0 only.
- The status and status description are only specified as “200 OK” when the operation was successful, and “500 Internal Server Error” when the operation failed.

(2) Response headers

Notes when receiving the headers:
- The headers must conform to the ICP 2.0 header specification*.
- The header name and header value are each set within 1,024 bytes.
- The maximum number of sent headers is 512.
- The “Content-Length” header is always set.
- The header name and header value are specified as ASCII character strings. (Other character encodings must be URL encoded)

(3) Response body

Notes when specifying the body:
- The body must conform to the ICP 2.0 body specification*.
- The body is always encrypted.

* For details, see “3.2 Message format” in the “IceWall Cert Protocol 2.0 Developer's Manual.”

4.17.3 Authentication Module HTTP send/receive settings

To send and receive ICP 2.0 with the HTTP interface, you must specify the HTTP listen port.

To use 8082 as the HTTP interface listen port, configure it as shown next.

Authentication Module configuration file (cert.conf)

HTTPPORT=8082

4.18 Specify the number of request threads that do not require DB connections (option) 10.0

You can ensure that requests that do not require DB connections have no impact even when a failure occurs on the DB by configuring the number of request threads that do not require DB connections in the ACCTHREAD parameter in the Authentication Module configuration file (cert.conf).
The number of the request threads configured in the ACCTHREAD parameter are allocated for requests that do not require DB connections such as authorization requests and Authentication Module command requests.

A DB busy error is returned when the number of requests that require DB connections exceeds the value of MAXREQTHREAD - ACCTHREAD.

4.19 Password encryption library (10.0)

SHA256 has been added as the SHA-2 encryption method. The password encryption libraries have also been merged into a single library which makes it possible to set the password encryption method when logging in and when changing passwords.

4.19.1 Specifying the password encryption method

The password encryption libraries have been merged so the encryption algorithm when logging in and when changing passwords can be set with the PWDLOGINHASH and PWDCHGHASH parameters in the Authentication Module configuration file (cert.conf).
Since the encryption algorithm can now be specified in the configuration, you can mix multiple algorithms.

The following encryption methods can be specified.

Setting value	Description
MD5	Perform MD5 encryption.
SHA	Perform SHA1 encryption.
SHA256	Perform SHA256 encryption.
PLAIN	No encryption.
AUTO-PLAIN	Determine the encryption method from the prefix. (No encryption if the prefix is missing.)
AUTO-MD5	Determine the encryption method from the prefix. (MD5 encryption if the prefix is missing.)

The following example is the configuration to set the encryption method when logging in and when changing passwords to SHA256:

Authentication Module configuration file (cert.conf)

```
PWDLOGINHASH=SHA256
PWDCHGHASH=SHA256
```

Note the following items when using this function:
- Users may not be able to log in if the encryption method setting used on logging in and on changing passwords are different.
- During BIND authentication, PWDLOGINHASH and PWDCHGHASH must be configured to use the same encryption algorithm (other than an AUTO-based algorithm). (During BIND authentication, the encryption algorithm cannot be automatically determined by the prefix, so the encryption algorithm when logging in must be configured. If the encryption algorithm when logging in is configured as

an algorithm other than an AUTO-based algorithm, users may not be able to log in after changing their passwords if the encryption algorithm when changing the password is configured as a different algorithm.)

4.19.2 Merging the password encryption libraries

In previous versions, there were libraries for MD5, SHA1, and no encryption. From version 10.0, these libraries have been merged as libiwPwdHash.sl.

4.19.3 SHA256 encryption method

From version 10.0, SHA256 has been added to the password encryption library.

Character string example encrypted with SHA256

```
{SHA256}xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

* 8-byte prefix + 44-byte hash value = 52 bytes total

4.20 Other useful functions

In addition to the functions described above, IceWall SSO has many other useful functions. This section describes some of those functions.

4.20.1 Operation in reverse proxy mode

This function provides only content transfer by using only the reverse proxy function of Forwarder without the authentication function of the IceWall SSO.
This function is configured as follows:

Forwarder configuration file (dfw.conf)

```
#CERT=localhost:14142  * Comment out the CERT parameter.
```

4.20.2 Configuring authentication cookie attributes

During normal usage, the attribute configuration of authentication cookies is not necessary; however, this function is effective when identical authentication cookies are sent by multiple Forwarders.
This function is configured as follows:

Configure the COOKIEATTR parameter in the Forwarder configuration file (dfw.conf).

```
Example 1: When configuring the domain attribute to “hp.com” and the path attribute to “/fw/dfw/.”
 COOKIEATTR=domain=hp.com; path=/fw/dfw

Example 2: To set the secure attribute in an SSL connection environment:
 COOKIEATTR=secure
```

4.20.3 Configuring a no-authentication URL/extension

Some URLs and request extensions can be configured not to require authentication (same operation as that of the reverse proxy mode) even when using the IceWall SSO authentication function.
This function is configured as follows:

(1) To not require authentication for some URLs:
Configure the REV_PATH parameter in the Forwarder configuration file (dfw.conf).
Example: to not require authentication for http://regist.svr.com/user/regist.cgi

```
REV_PATH=http://regist.svr.com/user/regist.cgi
```

(2) To not require authentication for some request extensions:
Configure the NOCHK_EXT_ALIAS parameter in the Forwarder configuration file (dfw.conf).
Example: to not require authentication for gif and jpeg extensions in a case sensitive way throughout the system

```
NOCHK_EXT_ALIAS=*,1,gif,jpeg
```

4.20.4 URL-Cookie authentication method

This function enables operation with session management using URL-cookies, which sends authentication cookies embedded in the URL when using IceWall SSO from a terminal that does not support cookies.
This function is configured as follows:

Forwarder configuration file (dfw.conf)

```
SESSION=1
DOCS=/opt/icewall-sso/dfw/cgi-bin/chtml.conf
```

If the configuration is changed, the URL to be accessed changes as follows:

Before configuration: http://www.svr.com/fw/dfw/sys/index.html
After configuration: http://www.svr.com/fw/dfw/**login**/sys/index.html

After authentication, the bold type “login” part of the URL changes to the authentication cookie.

4.20.5 Replacing error pages

Certain error pages sent from the Backend Web Servers can be replaced with different error pages on IceWall SSO.
This function is configured as follows:

Host configuration file (sample.conf)

Example: to change pages that contain “Server Error” to “/opt/icewall-sso/dfw/html/system_error.html”:

```
ERRKEY=Server Error,/opt/icewall-sso/dfw/html/system_error.html
```

4.20.6 Support for load balancing systems

If the browser connects to a load balancing system via SSL, and the load balancing system connects to Forwarder via HTTP, then the protocol of the URL information included in the HTTP header must be changed. This function is effective when using a load balancing system product that does not perform this change.
This function is configured as follows:

Forwarder configuration file (dfw.conf)

Example: to correctly operate Forwarder when the browser connects to the load balancing system via SSL, and the load balancing system connects to Forwarder via HTTP:

```
DFW_PROTOCOL=1
```

Note: Forwarders operating in an SSL environment cannot be operated in an HTTP environment. If this parameter is set to 1 in an environment without a load balancing system, correct operation of Forwarder is not guaranteed.

4.20.7 Setting the maximum number of simultaneous user logins

Configure the maximum number of users that can simultaneously log in to the Authentication Module.
This function is configured as follows:

Authentication Module configuration file (cert.conf)

Example: to set the maximum number of simultaneous user logins to 1000: CACHE=1000

Note: If multiple logins are performed for the same user ID when duplicate login is allowed (DUPLOGIN=1), each logged in user is counted as a valid user.

The amount of memory consumed for one logged-in user can be estimated using the memory calculation Excel worksheet made available on the support site.

4.20.8 Controlling response header URL conversion (for virtual hosts)

This function is used to prevent conversion of the URL of the Location and Set-cookie headers received from the Backend Web Server. This function is necessary when using a virtual host.
This function is configured as follows:

Host configuration file (sample.conf)

Example: to prevent conversion of the URL of the Location and Set-cookie headers: UNCONV_HEADER=LOCATION,SET-COOKIE

4.20.9 Function to configure the method of acquiring request path information

This function sets the method of acquiring the path information requested by the client. If a "=" exists in the path requested including a parameter such as "jsessionid," it is encoded as "%3d" and may cause the application to malfunction. In this kind of situation, change the path information retrieval method.
This function is configured as follows:

Forwarder configuration file (dfw.conf)

Example 1: to use the conventional method for acquiring the path info: REQUEST_URI=0 Example 2: to use the new method for acquiring the path info: REQUEST_URI=1

4.20.10 Changing the maximum number of allowable user logins 10.0

The number of allowable user logins can be changed without restarting the Authentication Module. Since the number of users can be configured for the Authentication Module with the reload command (reload-cert), there is no impact on the users already logged in.

The limit for the number of users that can be configured is up to the value for the maximum number of simultaneous logins (CACHE parameter).
This function is configured as follows:

Authentication Module configuration file (cert.conf)

```
MAXLOGINUSER=40
```

The behavior where the number of allowable user logins (MAXLOGINUSER parameter) and the maximum number of simultaneous logins (CACHE parameter) are configured is described below.

(1) For the case below, the number of logged-in users is fewer than the value for the MAXLOGINUSER parameter, so users up to the number configured by the MAXLOGINUSER parameter can log in.

```
CACHE=100
MAXLOGINUSER=10
(Logged-in users=5)
```

(2) For the case below, the number of logged-in users is larger than the value for the MAXLOGINUSER parameter, but processing can be continued for all logged-in users in this state. New logins are not possible.
However, when logged-in users log out and the number falls below the value for the MAXLOGINUSER parameter, new logins are possible.

```
CACHE=100
MAXLOGINUSER=10
(Logged-in users=20)
```

(3) For the case below, new logins are not possible even when logged-in users log out. Processing can be continued for logged-in users in this state.

```
CACHE=100
MAXLOGINUSER=0
(Logged-in users=10)
```

(4) For the case below, the value for the MAXLOGINUSER parameter is larger than the CACHE value, so the number of allowable user logins is the CACHE parameter value.

```
CACHE=100
MAXLOGINUSER=200
(Logged-in users=20)
```

4.20.11 Performing an Authentication Module alive check 10.0

You can perform an alive check for Authentication Modules from such as load balancing equipment by sending an alive check request to the Authentication Module.
The Authentication Module alive check can be an ICP 2.0 request and an HTTP-formatted request.
However, please note that the alive check response will slow down while the Authentication Module reload command (reload-cert) is being processed because the request threads stop.

(1) ICP 2.0 Authentication Module alive check request
To perform an alive check for an Authentication Module, you must send the request below.

```
REQ /healthcheck ICP/2.0 [CRLF]
Content-length: 0 [CRLF]
[CRLF]
```

* The request header request line “REQ /healthcheck ICP/2.0” is required.

(2) ICP 2.0 Authentication Module alive check response
When the request above is received, the response below is sent when the Authentication Module is up.

```
ICP/2.0 200 OK [CRLF]
Content-length: 0 [CRLF]
[CRLF]
```

* If the Authentication Module is down, no response is returned.

(3) HTTP-formatted Authentication Module alive check request
To perform an HTTP alive check, send the request below.

```
POST /healthcheck HTTP/1.0 [CRLF]
Content-Length: 0 [CRLF]
[CRLF]
```

(4) HTTP-formatted Authentication Module alive check response
When the request above is received, the response below is sent when the Authentication Module is up.

