IceWall SSO

Version 10.0

Web Application Developer's Manual

August 2010

Printed in Japan

HP Part No. B1544-97010

Rev. 111012A

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

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 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.

The copyright of this document is assigned to Hewlett-Packard Development Company, L.P.
No part of this document may be photocopied or reproduced without prior written consent by Hewlett-Packard.

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.

IceWall is a trademark of Hewlett-Packard.

Adobe is a trademark of Adobe Systems Incorporated in the United States and/or other countries.
Intel, the Intel logo, Intel. Leap ahead., Intel. Leap ahead. logo, Itanium and Itanium Inside are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
Microsoft and Windows are trademarks or registered trademarks of Microsoft Corporation in the United States and other countries.
Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

– Table of Contents –

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

1 Introduction

IceWall SSO is software that provides single sign on authentication and authorization functions for multiple Backend Web Server applications. By using IceWall SSO, Backend Web Server application developers can reduce the number of man-hours spent developing authentication, authorization, and security functions. This document describes the information necessary for Backend Web Server application development using IceWall SSO.

This document mainly describes operations in SSO mode, which is used for login and access control. In addition, it contains notes on reverse proxy mode that does not require authorization, and also includes only the necessary explanations for operations using Agent Option.

For details on installing and using Agent Option, see the “IceWall SSO Installation Guide for Agent Option.” For details on Backend Web Server application development when using IceWall MCRP, see the “IceWall MCRP Web Application Developer's Manual.”

1.1 About version designations in the text

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

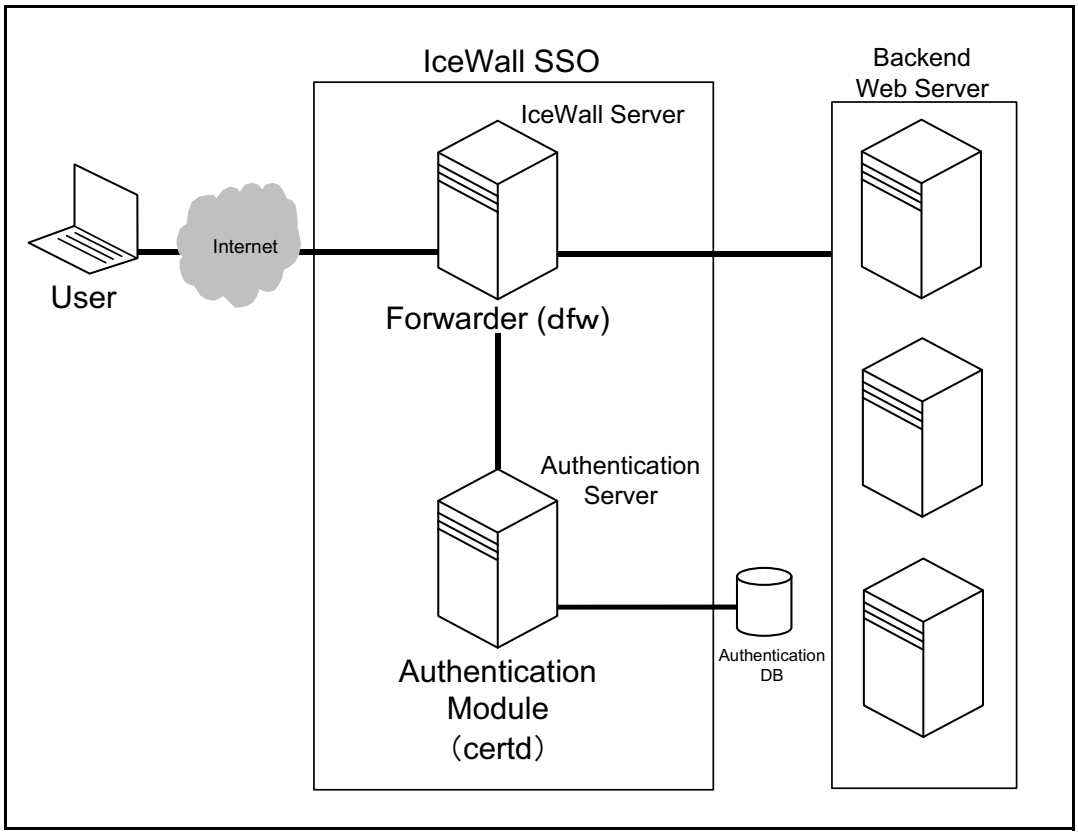
Designation	Meaning
10.0	An item added with the version enclosed by the square. In this case, the designation indicates the item was added with 10.0.
(10.0)	An item where the specification was changed or function added with the version enclosed in the oval. In this case, the designation indicates a specification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

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

2 Basic Configuration

IceWall SSO is middleware developed to provide single sign on authentication. It consists of two modules: Forwarder (dfw) and the Authentication Module (certd).