```
HTTP/1.0 200 OK [CRLF]
Content-length: 0 [CRLF]
[CRLF]
```

* If the Authentication Module is down, no response is returned.

4.20.12Connection target Authentication Module assignment function 10.0

From version 10.0, in communications between Forwarder and the Authentication Module, users from Forwarder can be assigned and connected to multiple Authentication Modules (load balancing).

(1) Assignment method when not logged in

When there is no session ID such as when a user is not logged in, load balancing is performed using a random number or the user ID as a key.
For the identification key when not logged in, the assignment method is configured by random number or user ID in the CERTLB_TYPE parameter. If this parameter is set to random number, the Authentication Module to which a user is assigned is random. If this parameter is set to user ID, the Authentication Module to which a user is assigned is always same if the user ID is the same.

Some load balancing system products cannot perform this change, so configure this function to make it possible to use these products.

Forwarder configuration file (dfw.conf)

```
CERTLB_TYPE=1
```

(2) Assignment method during login

Load balancing for the Authentication Modules is performed with the session ID as a key.
Configure the CERTLB parameter with the Authentication Module group to be connected to. The Authentication Module identification key configured for the CERTLB parameter uses the value configured on the CERTUNIQUEKEY parameter in the Authentication Module configuration file (cert.conf).

In the configuration example below, the Authentication Modules configured for failover are assigned/connected in two groups.

Group 1 Authentication Module configuration file (cert.conf)

```
CERTUNIQUEKEY=a
```

Group 2 Authentication Module configuration file (cert.conf)

```
CERTUNIQUEKEY=b
```

Forwarder configuration file (dfw.conf)

```
CERTLB=a=www.iwcert1.com:14142
CERTLB=b=www.iwcert2.com:14143
```

4.20.13Retrieving a running Authentication Module's configuration information 10.0

Retrieve the configuration information for a running Authentication Module with the configuration file information output command (dcump-cert).
The actual configuration information of the currently running Authentication Module is output to a file. You cannot check user information in the cache with the configuration file information output command.

4.20.14Linking IceWall SSO group and Active Directory group attributes 10.0

You can use Active Directory group attributes in the MSAD version of the Authentication Module by linking the Active Directory group attributes in the manner below.

(1) Retrieve Active Directory group information at Authentication Module startup.
* The interval to retrieve group information can be arbitrarily set.

(2) Store the Active Directory group information as user information in the cache at login.

(3) Set the Active Directory group information stored in the cache as user information to an IceWall SSO group.

(4) Authenticate access privileges with the IceWall SSO group information created based on the Active Directory group information.

The configuration parameters below are required when using Active Directory group attributes. For details, see the reference manual.
LDAPMULTIVAL, ADGROUP, ADGROUPDN, ADGROUPINTERVAL, ADGROUPPRINAME, ADGROUPPREFIX

5 High Availability

Service with higher availability can be provided by reducing the downtime required when failures occur. A replicated architecture is one of the ways to reduce downtime. When using IceWall SSO, it is necessary to consider preparing redundant IceWall servers, Authentication Servers, Backend Web Servers, and a redundant Authentication DB. Contact a Hewlett-Packard sales representative for details.

5.1 IceWall server redundancy

Redundant and load-balancing systems can be realized by implementing multiple IceWall servers and by using the Load Balance System product.

5.2 Authentication Server and Backend Web Server redundancy

IceWall servers are connected to the Authentication Module (certd) and Backend Web Servers. By implementing the failover option, a standby system can be made for each of these systems. If a communication failure occurs between IceWall server and another main system, this option automatically switches the destinations of connections from IceWall server to standby systems based on the configured settings. (To use this failover function, you must purchase the failover option for IceWall SSO separately.)

Furthermore, whether the Authentication Module is for the master or replica must be clearly defined using the CERTREPTYPE parameter of the Authentication Module configuration file. Because replication is performed between the master and replica even if requests from Forwarder or other sources are distributed to both, load balancing can not be precisely planned.

Furthermore, it is possible to specify failover execution for send, receive and no-data errors occurring during communication between Forwarder and the Backend Web Server. This function is configured as follows:

Host configuration file (sample.conf)

Example: to specify send errors to result in failovers, receive errors to result in fatal errors, and received size 0 to result in fatal errors during communication with the backend web server:

```
FO_SEND=1
FO_RECV=0
FO_NODATA=0
```

For more information on the failover function configuration parameters, see “2.2.6 System tuning configuration parameters” and “3.1.9 Replication parameters” of the “IceWall SSO Reference Manual.”

5.3 Authentication DB redundancy

The Authentication DB can be made redundant by using the replication function of the database itself. For details concerning database replication, see the product manual of the respective database.

5.4 Failover recovery function 10.0

If the master Authentication Module goes down during communication with Forwarder and is then recovered after a failover, the connections between Forwarder and the Authentication Module can be automatically restored to the same sequence as before the failover.
This feature is valid only when ICP 2.0 is used.

If the failover function for the Authentication Module is in use and Forwarder detects that the master Authentication Module is down, Forwarder will overwrite the Authentication Module information (CERT and CERTLB parameters) in the Forwarder configuration file (dfw.conf). After the master Authentication Module is recovered, dfw.conf will be automatically rewritten to its state before failover (called "failback")
To do failback, you must configure the Forwarder configuration file (dfw.conf) and the Authentication Module configuration files (cert.conf) accordingly.

5.4.1 Enabling the failback function

To use the automatic failback function, you must enable the failback parameter (FAILBACK) of the replica Authentication Module. The replica Authentication Module will monitor the master Authentication Module. If the master is running, the replica adds that information to the ICP 2.0 response.

Master Authentication Module configuration file (cert.conf)

```
CERTREPTYPE=0
FAILBACK=0
```

Replica Authentication Module configuration file (cert.conf)

```
CERTREPTYPE=1
FAILBACK=1
```

5.4.2 Enabling the failback function (Forwarder)

If Forwarder is notified that the master has been recovered from the replica Authentication Module, Forwarder makes the configuration to restore the Authentication Module connections to their original sequence.

The CERTFAILBACK parameter is provided for the CERT parameter and the CERTFAILBACKB parameter is provided for the CERTLB parameter used in load balancing.
This failback function is configured as follows:

(1) CERT parameter is used:
If failover is configured and Forwarder uses the CERT parameter, configure the failback by using CERTFAILBACK.

Forwarder configuration file (dfw.conf)

```
ICP_VERSION=2.0
CERT=iwcertdsvr1.com:14142,iwcertsvr2.com:14142
CERTFAILBACK=iwcertdsvr1.com:14142,iwcertsvr2.com:14142
```

(2) CERTLB parameter is used:
If failover is configured and Forwarder uses the CERTLB parameter, configure failback by using CERTLBFAILBACK.

Forwarder configuration file (dfw.conf)

```
ICP_VERSION=2.0
CERTLB=a=iwcertdsvr1.com:14142,iwcertsvr2.com:14142
CERTLB=b=iwcertdsvr3.com:14142,iwcertsvr4.com:14142
CERTLBFAILBACK=a=iwcertdsvr1.com:14142,iwcertsvr2.com:14142
CERTLBFAILBACK=b=iwcertdsvr3.com:14142,iwcertsvr4.com:14142
```

5.5 Authentication Module non-stop maintenance function 10.0

Under a replication architecture, the Authentication Module configuration can be changed and its component libraries can be updated while stopping the master and replica in turn, and not both at the same time.
With the non-stop maintenance function, the Authentication Module is started by running the startup command (start-cert) with the specified configuration file and only user information is downloaded from the replication target.

To use the non-stop maintenance function of an Authentication Module, you must configure to not to download the configuration information from the replicant target.

Authentication Module configuration file (cert.conf)

```
DOWNLOADCONFFLG=0
```

* The DOWNLOADCONFFLG parameter is always excluded from download and reload.

5.5.1 Changing a configuration that uses the Authentication Module's non-stop maintenance function

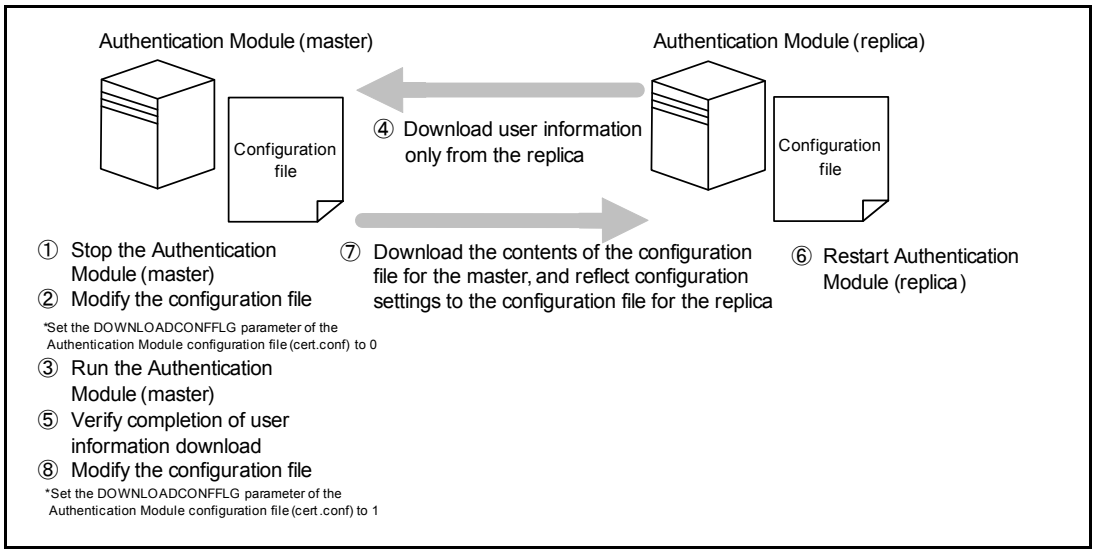
Here is how to reflect configuration settings using this function.

(1) If the configuration parameter to change is included in downloads:

1. Stop the master Authentication Module.
2. In the master configuration file, change the value of the parameter you wish to change.
   (At the same time, set DOWNLOADCONFFLG to 0)
3. Start the master Authentication Module.
4. Only the user information will be downloaded from the replica.
5. In the error log for the master, check for the message below, which states that the download of user information from the replica is complete.

Warning: CS_DownloadUsrCache Download End.

6. Restart the replica Authentication Module.
7. The master configuration information will be downloaded and reflected in the replica configuration.
8. After the configuration change is complete, return the master DOWNLOADCONFFLG to 1.
   (No restart is required.)



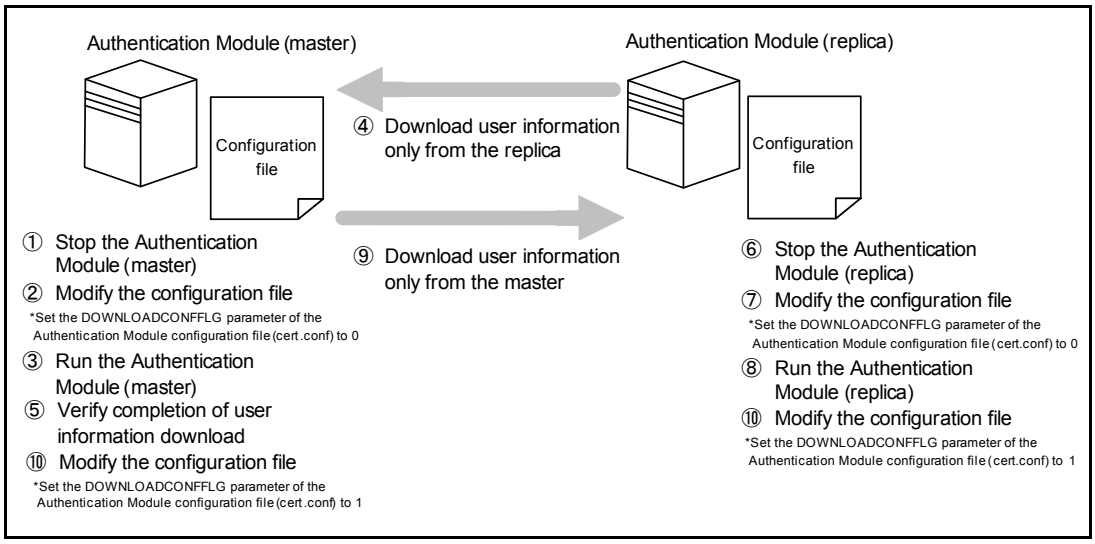
(2) If the configuration parameter to change is excluded from download:

1. Stop the master Authentication Module.
2. Change the applicable configuration settings for the master.
   (At the same time, set DOWNLOADCONFFLG to 0)
3. Start the master Authentication Module.
4. Only the user information will be downloaded from the replica.

5. In the error log for the master, check for the message below, which states that the download of user information from the replica is complete.

Warning: CS_DownloadUsrCache Download End.

6. Stop the replica Authentication Module.
7. Change the applicable configuration settings for the replica.
   (At the same time, set DOWNLOADCONFFLG to 0)
8. Start the replica Authentication Module.
9. Only the user information will be downloaded from the master.
10. After the configuration changes are complete, return the DOWNLOADCONFFLG to 1 for both the master and the replica.
    (No restart is required.)



5.5.2 Updating a library by using the non-stop maintenance function for the Authentication Module.

Here is how to update a library using this function:

1. Stop the master Authentication Module.
2. Update the master library.
   (At the same time, set DOWNLOADCONFFLG to 0)
3. Start the master Authentication Module.
4. Only the user information will be downloaded from the replica.
5. In the error log for the master, check for the message below, which states that the download of user information from the replica is complete.

Warning: CS_DownloadUsrCache Download End.

6. Stop the replica Authentication Module.
7. Update the replica library.
8. Start the replica Authentication Module.
9. Only the user information will be downloaded from the master.
10. After the library is updated, return the DOWNLOADCONFFLG to 1.
    (No restart is required.)

5.5.3 Remarks on changing a configuration by using the Authentication Module's non-stop maintenance function

Note the following items when using this function.

- Keeping configuration changes for the master only or replica only means that their cache content and behavior will be different because their configurations are different. (Except for certain configuration parameters, keeping behavior always different between the master and replica configurations is not supported.)

- Because the master and replica configurations will differ temporarily while you use this function, we recommend that you stop both the master and replica in environments where it is possible, and then change the configuration.

- Certain configuration parameters of the Authentication Module configuration file (cert.conf) can not be changed, even by using this function.

- Note the following points when changing a configuration:

Whether the parameter configuration can be changed
○ Configuration may be changed
△ Configuration may be changed but the cache must not be allowed to change.
○ Configuration may not be changed

Authentication Module configuration file (cert.conf)

Parameter name	Included in download	Change allowed	Reason
ALEVEL	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, their log messages have differences. •Configuration changes may be made through the reload command, but the master and replica will not be synchronized.
ELEVEL	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, their log messages will have differences. •Configuration changes may be made through the reload command, but the master and replica will not be synchronized.
ACCESS	×	○	•May be configured differently for master and replica.
ERROR	×	○	•May be configured differently for master and replica.

Parameter name	Included in download	Change allowed	Reason
TRACE	×	○	•May be configured differently for master and replica.
CATALOG	×	○	•May be configured differently for master and replica.
LOGINFO	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, their log messages will have differences. •Configuration changes may be made through the reload command, but the master and replica will not be synchronized.
LOGPERF	×	○	•May be configured differently for master and replica. •Configuration changes may be made through the reload command, but the master and replica will not be synchronized.
TRANSID	○	○	•May be configured differently for master and replica. •Configuration changes may be made through the reload command. •Note that the log output formats will differ when the master and replica have different configurations.
PERFORMANCE	×	○	•May be configured differently for master and replica.
INFORMATION	×	○	•May be configured differently for master and replica.
LOGMULTITHREAD	×	○	•May be configured differently for master and replica.
GROUP	×	○	•May be configured differently for master and replica. •Group information acquired at login may be different before and after a change. For this reason, the file contents specified by this configuration parameter must be configured to the same values in the master and replica.

Parameter name	Included in download	Change allowed	Reason
ACL	×	○	•May be configured differently for master and replica. •Access control information may be different before and after changes are made. For this reason, the file contents specified by this configuration parameter must be configured to the same values in the master and replica.
ACLREQUEST	×	○	•May be configured differently for master and replica. •Request control information may be different before and after changes are made. For this reason, the file contents specified by this configuration parameter must be configured to the same values in the master and replica.
ADGROUP	×	△	•May be configured differently for master and replica. •If master and replica are configured differently, their behavior concerning whether Active Directory group information is included in user information will be different. We recommend that you configure this parameter the same for master and replica. •Depending on the Active Directory group information, authorization behavior will also be different if you divide groups by Authentication Module and apply access control to these groups.
ADGROUPDN	×	△	•May be configured differently for master and replica. •Must be configured so that the Active Directory group information acquired by the master and replica is the same. •If the group information acquired is different before and after a change, the group information acquired at login will also be different. We recommend that you configure this parameter the same for master and replica.

Parameter name	Included in download	Change allowed	Reason
ADGROUPINTERVAL	×	△	•May be configured differently for master and replica. •The interval must be set so that the Active Directory group information acquired by the master and replica is the same. •If the group information acquired is different before and after a change, the group information acquired at login will also be different. We recommend that you configure this parameter the same for master and replica.
ADGROUPPRINAME	×	△	•May be configured differently for master and replica. •Must be configured so that the Active Directory group information acquired by the master and replica is the same. •If the group information acquired is different before and after a change, the group information acquired at login will also be different. We recommend that you configure this parameter the same for master and replica.
ADGROUPPREFIX	×	△	•The same values must be configured for master and replica. •If the master and replica are configured differently, the group names stored in user information will be different. We recommend that you configure this parameter the same for master and replica.
ADGROUPMAXMEMBER	×	△	•May be configured differently for master and replica. •Must be configured so that the Active Directory group information acquired by the master and replica is the same. •If the group information acquired is different before and after a change, the group information acquired at login will also be different. We recommend that you configure this parameter the same for master and replica.

Parameter name	Included in download	Change allowed	Reason
PORT	×	×	•Can not be changed since user information of logged-in users can not be downloaded.
IPV6LISTEN	×	△	•May be configured differently for master and replica. •Must match the IP version of the requester's CERT parameter. •Must match the CERT parameter of the associated replication target's Authentication Module. •If the master and replication configurations are different, the IP version that received will also be different. We recommend that you configure this parameter the same for master and replica.
HTTPPORT	×	○	•May be configured differently for master and replica. •When this configurations is changed, the request target of the requesting Authentication Module must also be changed.
HTTPECHOHEADER	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, then the ICP 2.0 response in HTTP format will be different between master and replica. We recommend that you configure this parameter the same for master and replica. •Configuration changes may be made through the reload command.
COOKIETIME	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, the cookies of their logged-in users will have different expiration times.
COOKIEEXP	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, the cookies of their logged-in users will have different expiration times.

Parameter name	Included in download	Change allowed	Reason
COOKIERETRY	○	△	•May be configured differently for master and replica. •If master and replica are configured differently, the number of retries by the routine that generates session IDs will be different. We recommend that you configure this parameter the same for master and replica.
LOMETHOD	×	△	•The same values must be configured for master and replica. •The method used to calculate the login expiration time of logged-in users will be different.
DUPLOGIN	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, support for duplicate logins will be different between master and replica.
DUPKIND	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, their login policies will be different. We recommend that you configure this parameter the same for master and replica.
PARALOGIN	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, their login policies will be different. We recommend that you configure this parameter the same for master and replica.
ACCCTRLFLG	○	△	•May be configured differently for master and replica. •If master and replica are configured differently, the security levels of access control when using a client certificate will be different. We recommend that you configure this parameter the same for master and replica.
CERTUNIQUEKEY	○	×	•Can not be changed because the Authentication Module identification keys in the session IDs generated by master and replica will be different.

Parameter name	Included in download	Change allowed	Reason
SESSIONIDLEN	×	×	•Can not be changed because the lengths of the session ID keys generated by master and replica will be different.
PWDLOGINHASH	○	△	•May be configured differently for master and replica. •Must match with the values in the password column of the Authentication DB.
PWDCHGHASH	○	△	•May be configured differently for master and replica. •Must match with the values in the password column of the Authentication DB.
PWDMINLEN	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, their password policies will be different. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command.
PWDMAXLEN	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, their password policies will be different. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command.
PWDALPHANUM	○	△	•May be configured differently for master and replica. •If master and replica are configured differently, their password policies will be different. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command.

Parameter name	Included in download	Change allowed	Reason
PWDEXPIRE	○	△	•May be configured differently for master and replica. •If master and replica are configured differently, their password policies will be different. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command.
PWDSAMEPASS	○	△	•May be configured differently for master and replica. •If master and replica are configured differently, their password policies will be different. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command.
LOCKCOUNT	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, their login policies will be different. We recommend that you configure the same values for master and replica.
PWDEXPCHK	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, their password policies will be different. We recommend that you configure the same values for master and replica.
PWDHISCHK	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, their password policies will be different. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command.

Parameter name	Included in download	Change allowed	Reason
PWDHISCNT	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, their password policies will be different. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command.
PWDFORBID	×	○	•May be configured differently for master and replica. •Forbidden password string information may be different before and after the change. For this reason, the file content specified by this configuration parameter must be configured to the same values in the master and replica.
PWDEXPWARN	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, their password policies will be different. We recommend that you configure the same values for master and replica.
DBHOST	×	△	•May be configured differently for master and replica. •Must be configured so that the user information acquired by the master and replica is the same. •May be configured if the same table information and data will be specified before and after a change. •If the information acquired is different before and after a change, the user information acquired at login will also be different.

Parameter name	Included in download	Change allowed	Reason
DBUID	×	△	•May be configured differently for master and replica. •Must be configured so that the user information acquired by the master and replica is the same. •Changes may be made if the same table information and data are acquired before and after a change •If the information acquired is different before and after a change, the user information acquired at login will also be different.
DBPWD	×	△	•May be configured differently for master and replica. •Must be configured so that the user information acquired by the master and replica is the same. •Changes may be made if the same table information and data are acquired before and after a change •If the information acquired is different before and after a change, the user information acquired at login will also be different.
DBTBL	×	△	•May be configured differently for master and replica. •Must be configured so that the user information acquired by the master and replica is the same. •Changes may be made if the same table information and data are acquired before and after a change •If the information acquired is different before and after a change, the user information acquired at login will also be different.
DBATTR	×	○	•May be configured differently for master and replica. •The Authentication DB column information may be different before and after a change. For this reason, the file content specified by this configuration parameter must be configured with the same values in the master and replica.

Parameter name	Included in download	Change allowed	Reason
DBEXATTR	×	△	•May be configured differently for master and replica. •Must be configured so that the user information acquired by the master and replica is the same. •Changes may be made if the same table information and data are acquired before and after a change •If the information acquired is different before and after a change, the user information acquired at login will also be different.
LDAPBIND	○	△	•May be configured differently for master and replica. •If master and replica are configured differently, their behavior for logins will be different. We recommend that you configure the same values for master and replica.
LDAPPCHG	○	△	•May be configured differently for master and replica. •If master and replica are configured differently, their behavior for password changes will be different. We recommend that you configure the same values for master and replica.
LDAPLANG	○	△	•May be configured differently for master and replica. •If master and replica are configured differently, the character encodings of the LDAP data will be different. We recommend that you configure the same values for master and replica.
LDAPSSL	×	△	•May be configured differently for master and replica. •If master and replica are configured differently, they will have different behavior (i.e., in the case MSAD edition is used, the password can not be changed.) We recommend that you configure the same values for master and replica.

Parameter name	Included in download	Change allowed	Reason
LDAPCACERT	×	△	•May be configured differently for master and replica. •If master and replica are configured differently, the CA certificates in use will be different. We recommend that you configure the same values for master and replica. •Note the related LDAPVERIFYSCRCERT configuration.
LDAPVERIFYSVRCERT	×	△	•May be configured differently for master and replica. •If master and replica are configured differently, they will have different behavior concerning whether to verify CA certificates in use. We recommend that you configure the same values for master and replica.
LDAPCIPHERSUITE	×	△	•May be configured differently for master and replica. •If master and replica are configured differently, the encryption strength of SSL connections will be different. We recommend that you configure the same values for master and replica.
LDAPSSLBIND	×	△	•May be configured differently for master and replica. •If master and replica are configured differently, whether SSL communication is established during BIND authentication will be different. We recommend that you configure the same values for master and replica.
ADPCHG	×	△	•May be configured differently for master and replica. •If master and replica are configured differently, their behavior will be different concerning whether a password change is allowed. We recommend that you configure the same values for master and replica.

Parameter name	Included in download	Change allowed	Reason
LDAPMULTIVAL	×	△	•May be configured differently for master and replica. •If master and replica are configured differently, the user information acquired at login will be different. We recommend that you configure the same values for master and replica.
LDAPREFERRAL	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, their behavior for the LDAP referral function will be different. We recommend that you configure the same values for master and replica.
ADDGFWBIND	×	△	•The same values must be configured for master and replica. •If master and replica are configured differently, their FEDERATION authentication behavior for BIND authentication will be different.
DBCRYPTOTYPE	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, the user information acquired at login will be different between master and replica.
DBIWCRYPTOSEED	○	△	•The same values must be configured for master and replica. •Data stored in the Authentication DB must be encrypted by the same seed.
DBCRYPTOATTR	○	△	•The same values must be configured for master and replica. •The column name configured in this parameter must be specified in DBEXATTR. •If master and replica are configured differently, the user information acquired at login will be different between master and replica.
LOGDBTBL	×	△	•May be configured differently for master and replica. •Note the related values configured in logdbattr.conf and in LOGDBSEQNAME.

Parameter name	Included in download	Change allowed	Reason
LOGDBATTR	×	○	•May be configured differently for master and replica. •The log column information may be different before and after a change. For this reason, the file content specified by this configuration parameter must be configured with the same values in the master and replica.
LOGDBSEQNAME	×	△	•May be configured differently for master and replica. •Note the related values configured for logdbattr.conf and LOGDBSEQNAME.
REFTBL	×	△	•May be configured differently for master and replica. •Must be configured so that the table information acquired by the master and replica is the same. •Changes may be made if the same table information and data are acquired before and after a change •If the information acquired is different before and after a change, the table information acquired at login will also be different.
REFATTR	×	△	•May be configured differently for master and replica. •Must be configured so that the table information acquired by the master and replica is the same. •Changes may be made if the same table information and data are acquired before and after a change •If the information acquired is different before and after a change, the table information acquired at login will also be different.

Parameter name	Included in download	Change allowed	Reason
REFUID	○	△	•May be configured differently for master and replica. •Must be configured so that the table information acquired by the master and replica is the same. •Changes may be made if the same table information and data are acquired before and after a change •If the information acquired is different before and after a change, the table information acquired at login will also be different.
MAXREQTHREAD	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, then the number of requests that they can handle will be different. We recommend that you configure the same values for master and replica.
ACCTHREAD	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, then the number of requests that they can handle will be different. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command, but they will not be synchronized.
REQQUESIZE	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, then the number of requests that they can handle will be different. In order for replication between master and replica to be processed by the sending replication thread and the receiving request thread, we recommend that the master and replica be configured with the same values.

Parameter name	Included in download	Change allowed	Reason
MAXDBCONNECT	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, the number of simultaneous connections to the Authentication DB will be different. We recommend that you configure the same values for master and replica.
LOGBUFSIZE	×	○	•May be configured differently for master and replica.
CACHE	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, replication may not be performed after login, if there are more logins than the smaller CACHE value.
MAXLOGINUSER	○	△	•The same values must be configured for master and replica. •Configuration changes can also be made through the reload command. •If master and replica are configured differently, the allowable number of user logins will be different between master and replica.
RECVWAITTIME	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, the waiting time for receiving requests will be different between master and replica. We recommend that you configure the same values for master and replica.
THREADSTACKSIZE	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, their thread-related behavior will be different. We recommend that you configure the same values for master and replica.
CERT	×	×	•Can not be changed because user information of logged-in users can not be downloaded.
CERTREPTYPE	×	×	•The roles of master and replica can not be changed.

Parameter name	Included in download	Change allowed	Reason
RETRYCNTC	×	○	•May be configured differently for master and replica. •If the master and replica are configured differently, the number of connection attempts for replication between Authentication Modules will be different. We recommend that you configure the same values for master and replica.
RETRYTMC	×	○	•May be configured differently for master and replica. •If the master and replica are configured differently, then the number or connection attempts for replication between Authentication Modules will be different. We recommend that you configure the same values for master and replica.
LIVETIMER	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, then the retry interval after a replication failure between master and replica will be different. We recommend that you configure the same values for master and replica.
HEALTHTIMER	×	○	•May be configured differently for master and replica. •Valid for the replica only. The behavior will not change even when the master configuration is changed. •Note the related HEALTHCNT configuration value.
HEALTHCNT	×	○	•May be configured differently for master and replica. •This configuration parameter is valid for the replica only. The behavior will not change even when the master configuration is changed. •Note the related HEALTHTIMER configuration value.
DOWNLOADCONFFLG	×	○	Configuration parameters for using this function.

Parameter name	Included in download	Change allowed	Reason
FAILBACK	×	○	•May be configured differently for master and replica. •This configuration parameter is valid for the replica only. The behavior will not change even when the master configuration is changed. •The configuration can also be changed by synchronizing with the replica.
MAXREPTHREAD	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, then the number of requests that they can handle will be different. We recommend that you configure the same values for master and replica.
REPQUESIZE	×	○	•May be configured differently for master and replica. •If master and replica are configured differently, then the number of requests that they can handle will be different. In order for replication between the master and replica to be processed by both the sending replication thread and the receiving request thread, we recommend that the master and replica be configured with the same values.

Group configuration file (cert.grp)

Parameter name	Included in download	Change allowed	Reason
Group configuration file (cert.grp)	○	△	•May be configured differently for master and replica. •Note that a group to which a user belongs may be changed before and after a change. (The group to which an already logged in user belongs to will not change.) •Configuration changes can also be made through the reload command.

Access control file (cert.acl)

Parameter name	Included in download	Change allowed	Reason
Access control file (cert.acl)	○	△	•May be configured differently for master and replica. •Pay attention to the consistency, before and after the configuration change, between this file and the group configuration file. •Configuration changes can also be made through the reload command.

Authentication DB column information file (dbattr.conf)

Parameter name	Included in download	Change allowed	Reason
Authentication DB column information file (dbattr.conf)	○	△	•The same values must be configured for master and replica. •If master and replica are configured differently, the user information acquired at login will be different between master and replica. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command.

Log column information file (logdbattr.conf)

Parameter name	Included in download	Change allowed	Reason
Log column information Log Files (logdbattr.conf)	○	△	•May be configured differently for master and replica. •If master and replica are configured differently, then data will be added to each module's specified tables and columns. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command.

Forbidden password configuration file (pwdforbid.conf)

Parameter name	Included in download	Change allowed	Reason
Forbidden password string configuration file (pwdforbid.conf)	○	△	•May be configured differently for master and replica. •If master and replica are configured differently, then password checking will be performed with each forbidden password string condition specified. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command.

Request control configuration file (request.acl)

Parameter name	Included in download	Change allowed	Reason
Request control configuration file (request.acl)	○	△	•May be configured differently for master and replica. •If master and replica are configured differently, then request checking will be performed with each request control configuration file condition specified. We recommend that you configure the same values for master and replica. •Configuration changes can also be made through the reload command.

5.6 Transaction IDs 10.0

From version 10.0, transaction ID numbering can be applied to each request. Outputting the common ID of the same transaction to an assigned log and the screen allows you to troubleshoot more quickly.

5.6.1 How to make the configuration

To enable the use of the transaction ID, these configuration settings are required:

Forwarder configuration file (dfw.conf)

```
TRANSID=1
```

Authentication Module configuration file (cert.conf)

```
TRANSID=1
```

The string can be configured to identify which Forwarder has generated the transaction ID in a multi-machine Forwarder architecture.

To configure a string which identifies Forwarder included in a transaction ID to iwdfw01:

```
TRANSID=1
TRANSID_STR=iwdfw01
```

* ICP 2.0 must be used to send the transaction ID to the Authentication Module. However, transaction IDs will be generated regardless of ICP version.

5.6.2 Logs

When transaction IDs have been enabled, the transaction ID issued will be added to all log files.

5.6.3 Headers

The transaction ID can be sent to Backend Web Servers as an HTTP header.
To send a transaction ID to a Backend Web Server, these configuration settings are required:

Forwarder configuration file (dfw.conf)

```
HEADER_NAME_TID=[header name]
```

In this example, “Icewall-transactionid” is the name of a header for the transaction ID information sent to a backend web server.

```
HEADER_NAME_TID=Icewall-transactionid
```

5.6.4 Special keywords

A transaction ID can be output to content as a special keyword.
This keyword is “$IWTID”.

6 Security

In order to provide services over the Internet, an “integration of authentication and authorization” function and a “reverse proxy” function are both required. These functions are provided by IceWall SSO.
The following typical measures are included to improve security levels:

- Sniffing prevention
- High-strength session IDs
- Encryption of Authentication DB reference columns
- Session ID encryption
- Spoofing prevention
- Cross site scripting prevention
- Provisions against buffer overflows

6.1 Sniffing prevention

Data encryption is an effective countermeasure for “sniffing” of data during communications.
IceWall SSO can use SSL (Secure Socket Layer) for communications between IceWall server and the Backend Web Servers. (To conduct SSL communications between IceWall SSO and Backend Web Servers, the SSL Option of IceWall SSO must be purchased separately.)

6.2 High-strength session IDs 10.0

If ICP 2.0 is used, IceWall SSO can generate and use 64-byte session IDs, in addition to the existing 32-byte ones. As a result, session IDs will be more difficult to be deciphered.

Authentication Module configuration file (cert.conf)

Example of generating a 64-byte session ID SESSIONIDLEN=64

If IceWall SSO tries to generate a 64-byte session ID without using ICP2.0 communications, a session ID error will occur. Configurations that generate session IDs with different byte lengths between master and replica are not supported.

6.3 Encryption of Authentication DB reference columns 10.0

You can now use pre-encrypted values in reference columns that you included in an Authentication DB in addition to the columns (attributes) that IceWall acquires.
If you store values encrypted by proprietary methods in a reference column, you must develop your own Authentication DB encryption library for decoding column values. For details, see the "Authentication DB Encryption Library Development Manual."

6.4 Session ID encryption

If the session ID that is required for accessing the IceWall SSO is cracked, access to the IceWall SSO will be possible, but since further encryption is performed when Forwarder notifies the client, and the encryption method can be changed for each Forwarder, the scope of the illegal use by the cracker can be limited to a minimum range.

Forwarder configuration file (dfw.conf)

Example: to use "ab12AB-_!" as the key when encrypting and decrypting the session ID:

```
SESSION_ENC_KEY=ab12AB-_!
```

6.5 Spoofing prevention

As standard practice, IceWall SSO uses a user ID and password to verify identity (spoofing prevention). However, client certificates can be used to perform more advanced identification.
Client certificates can be used for identification by implementing IceWall SSO Client Certificates Option. (These certificates are installed on clients in advance.)
This option allows the use of two types of authentication systems: one is authentication with "user ID and password," the other is authentication with "certificate and password."
(To use client certificates, the Client Certificates Option for IceWall SSO must be purchased separately.)

6.6 Cross site scripting prevention

When a malicious script is written to Backend Web Servers through the use of CGI scripts, such as bulletin boards running on the Backend Web Servers, personal information of users, such as cookies, may be stolen. These attacks involving malicious scripts are called "cross-site scripting," and IceWall SSO has a preventative function against them.

6.6.1 Configuring a filter for requests sent to Backend Web Servers

- Use GETFILTER to prevent malicious scripts contained in query strings sent to Backend Web Servers.

The following example configuration will filter SCRIPT tags contained in query strings.

Host configuration file (sample.conf)

```
GETFILTER=SCRIPT
```

- Use POSTFILTER to prevent malicious scripts contained in POST data sent to Backend Web Servers.

The following example configuration will filter SCRIPT tags contained in POST data.

Host configuration file (sample.conf)

```
POSTFILTER=SCRIPT
```

6.6.2 Configuring a filter for content received from Backend Web Servers

- Use HTMLFILTER to prevent attacks when content with a malicious script embedded is displayed.

The following example configuration will filter SCRIPT tags contained in the attribute values of content received from a Backend Web Server.

Host configuration file (sample.conf)

```
HTMLFILTER=SCRIPT
```

- Use SVRFILTER to prevent malicious links to servers.

The following example configuration will filter undefined hosts described in the attribute value of a tag that is subject to URL conversion.

Host configuration file (sample.conf)

```
SVRFILTER=1
```

6.6.3 Setting a filter by using a designated keyword

- Use DFWFILTER to protect against cross site scripting between Backend Web Server content and template pages output independently by IceWall SSO.

The following example configuration will filter special keywords of Backend Web Server content and of a template page output independently by IceWall SSO.

Host configuration file (sample.conf)

```
DFWFILTER=2
```

6.6.4 Filter dry-run function 10.0

- To help you check how filtering is operating, the dry-run function outputs information to a test error log when it finds a string to be filtered without actually filtering. Check all values configured for the filter function.

The following example configuration and output show a dry run test where QueryString in the request sent to a Backend Web Server includes a SCRIPT tag.

Host configuration file (sample.conf)

```
GETFILTER=SCRIPT
GETFILTERERR=2
```

Forwarder error log file

```
Warning: GET Filter TEST. Host:[www.host.com:80] Path:[/index.cgi]
QUERY_STRING:[aaa=<script>] Tag:[<script]
```

6.7 Provisions against buffer overflows

The system also provides a function for protecting web servers from attacks that are intended to cause failure due to buffer overflow, triggered by the transmission of intentionally long URLs sent to Backend Web Servers.

(1) To prevent web sever buffer overflow due to long URLs:

Forwarder configuration file (or host configuration file)

```
MAXURL=512
```

(2) To prevent web server buffer overflow due to long arguments:

Forwarder configuration file (or host configuration file)

```
MAXQUERY=256
```

6.8 Other Topics

An IceWall SSO session ID is included in the URL when IceWall SSO is used with session management using URL-Cookies. For this reason, session IDs can be cracked with the HTTP_REFERER on malicious sites.
A user should log out after operations whenever IceWall SSO is operated with session management using URL-Cookies.

6.9 Improving security

The following measures can be used to improve security:

6.9.1 Common configuration for the user ID error and password ID error pages used at login

The default configuration of IceWall SSO displays different error pages for mistyped user IDs and mistyped passwords. This configuration allows for repeated typing of the user ID; therefore, it is possible to search for a user ID that is accepted for login. If the password for that user is found, the attacker can log in.
To prevent this, configure the login user ID error and password error pages to be the same, so the user does not know which error occurred.

6.9.2 Prohibiting substitution of special keywords with user information

The default configuration of IceWall SSO causes special keywords included in the content to be replaced by the user information of the logged-in user. In this case, if a malicious user includes special keywords in the arguments of the request, unintentional substitution can occur and user information may be disclosed.
To prevent this, it is recommended to set all special keywords that are not used in web applications or content to “no conversion.”

(1) To disallow conversion for all special keywords in the content of a Backend Web Server

Host configuration file

```
CTRL_SPKEY=
```

(2) To allow conversion of $DFW, $USER_ID, and $ALIAS only in the content of a Backend Web Server

Host configuration file

```
CTRL_SPKEY=$DFW,$USER_ID,$ALIAS
```

6.9.3 Changing templates

The default configuration of IceWall SSO displays the user ID of the logged-in user on the password change page. This is because the password change page template includes a special keyword that is replaced by the user ID. Since it is possible for a login-capable user ID to be revealed from this page, it is recommended to customize this template so it does not display the user ID.

7 Performance Tuning

This section introduces methods for improving IceWall SSO performance and preventing performance degradation. In order to improve performance, the configuration settings of Forwarder and the Authentication Module need to be optimized. All configuration values used in this chapter are initial values. For details on individual parameters, see the “IceWall SSO Reference Manual.”

7.1 Forwarder performance

Performance of Forwarder can be adjusted by changing the parameters related to content transmission and to connections to the Backend Web Servers.

7.1.1 Frequent transactions exceeding 64 KB

Adjust the content buffer size.
Configure the ALLOC parameter in the Forwarder configuration file.

Forwarder configuration file (dfw.conf)

```
ALLOC=65536
```

* If the size is set too large, performance will deteriorate.
* The configuration setting is common to all Backend Web Servers.

7.1.2 If the non-text-based (binary) content in the browsing content is too large

Configure the forwarding method setting for non-HTML content to non-buffering mode.
The configuration settings are conducted for each host configuration.

Host configuration file (sample.conf)

```
BUFFER=0
```

* Applicable content types are those that are not set in the CTYPE parameter of the host configuration file.

7.1.3 Keep-Alive between Forwarder and a Backend Web Server

Disable the Keep-Alive configuration of the Backend Web Server accessed by IceWall SSO, or set the KeepAliveTimeout configuration to 0. This is done because IceWall SSO response time may become quite long.
However, if the Backend Web Server is a Microsoft Internet Information Services server, we recommend that you enable the Keep-Alive configuration for the web server instance. If Keep-Alive is not enabled, Forwarder will not be able to judge that the browser has received all content from the server, even if the content is displayed.

For Backend Web Servers for which the Keep-Alive configuration cannot be changed, perform configurations according to “7.1.4 Adjusting the Forwarder disconnection waiting time.”

7.1.4 Adjusting the Forwarder disconnection waiting time

This parameter sets the time to wait (in seconds) until a connection is disconnected by the IceWall server when it is not disconnected by a Backend Web Server.
Configure the timeout setting for each host configuration file.

Host configuration file (sample.conf)

```
CLOSETIME=3
```

This setting is only valid when a Content-length header is not sent from the Backend Web Server.

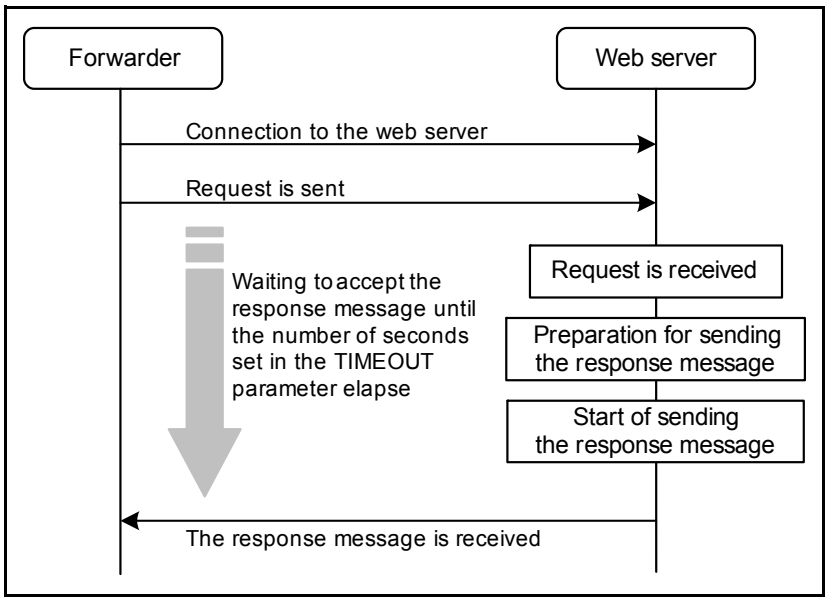
7.1.5 Adjusting response timeout between Forwarder and the Backend Web Server

This parameter sets a timeout value (in seconds) when waiting to accept a response message after connection to a Backend Web Server.
Configure the timeout setting for each configuration file.

Host configuration file (sample.conf)

```
TIMEOUT=180
```

* Please note that this setting does not specify the timeout for connecting to the web server.



7.1.6 Adjusting the connection timeout between Forwarder and the Backend Web Server (10.0)

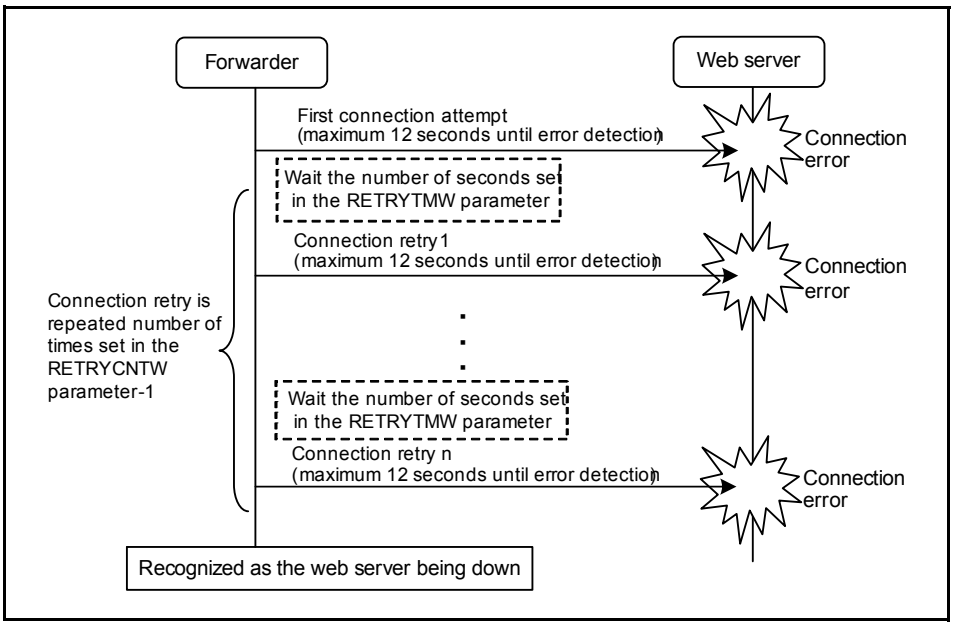
The time to detect that the Backend Web Server is down can be set by adjusting the retry count and the retry interval (in seconds).
Configure the timeout setting for each configuration file.

Host configuration file (sample.conf)

```
RETRYCNTW=10
RETRYTMW=3
```

The timeout value for detecting a connection error to a Backend Web Server is a maximum of 12 seconds if the Backend Web Server is down and therefore not on the network; if the Backend Web Server is running but the port is down, there is practically no waiting time.
Time until the backend server is detected as down:
[(RETRYCNTW) × [connection error detect timeout] {max 12 sec} + ((RETRYCNTW-1) × RETRYTMW)]

Example: RETRYCNTW=5
RETRYTMW=2
Time to detect the Backend Web Server is down: .
(5 × 12) + ((5 - 1) × 2) = 68 seconds



7.1.7 Adjusting connection timeout between Forwarder and the Authentication Module (10.0)

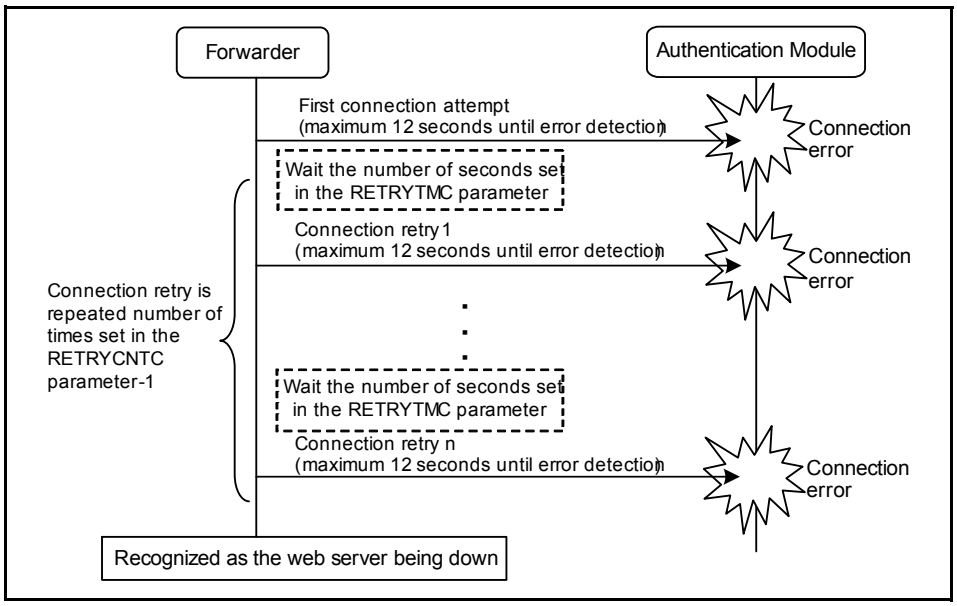
The time to detect that the Authentication Module is down can be set by adjusting the retry count and the retry interval (in seconds).
Configure the timeout setting in the Forwarder configuration file.

Forwarder configuration file (dfw.conf)

```
RETRYCNTC=10
RETRYTMC=3
```

The timeout value for detecting a connection error to the Authentication Module is a maximum of 12 seconds if the Authentication Server is down and therefore not on the network; if the server is running but the Authentication Module is down, there is practically no waiting time.
Time until the Authentication Module is detected as down:
[(RETRYCNTC) × [connection error detect timeout] {max 12 sec} + ((RETRYCNTC-1) × RETRYTMC)]

Example: RETRYCNTC=10
RETRYTMC=3
Time to detect the Authentication Module is down:
(10 × 12) + ((10 - 1) × 3) = 147 seconds



7.1.8 Adjusting response timeout between Forwarder and the Authentication Module

This parameter sets a timeout value (in seconds) for receiving a response message after connecting to the Authentication Module.

Configure the timeout setting in the Forwarder configuration file.

Forwarder configuration file (dfw.conf)

```
CERT_TOUT=600
```

* Please note that this does not specify the timeout for connecting to the Authentication Module.

7.2 Authentication Module performance

When request processes for the Authentication Module begin to pile up, the following phenomena occur:

- System response is slow when a user logs in.
- A time lag occurs in displaying page even when static content is being browsed.

If this kind of phenomenon occurs, verify the performance status of the Authentication Module by executing the command to get operating information (info-cert):

```
$ /opt/icewall-sso/certd/bin/info-cert
```

This command should be executed by the iwadmin user.

If the request queue overflow occurs, the info-cert request may be ignored.

When this command is executed, the following Authentication Module performance information is output to an access log file.

Output messages

Message	Description
CERTINFO USER: Used[n1] / Max[n2] = n3% (10.0)	n1: Number of current logged-in users n2: Allowable number of user logins n3: Ratio of logged-in users to allowable number of user logins
CERTINFO CACHE: Used[n1] / Max[n2] = n3% [10.0]	n1: Number of current logged-in users n2: Cache size n3: Cache usage rate
CERTINFO REQUEST_QUEUE : Used[n1] / Max[n2] = n3% Over[n4]	n1: Number of request queues currently in use n2: Request queue size n3: Request queue usage rate n4: Request queue overflow count (The queue overflow count is measured between the previous and current execution of info-cert.)
CERTINFO ACCTHREAD : Used[n1] / Max[n2] = n3%	n1: Number of access threads currently in use n2: Number of request threads n3: Request thread usage rate
CERTINFO REPRICA_QUEUE : Used[n1] / Max[n2] = n3% Over[n4]	n1: Number of replication queues currently in use n2: Replication queue size n3: Replication queue usage rate n4: Replication queue overflow count (The queue overflow count is measured between the previous and current execution of info-cert.)

Message	Description
CERTINFO REQUEST_THREAD : Used[n1] / Max[n2] = n3%	n1: Number of threads currently in use n2: Total number of threads n3: Thread usage rate
CERTINFO REPRICA_THREAD : Used[n1] / Max[n2] = n3%	n1: Number of threads currently in use n2: Total number of threads n3: Thread usage rate
CERTINFO DBCONNECT : Used[n1] / Max[n2] = n3% Over[n4]	n1: Number of connections currently in use n2: Total number of connections n3: Connection usage rate n4: Wait count for an open connection (cumulative total number between the previous and current execution of info-cert.)

Info-cert will output the status of the Authentication Module when the command is run; therefore, you can obtain information about how the Authentication Module operates in the period of time by running the info-cert command at regular intervals such as every 10 minutes.

- Checking request thread status
  Requests against the Authentication Module will be processed by a request thread. If a shortage of request threads occurs, the request process must wait for the request threads to be freed, so performance will deteriorate.
  If a failure occurs in the Authentication DB, the request threads may all become occupied due to requests requiring an Authentication DB connection. As a result, performance may deteriorate to the point that logins can not be handled.

  Related messages
  CERTINFO REQUEST_THREAD : Used[n1] / Max[n2] =n3%

- Checking the status of Authentication DB connections
  User information in the Authentication DB can be acquired or updated when the Authentication Module processes logins, logouts, password changes, and other requests.
  If many logins are attempted or Authentication DB failure occurs, login may take a long time due to the lack of available connections to the DB and many request threads may be occupied due to requests which need to connect to the Authentication DB.

  Related messages
  CERTINFO DBCONNECT : Used[n1] / Max[n2] = n3% Over[n4]

- Checking the request queue status
  After requests against the Authentication Module have entered the request queue, they will be passed to a request thread and processed.
  Requests in the request queue have a "waiting for idle request thread" state.
  If the request queue usage rate is continually high, increase the number of request threads.

  Related messages
  CERTINFO REQUEST_QUEUE : Used[n1] / Max[n2] = n3% Over[n4]

- Checking replication thread status
  If the Authentication Module has a replication architecture, replication threads will send requests to the replication target so that cache and configuration information can be synchronized between master and replica.

  Related messages

CERTINFO REPRICA_THREAD : Used[n1] / Max[n2] = n3%

- Checking replication queue status
  After a replication request between master and replica has entered the replication queue, it will be passed to a replication thread, which will then send the replication request to the replication target.
  If the replication queue usage rate is continually high, increase the number of replication threads.

  Related messages

CERTINFO DBCONNECT : Used[n1] / Max[n2] = n3% Over[n4]

- Checking cache usage
  Check the cache usage.

  Related messages

CERTINFO CACHE: Used[n1]/ Max[n2]= n3%

- Checking the number of logged-in users
  Check the status of current logins for the allowable number of user logins.
  If login requests exceed the allowable number of user logins, a “login limit exceeded” error will be thrown.

  Related messages

CERTINFO USER: Used[n1] / Max[n2] = n3%

- Checking the number of access-specific threads
  Check the usage of requests threads for requests that do not require DB connections.
  Access-specific threads are used when the number of requests equals or exceeds the value of total request threads minus access-specific threads.
  Continual use of access-specific threads could present a situation in which the number of requests exceeds the number of request threads, a shortage of DB connections arises, and DB failures occur.
  For this reason, you should consider measures like checking for DB failures and increasing the number of request threads and DB connections.

  Related messages

CERTINFO ACCTHREAD : Used[n1] / Max[n2] = n3%

The request processing count for the Authentication Module below is configured using performance information.

① Increases the number of threads processed simultaneously.
Authentication Module configuration file (cert.conf)

```
MAXREQTHREAD=10
```

② Increases the number of request threads that do not require a DB connection. 10.0
Authentication Module configuration file (cert.conf)

```
ACCTHREAD=4
```

③ Increases the queue size of spooled requests.
Authentication Module configuration file (cert.conf)

```
REQQUESIZE=20
```

④ Increases the number of simultaneous connections to the Authentication DB.
Authentication Module configuration file (cert.conf)

```
MAXDBCONNECT=2
```

In general, the log buffer size (LOGBUFSIZE) does not need to be changed.

7.2.1 Adjusting the request receive timeout

If the traffic increases between Forwarder or the agent and the Authentication Module, request receive errors may occur. In such cases, the receive error can be prevented by adjusting the request receive timeout. This setting is configured in the Authentication Module configuration file.

Authentication Module configuration file (cert.conf)

```
RECVWAITTIME=5
```

7.2.2 Receive timeout from the Authentication DB

The response timeout between the Authentication Module and the Authentication DB can be adjusted in the LDAP, OpenLDAP, MSAD, and NED editions only. This setting is added to the Authentication Module startup command.
The following example shows the startup command for the Authentication Module in the LDAP edition.

Authentication Module startup command (start-cert)

```
#!/bin/sh

export IW_HOME=/opt/icewall-sso
export SHLIB_PATH=$IW_HOME/lib/certd
export LD_LIBRARY_PATH=$SHLIB_PATH
export IW_DBTIMEOUT=30

$IW_HOME/certd/bin/certd -c $IW_HOME/certd/config/cert.conf
```

* The setting must be added to a line that is preceding the line starting up the Authentication Module.

The unit for setting the waiting time is seconds.
Restart the Authentication Module to enable the new setting. If a replication architecture is used, both master and replica must be configured.

7.2.3 Adjusting the timeout for connections to the Authentication Module 10.0

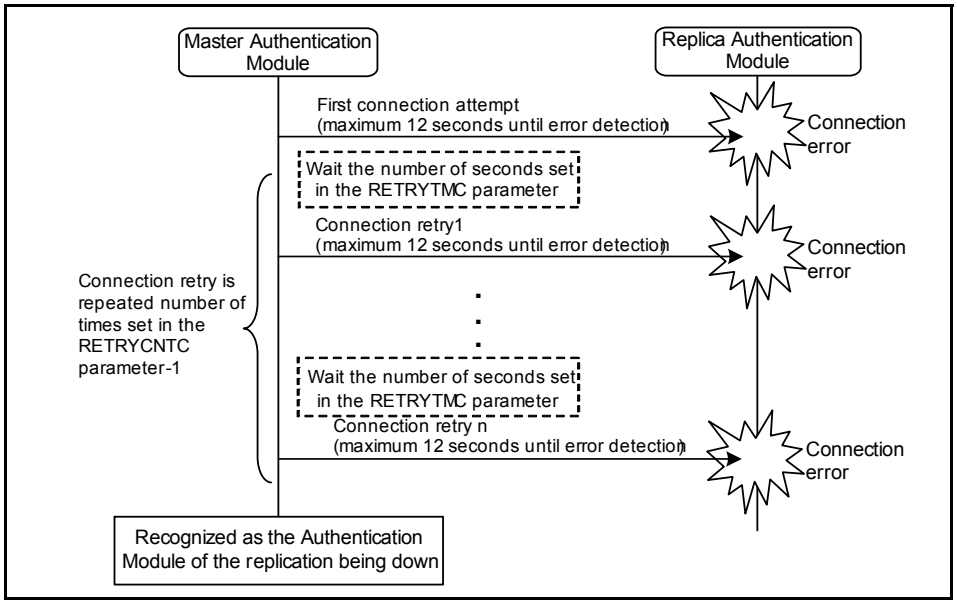
The time to detect that the replicated Authentication Module is down can be set by adjusting the retry count and the retry interval (in seconds).
This timeout setting is configured in the Authentication Module configuration file.

Authentication Module configuration file (cert.conf)

```
RETRYCNTC=10
RETRYTMC=3
```

The timeout for detecting a connection error to the replicated Authentication Module is a maximum of 12 seconds if the replicated Authentication Server is down and therefore not present on the network; if the server is running but the Authentication Module is down, there is practically no waiting time.
Time until Authentication Module is detected as down:
[(RETRYCNTC) × [connection error detect timeout] {max 12 sec} + ((RETRYCNTC-1) × RETRYTMC)]

Example:RETRYCNTC=10
RETRYTMC=3
Time to detect the Authentication Module is down:
(10 × 12) + ((10 - 1) × 3) = 147 seconds



7.3 Preventing Authentication Module performance deterioration

The measures for preventing a deterioration in performance of the Authentication Module are described below:

- Do not configure more than the necessary number of groups.
- Do not configure more than the necessary number of URLs for access control.

7.4 Parameter calculation method

These are calculation methods for parameters required to process anticipated requests against the Authentication Module. Calculations are based on the number of CPUs (including virtual CPUs) installed on the Authentication Server and the anticipated number of simultaneous logins during peak times. For CPUs with two or more cores, count the number of cores as separate CPUs.
It may be necessary to change kernel parameters depending on the calculated parameters of the Authentication Module and the number of simultaneous users. However, if the performance of the Authentication DB is low, or a process is added to the UserExit routine, separate tuning is necessary to achieve the optimal values.

7.4.1 Calculating MAXDBCONNECT

MAXDBCONNECT is calculated as follows:

MAXDBCONNECT =

([expected peak login requests per second] - 45) ÷ 15 + 2

If the calculated value is less than 2, then it is considered to be 2.

7.4.2 Calculating ACCTHREAD

ACCTHREAD is calculated as follows:

ACCTHREAD = ([Number of CPUs] × 2)

The threads in the thread count specified by ACCTHREAD will only be used in requests that do not require an Authentication DB connection.
Note that the request threads in the thread count (MAXREQTHREAD - ACCTHREAD) are used by both the requests that require an Authentication DB connection and those that do not require the connection, instead of only being used by requests that require an Authentication DB connection.
The behavior of requests that require an Authentication DB connection and which exceed MAXDBCONNECTION will be different depending on whether ACCTHREAD is configured.

- ACCTHREAD is not configured
  The request waits for Authentication DB connections to be freed within the number of request threads specified by MAXREQTHREAD. If all request threads are occupied, requests requiring an Authentication DB connection will also go in the request queue.

- If ACCTHREAD is configured to a value of one or greater
  Requests will wait for Authentication DB connections to be freed up to the number of request threads specified by (MAXREQTHREAD - ACCTHREAD). A “DB busy” error will be returned for requests requiring an Authentication DB connection in excess of (MAXREQTHREAD - ACCTHREAD).

7.4.3 Calculating MAXREQTHREAD

MAXDBCONNECT is calculated as follows:

MAXREQTHREAD = MAXDBCONNECT + ([number of CPUs] × 2)
+ ACCTHREAD + [threads waiting for DB connection]

* [threads waiting for DB connection]
  If ACCTHREAD is configured, a “DB busy” error will be returned to requests that require an Authentication DB connection and which exceed (MAXREQTHREAD - ACCTHREAD). So that a “DB busy” error will not be returned for a request that requires an Authentication DB connection that instantaneously exceeds (MAXREQTHREAD-ACCTHREAD), the number of threads waiting for DB connections to be freed in the request thread should be included in MAXREPTHREAD.

7.4.4 Calculating REQQUESIZE

REQQUESIZE is calculated as follows:

REQQUESIZE = (MAXREQTHREAD × 2) + 10

7.4.5 Calculating MAXREPTHREAD

The calculation formula is the following:

MAXREPTHREAD = MAXDBCONNECT + ([number of CPUs] × 2)

However, if MAXREPTHREAD is greater than REQQUESIZE, the queue may overflow, so if the calculated value exceeds REQQUESIZE, then either increase REQQUESIZE, or set MAXREPTHREAD to a smaller value.

7.4.6 Calculating REPQUESIZE

There is no formula for this parameter. Generally, this item is set to 500.
Configure the CACHE parameter of the Authentication Module configuration file (cert.conf) to 1,000 if the number of simultaneous users exceeds 100,000.

7.4.7 Calculating THREADSTACKSIZE

There is no formula for this parameter. Generally, this item is set to 256.

7.4.8 Calculating kernel parameters

Calculate the related kernel parameters for HP-UX by the following formula:

max_thread_proc = MAXREQTHREAD + 10 or more

maxfiles = MAXREQTHRED + MAXDBCONNECT + 10 or more

maxdsiz = ([group list] + [user data] + 640) × CACHE + 2MB or more

Group list = (40 + [length of group name]) × [number of lists]

User data = (64 + [length of column name] + [length of column]) × [number of columns]

7.4.9 Authentication Module and Authentication Server default values

Parameter name	Description	Default value
MAXREQTHREAD	Number of request threads	10
MAXDBCONNECT	Number of DB connection pools	2
REQQUESIZE	Request queue size	20
LOGBUFSIZE	Log buffer size	1,000
MAXREPTHREAD	Number of replication threads	5
REPQUESIZE	Replication queue size	1,000

Related kernel parameters for HP-UX

Parameter name	Description
max_thread_proc	Maximum number of threads that one process can Create
maxdsiz	Maximum process data segment size
maxdsiz_64bit	

Parameter name	Description
maxfiles	Soft limit for open files
maxfiles_lim	Hard limit for open files

These parameters should be set to the greater of the OS-installed default values or the values calculated by the formula above.
There is no need to modify the kernel parameters for Red Hat Enterprise Linux.

7.4.10 Calculating the maximum number of file descriptors

The calculation formula is the following:

Maximum number of file descriptors =
MAXDBCONNECT + MAXREQTHREAD + REQQUESIZE +
MAXREPTHREAD + 20

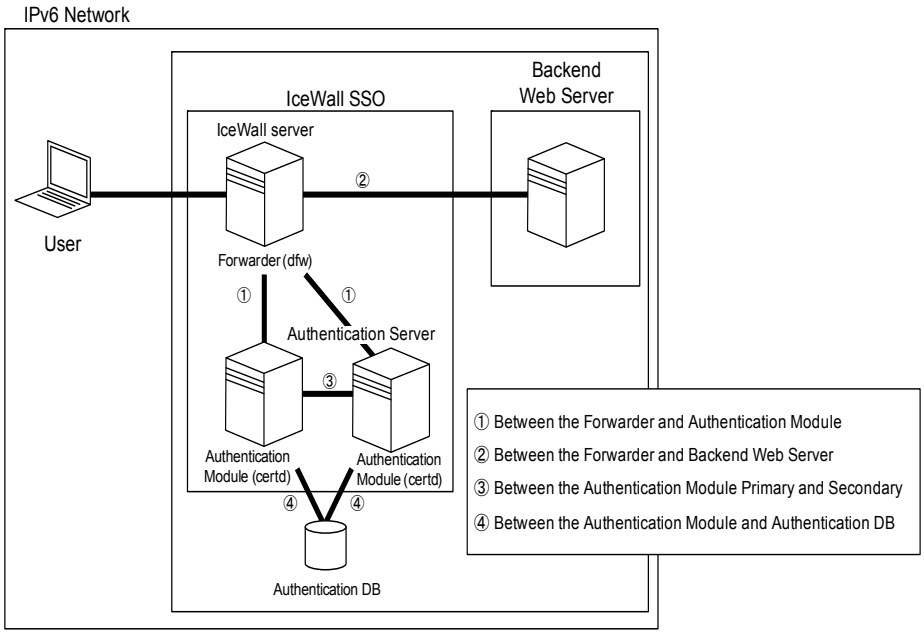
Maximum number of file descriptors for the Authentication Module for version 10.0:

HP-UX: 60,000, Linux: 8,192

If the calculated value exceeds the maximum number of file descriptors for the Authentication Module, the related parameters should be adjusted. Otherwise, the Authentication Module may encounter a fatal exception.

8 IPv6 Support 10.0

IceWall SSO can be used in an IPv6 network.



8.1 IPv6 specifications for IceWall SSO

8.1.1 Prerequisite conditions

Each module must be in an IPv6 network, and the servers and the OS that run each module must be able to communicate with each other at machine level through IPv6.

8.1.2 IP address format in IPv6

For IceWall SSO, IP addresses in IPv6 format are enclosed in square brackets ([~]). IP addresses in IPv6 format can also be specified in abbreviated form.

- Examples of IP address formats in IPv6

```
[2001:0db8:0000:0000:0000:0000:0000:0001]
[2001:0db8:0:0:0:0:0:1]
[2001:0db8::1]
```

* These addresses are treated as the same value.

- Host name format in IPv6:

When communicating over IPv6 to a connection specified as a host name, the host name is enclosed in square brackets ([~]).

- Configuring a connection port number in a specified address

The port number is placed outside of the square brackets ([~]).

• Examples of IPv6 formats that contain connection ports

```
[2001:0db8::1]:80
[iwsrv02]:80
```

8.1.3 Precedence of IP versions during communication

When a host name specified for an IceWall SSO connection is used, IPv4 takes precedence if the host name is not enclosed in square brackets ([~]). If the host name is enclosed in square brackets ([~]), IPv6 takes precedence.

Configuration example of IPv4 precedence

```
CERT=icewall.certd.com:14142
```

Configuration example of IPv6 precedence

```
CERT=[icewall.certd.com]:14142
```

The IP version is determined as follows:

1. Network information is acquired from the host name or address.
2. IP version is determined from the network information.
   2-1. If the network information contains both IPv6 and IPv4:
      2-1-1. If there are square brackets ([~]) enclosing the the host name or IP address, the connection's IP version will be considered IPv6.*
      2-1-2. If there are no square brackets ([~]) enclosing the host name or IP address, the connection's IP version will be considered IPv4.*
   2-2 If the network information contains IPv6 only, the connection's IP version will be considered IPv6.
   2-3 If the network information contains IPv4 only, the connection's IP version will be considered IPv4.
   2-4. If the network information contains neither IPv6 nor IPv4, it will be considered an error.
3. A connection is established using the IP version determined from the network information.

* If connection by the method with precedence fails, a reconnection through the other method will not be attempted. A failure when connection is attempted through the method with precedence will be judged an error.

Example: When a host name of “icewall.certd.com” and acquired IP addresses of “127.0.0.1” and “2001:0db8::1”:
If icewall.certd.com is specified, a connection will be made to “127.0.0.1” over IPv4.
If [icewall.certd.com] is specified, a connection will be made to “2001:0db8::1” over IPv6.

8.2 IPv6 support in Forwarder

IPv6 can be used for the following functions:
- Communication with the Authentication Module
- Permanent redirect after authentication
- Communication with a Backend Web Server
- Authentication control using URLs
- Acquiring content on a remote server specified in the host configuration file
- Agent connection control
- Excluded host settings for the host filter

8.2.1 IPv6 support in the Forwarder configuration file (dfw.conf)

・CERT

Format

CERT=[host name|IP address]:port number,....

Configuration examples

```
CERT=[1234:5678::9A]:14142
CERT=[certsvr_replica]:14142
```

・CERTLB

Format

CERTLB=authentication module identification key=[host name|IP address]:port number,...

Configuration examples

```
CERTLB=a=[1234:5678::9A]:14142
CERTLB=b=[certsvr_replica]:14142
```

・CERTFAILBACK

Format

CERTFAILBACK=[host name|IP address]:port number,....

Configuration examples

```
CERTFAILBACK=[1234:5678::9A]:14142,[1234:5678::9B]:14142
CERTFAILBACK=[certsvr_master]:14142,[certsvr_replica]:14142
```

・REDIRECT

Format

REDIRECT=redirected URL

Configuration example

```
REDIRECT=http://[1234:5678::9B]/index.html
```

・HOST

Format

HOST=alias name=[host name]:port number (proxy server name:proxy port number)

Configuration examples

```
HOST=alias1=[2001:0db8::1]:80([2001:0db8::2]:8080)
HOST=alias2=[www.example.com]:80([proxy.example.com]:8080)
```

・SHOST

Format

SHOST=alias name=[host name].port number

Configuration examples

```
SHOST=alias1=[2001:0db8::1]:443
SHOST=alias2=[www.example.com]:443
```

・REV_PATH

Format

REV_PATH=URL of backend web server performing authentication control

Configuration examples

```
REV_PATH=http://[2001:0db8::1]/
REV_PATH=http://[www.example.com]/
```

・LOCATIONURL

Format

LOCATION=redirectable URL

Configuration examples

```
LOCATIONURL=http://[2001:0db8::1]/
LOCATIONURL=http://[www.example.com]/
```

・AGENT_PERMIT

Format

AGENT_PERMIT=agent alias,determination method,value

Configuration examples

```
AGENT_PERMIT=ALIAS,IPV6,1234:5678::9
AGENT_PERMIT=ALIAS,IPV6,1234:5678::1-1234:5678::9
```

* If IPv6 is set by the AGENT_PERMIT parameter and the determination method is "IPV6", the value portion should not be enclosed in square brackets ([~]).

8.2.2 IPv6 support for parameter values in the host configuration file (any file name)

・SVREXCEPTION

Format

SVREXCEPTION=names of excluded hosts

Configuration examples

```
SVREXCEPTION=http://[2001:0db8::1]
SVREXCEPTION=http://[www.example.com]
```

・SYSERR

Format

SYSERR=fully specified file name,status code,status message

Configuration examples

```
SYSERR=http://[2001:0db8::1]/html/webserver_down.html,200,System Error
SYSERR=http://[www.example.com]/html/webserver_down.html,200,System
Error
```

・SYSTOUT

Format

SYSTOUT=fully specified file name,status code,status message

Configuration examples

```
SYSTOUT=http://[2001:0db8::1]/html/webserver_timeout.html,200,System Error
SYSTOUT=http://[www.example.com]/html/webserver_timeout.html,200,System
Error
```

・ERRKEY

Format

ERRKEY=search keyword,fully specified HTML file

Configuration examples

```
ERRKEY=Server Error,http://[2001:0db8::1]/html/system_error.html
ERRKEY=Server Error,http://[www.example.com]/html/system_error.html
```

8.2.3 IPv6 support for parameter values in the HTML configuration file (html.conf)

- You can describe all configuration parameters of the HTML configuration file in IPv6 format.

Format

Configuration parameter=fully specified HTML file,status code,status message

Configuration examples

```
LOGIN_UID=http://[2001:0db8::1]/html/login.html,200,IceWall Login
LOGIN_UID=http://[www.example.com]/html/login.html,200,IceWall Login
```

8.2.4 Restrictions on absolute URLs in Backend Web Server content

Forwarder's URL conversion function converts absolute URLs included in content, but the following restrictions apply when the absolute URL is described in IPv6 format:

Square brackets ([~]) are required if the host name is described by an IP address in IPv6 format.

```
<a href="http://[1234:5678::1]/index.html">
<a href="http://[1234:5678::1]:81/index.html">
```

If there are no square brackets for an IP address in IPv6 format, URLs can not be converted since they can not be parsed by normal rules.

```
<a href="http://1234:5678::1/index.html">
<a href="http://1234:5678::1:81/index.html">
```

Square brackets ([~]) are not needed if the host name is described as a host name in IPv6.

```
<a href="http://www.hp.com/index.html">
<a href="http://www.hp.com:81/index.html">
```

If the host name is enclosed in square brackets ([~]), URLs can not be converted since they can not be parsed by normal rules.

```
<a href="http://[www.hp.com]/index.html">
```

8.3 IPV6 support in the Authentication Module

IPv6 can be used for the following functions:

- Receiving an ICP request in IPv6 communications
- Communicating between the Authentication Module master and replica
- Communicating between the Authentication Module and the Authentication DB
- Access control
- Request control
- Communication of Authentication Module commands

8.3.1 Authentication Module reception of IPv6 requests

In order for the Authentication Module to receive an IPv6 request, you must first configure it to do so.
For configuration parameter details, see the section on the IPV6LISTEN parameter in the “IceWall SSO Reference Manual.”

8.3.2 IPV6 support for parameter values in the Authentication Module configuration file (cert.conf)

・CERT

IPv6 configuration settings can be made for communication with a replicated Authentication Module.

Format

CERT=[host name|IP address]:port number

Configuration examples

```
CERT=[1234:5678::9A]:14142
CERT=[certsvr_replica]:14142
```

・DBHOST (LDAP, MSAD, NED editions)

Format

DBHOST=[host|IP address]:port for search and update:BIND port number,...

* The port number for search and update, and the BIND port number can be omitted.

Configuration examples

```
DBHOST=[1234:5678::9A]:389,[1234:5678::9B]:389
DBHOST=[ldapsvr_master]:389,[ldapsvr_replica]:389
```

8.3.3 IPV6 support for configuration parameters of the access control file (cert.acl)

・Host component of the URL

Format

target URL=group name

Configuration example

```
http://[1234:5678::90]/=ALL
```

* You may use an abbreviated format if you specify an IP address in IPv6 format in the target URL.

The IPv6 address component can be accessed as the same value for the following access control request from Forwarder.

・URL included in an access control request from Forwarder

```
http://[1234:5678:0000:0000:0000:0000:0000:90]/
```

・Access control file configuration for the Authentication Module

```
http://[1234:5678::90]/=ALL
```

Users in the ALL group have access permission for http://[1234:5678:0000:0000:0000:0000:0000:90]/.

8.3.4 IPv6 support for parameter values in the group configuration file (cert.grp)

You can configure REMOTE_ADDR in IPv6 as a conditional expression in a group configuration.

・REMOTE_ADDR

Format

Group name,conditional expression|logical expression

Configuration example

```
GRPIPV6,REMOTE_ADDR=[1234:5678::90]-[1234:5678::A0]
```

* In this example, the group name GRPIPV6 is specified for users whose IPv6 REMOTE_ADDR lies between 1234:5678::90 and 1234:5678::A0.

8.3.5 IPV6 support for parameter values in the request control configuration file (request.acl)

You can configure SOURCE_ADDR in IPv6 as a conditional expression for designating a request.

・SOURCE_ADDR

Format

TARGET=conditional expression
{
}

Configuration example

```
TARGET=SOURCE_ADDR=[1234:5678::50]-[1234:5678::60]
{
REJECT=LOGIN
}
```

* In this example, logins are prohibited at IPv6 SOURCE_ADDR between 1234:5678::50 and 1234:5678::60.

8.3.6 IPv6 support for various commands in the Authentication Module

If the Authentication Module is configured to receive IPv6 only, all commands (end-cert, reload-cert, cdump-cert, logout-cert) must also send requests through IPv6. When using IPv6, specify the argument for the -H command option in square brackets ([localhost]) for each command.

IPv6 support for the stop command

```
#!/bin/sh

export IW_HOME=/opt/icewall-sso
export SHLIB_PATH=$IW_HOME/lib/certd
export LD_LIBRARY_PATH=$SHLIB_PATH

/opt/icewall-sso/certd/bin/certd -F -K -H [localhost] -P 14142
```

9 Restrictions on Various Authentication DBs

This chapter describes restrictions that are imposed by the functions of the Authentication DB used. For details on individual parameters, see the “IceWall SSO Reference Manual.”

9.1 Common restrictions for Authentication DBs

Special characters, other than alphanumeric characters, that may be used for user IDs are as follows:

```
! % $ & @ _ - .
```

* A blank space character is not supported.

Further, a period “.” may not be used at the end of user IDs.
Operation of the system when used with special characters other than those described above is not guaranteed.

The length of the user ID is limited to 64 bytes. This maximum value cannot be changed by a configuration parameter. If a user ID longer than 64 bytes is received from a user, the following error message is displayed on the user ID error page. At this time, a login request for the Authentication Module from Forwarder does not occur.

```
Warning: UserID MaxSize Error. [ED03610-10208]
```

9.2 Restrictions for an ORACLE Authentication DB

The following restrictions apply when Oracle 11g is used as the Authentication DB:

- Failover function for the Authentication DB provided by failover option is not supported.
- Character encoding conversion of user data is not supported.

9.3 Restrictions for an LDAP, OpenLDAP, or NED Authentication DB

The following restrictions apply when Sun Java System Directory Server 7.0, HP Directory Server 8.1, Novell eDirectory 8.8 or OpenLDAP 2.4.16 is used as the Authentication DB:

- Audit log output function cannot be used.
- Reference table function cannot be used.

9.4 Restrictions for an MSAD Authentication DB

The following restrictions apply when Microsoft Active Directory is used as the Authentication DB:

- Audit log output function cannot be used.

- Reference table function cannot be used.
- Character encoding conversion of user data is not supported.

9.5 Restrictions on a MySQL Authentication DB

The following restrictions apply when MySQL is used as the Authentication DB:

- Audit log output function cannot be used.
- Character encoding conversion of user data is not supported.

9.6 Restrictions for a CSV Authentication DB

The following restrictions apply when a text Authentication DB file is used as the Authentication DB:

- Audit log output function cannot be used.
- Reference table function cannot be used.
- Failover function for the Authentication DB provided by the failover option is not supported.
- Replication of CSV file is not supported when using the failover option.
- Character encoding conversion of user data is not supported.
- As a relatively small number of users (approximately 1,000) are anticipated, performance deteriorates if a large number of users registers.
- When the user data starts the Authentication Module, the data is expanded in the memory. For this reason, if the text file user data is modified during start up, the change does not go into effect.
  If an update is issued from the Authentication Module, the content of the memory is overwritten.

9.6.1 CSV file specifications

The specifications of files that can be used as a text Authentication DB are as follows:

- Create the file as a text file.
- Set the file owner to “iwadmin.” (Authentication Module execution user)
- Set the file's access permissions to “644.”
- Configure column names in the first line.
- Create user data in the following lines.
- The maximum number of bytes per line is 4,096.
- Separate data only with commas.
- To use a comma in the data, place the comma between double quotes.
- Use a series of commas to omit values.
- If the column number and value do not match, the value is regarded as omitted.
- Line feed code reading supports CRLF and LF.
- Line feed codes are standardized to LF after file update.

Example of a text file (the password is set to “non-encryption”)

```
USERID,PASSWD,PWDEXIRE,LOCKCOUNT,...
user01,password,20030430115959,0,...
user02,user02,20030430115959,0,...
user03,03user,,0,...
```

10 Log Files

The log files output by IceWall SSO are described in this chapter. For details on individual parameters, see the “IceWall SSO Reference Manual.”

10.1 Access logs

Access logs are output by Forwarder and the Authentication Module.
The output format of the respective access logs is different.

10.1.1 Forwarder access log format (10.0)

[Date and time] [Time 1] [Time 2] [Time 3] [User ID] [Request method] [Request URL] [Content size] [IP address] [Backend web server status] [Transaction ID]

Parameter name	Description
Time 1	Shows the time elapsed from starting up Forwarder until connecting to the Backend Web Server. The unit is in seconds, up to six decimals. When 1000 seconds or higher, shown as 999.999999. (10.0) This time includes the time spent sending the request to the Authentication Module and receiving the response from it. If the following UserExit routines are used, the processing times are included: Entry before request analysis Before sending the access control request to the Authentication Module *1 After receiving the access control response from the Authentication Module *1 Entry before sending request to the web server *1 Only when using ICP 2.0
Time 2	Shows the communication time with the Backend Web Server. Depending on the reception mode of Forwarder, the measurement range may be different. • When using the buffering mode from the time of opening the connection to the web server until receipt of content is complete. • When using the non-buffering mode from the time of opening the connection to the web server until the output to the web server. The unit is in seconds, up to six decimals. When 1000 seconds or higher, shown as 999.999999. (10.0)

Parameter name	Description
Time 3	Shows the time elapsed from receipt of the content until the output to the web server. When using non-buffering mode, the time necessary is nearly 0. The unit is in seconds, up to six decimals. When 1000 seconds or higher, shown as 999.999999. (10.0) If the following UserExit routines are used, the processing times are included: Entry after receiving response from the web server Entry before outputting content to client
User ID	Outputs the user ID that is subject to request processing.
Request method	Outputs the request method.
Requested URL	Outputs the requested URL.
Content size	The received content size. The unit is in bytes.
IP address	The IP address of the client.
Backend Web Server status code	Outputs the HTTP status code of the response received from the Backend Web Server.
Transaction ID	Outputs the transaction ID when configured to output the transaction ID (TRANSID=1). If the transaction ID cannot be retrieved, there is no output. * Only when using ICP 2.0

Example: [2002/08/05 15:24:23] 0.008923 0.027168 0.000053 user01 GET [www.svr.com:80/] 3301 xxx.xxx.xxx.xxx 200 TID=xxxxxxxxxx

The IP address can be output by setting the output level (ALEVEL of dfw.conf) to two (ALEVEL=2).

10.1.2 Authentication Module access log format (10.0)

[Date and time] [Log message] [Transaction ID] [Message ID]

Example: [2010/07/09 11:08:11] Unknown UserID. UserID=user01 TID=xxxxxxxxxxx [ACxxxxx-xxxxx]

Note: The log message portion which exceeds 1024 characters is not correctly output.

If the performance information output configuration parameter is set to 1 (LOGPERF=1 in cert.conf), the time required for processing the access to the Authentication DB and the Authentication DB's internal request processing time is included in the Authentication Module access log in the following format:

[Date and time] [Keyword] [Thread ID] [Request type] [User ID] [Request processing time] [DB processing time 1] [DB processing time 2] ... [DB processing time n] [Transaction ID]

<table>
<tr><th>Parameter name</th><th>Description</th></tr>
<tr><td>Keyword</td><td>“PERF” is the fixed keyword to indicate a performance log.</td></tr>
<tr><td>Thread ID</td><td>Outputs the thread ID of the requested thread.</td></tr>
<tr><td>Request type</td><td>Outputs the type of the request. The following output strings represent the various request types.
<table>
<tr><th>Request</th><th>String</th></tr>
<tr><td>User ID login</td><td>LOGINUID</td></tr>
<tr><td>Forced user ID login</td><td>FLOGINUID</td></tr>
<tr><td>Certificate login</td><td>LOGINCERT</td></tr>
<tr><td>Forced certificate login</td><td>FLOGINCERT</td></tr>
<tr><td>SAML login</td><td>LOGINSAML</td></tr>
<tr><td>Forced SAML login</td><td>FLOGINSAML</td></tr>
<tr><td>Federation login</td><td>LOGINFEDE</td></tr>
<tr><td>Forced federation login</td><td>FLOGINFEDE</td></tr>
<tr><td>User ID access control</td><td>ACCESSUID</td></tr>
<tr><td>Certificate access control</td><td>ACCESSCERT</td></tr>
<tr><td>Password change</td><td>PWDCHG</td></tr>
<tr><td>Logout</td><td>LOGOUT</td></tr>
<tr><td>Automatic logout</td><td>AUTOLOGOUT</td></tr>
<tr><td>External cache reference
(Login status/Forced logout)</td><td>FLOGOUT</td></tr>
</table>
</td></tr>
<tr><td>User ID</td><td>Outputs the user ID of the user for whom the request is being processed. If the user ID cannot be acquired, there is no output.</td></tr>
<tr><td>Request processing time</td><td>Outputs the elapsed time of the request process.
The unit is seconds, up to six decimals. (10.0)
Values exceeding 999 hours 59 minutes and 59 seconds cannot be displayed correctly.</td></tr>
</table>

<table>
<tr><th>Parameter name</th><th>Description</th></tr>
<tr><td>DB processing time</td><td>Outputs the time elapsed for the DB process. Outputs for each performed DB operation. The keyword and the processing time are output as a set.
The unit is in seconds, up to six decimals. (10.0)
<table>
<tr><th>DB operation</th><th>Keyword</th></tr>
<tr><td>Authentication DB/Reference DB access (Select)</td><td>S</td></tr>
<tr><td>Authentication DB update process (Update)</td><td>U</td></tr>
<tr><td>Historical DB process (Insert)</td><td>I</td></tr>
<tr><td>Bind operation to Authentication DB (Bind) [10.0]</td><td>B</td></tr>
</table>
</td></tr>
<tr><td>Transaction ID</td><td>Outputs the transaction ID when configured to output the transaction ID (TRANSID=1). If the transaction ID cannot be retrieved, there is no output.
* Only when using ICP 2.0</td></tr>
</table>

```
Example:[2010/07/09 11:08:11] PERF 8 LOGOUT user01 0.234567 U:0.043210
I:0.032109 TID=xxxxxxxxxxx [ACxxxxxxxxxx]
```

In this example, the logout processing time of user ID “user01” from the requested thread with thread ID “8” took 0.234567 seconds. The breakdown of the logout process shows 0.043210 seconds for the Authentication DB update process and 0.032109 seconds for inserting into the historical DB.

10.2 Error logs

Error logs are output by Forwarder and the Authentication Module.
The format of the output is the same for all errors.

10.2.1 Error log format

[Date and time] [Error level] [Error message] [Transaction ID] [Message ID]

```
Example: [2002/08/05 13:55:38] Warning: Backend Web Server Down.
[www.svr.com:80] TID=xxxxxxxxxxx [EXxxxxx-xxxxx]
```

10.3 Audit log

The audit log is output as a table by only the ORACLE edition of the Authentication Module.
The following information is output:

[Sequential number], [Date and time], [User ID], [Type], [Process result], [Log message], [IP address]
The audit log is output based on the settings of the audit log output table name (LOGDBTBL in cert.conf), the audit log column (LOGDBATTR in cert.conf), the audit log output sequential object (LOGDBSEQNAME in cert.conf), and the log column information file (logdbattr.conf).

10.4 Security log

The security log is output by Forwarder at the time of login, logout, password change, and agent transfer success.

10.4.1 Security log format

[Date and time] [Message] [Agent host name] [Agent key] [User ID] [IP address]

Example: [2002/08/05 13:55:38] Agent userid login. agent=xxxx key=xxxx uid=xxx ip=xxx.xxx.xxx.xxx

The agent host name, user ID and IP address are output into the log if the output level (SECLEVEL in dfw.conf) is set to 1 (SECLEVEL=1), and the agent key is output if the level is set to 2 (SECLEVEL=2).

10.5 Information log

It is possible to output the performance information resulting from executing the info-cert command as an information log (INFORMATION in cert.conf), separately from the access log of the Authentication Module. The content of the output is the same as the access log (however, the message ID is not included in the output).

10.6 Performance log

It is possible to output the time required for processing access to the Authentication DB and the Authentication DB's internal request processing time as a performance log (PERFORMANCE in cert.conf), separately from the access log of the Authentication Module. The content of the output is the same as the access log (however, the message ID is not included in the output).
When outputting the performance log, it will be output regardless of the performance information output setting (LOGPERF).

10.7 Trace time log 10.0

The detailed internal processing time from when Forwarder receives the request until the response is returned can be output as the trace time log. Normally, the trace time log does not need to be set. It should be set only when necessary such as when providing information to technical support.