Forwarder runs on a server called the “IceWall server,” and the Authentication Module runs on the “Authentication Server.”



(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

3 Basic Functions

The functions provided by IceWall SSO are as follows.

3.1 Login function

This function enables users to begin to access content on Backend Web Servers. This function performs authentication of user IDs and passwords in connection with the Authentication DB (a directory service or database).

3.2 Logout function

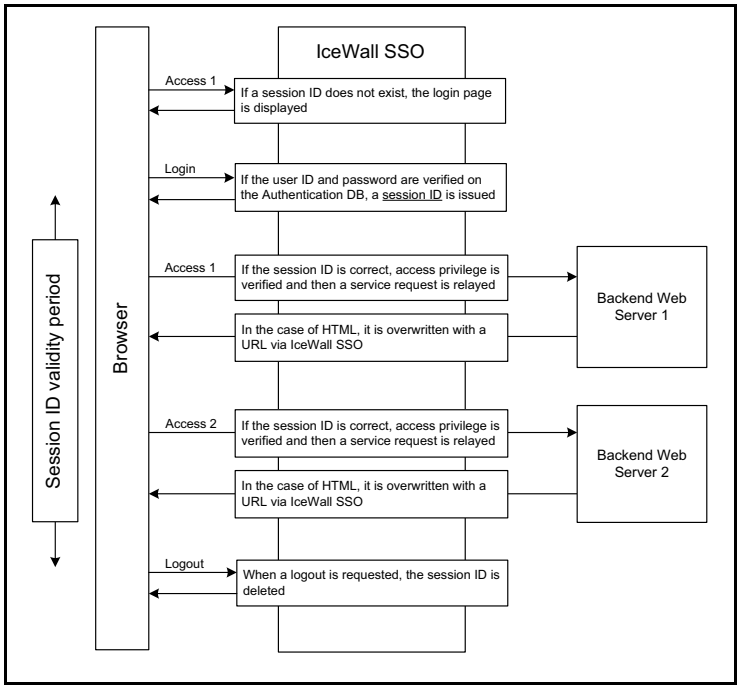
This function allows the user to terminate access to Backend Web Servers. When this function is used, the users can no longer access the Backend Web Servers. (To access the Backend Web Servers again, users must use the login function.)

3.3 Access control function

This function controls a user's access to content on Backend Web Servers, based on the authorization information of the group to which the authenticated user belongs.

3.4 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 performs user access control (access permit/access denial per URL).



(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

Note: The session ID is set in the browser as a cookie or URL-Cookie (not the query string, but in a format in which the session ID is embedded in a portion of the URL path). The name of a cookie can be arbitrarily specified (the default name is “IW_INFO”).

3.5 Information inheritance function

This function passes user login information and information on the Authentication DB to the Backend Web Server via HTTP headers. The Backend Web Server receives this information as environment variables and so on. The specified Authentication DB column configuration can be passed to a Backend Web Server by specifying it in the IceWall SSO configuration file (host configuration file).

3.6 URL conversion function

This function changes the URL included in the content received from a Backend Web Server into a URL accessible via the IceWall server. This function is used to conceal Backend Web Server information from users.
Since all URLs included in the content are converted to subdirectories of the IceWall server, it appears as if the users are accessing a single web server.

3.7 Keyword conversion function

This function converts strings included in the content received from the Backend Web Server to specified strings. This function can be used to convert any strings that cannot be converted by the URL conversion function described above.

3.8 Reverse proxy function

This function allows the IceWall server, instead of Backend Web Servers, to receive access requests from clients. 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 client access goes through it.

3.9 Password change function

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

3.10 Access logging function

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

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

3.11 SSL communication function

Besides HTTP communication, HTTPS (HTTP over SSL) communication can also be used between the IceWall server and Backend Web Servers. This allows secure single sign on to Backend Web Servers located on the Internet as well.

3.12 Failover function

This function allows the Forwarder to use alternative Backend Web Servers and alternative Authentication Servers in order to improve system availability.

3.13 Client authentication function

This function performs identification using a Client Certificate. Sophisticated identification can be implemented by storing Client Certificates on the browser.

3.14 Automatic Form Authentication function

This function automatically logs a user in to the authentication form for logging into an application on a Backend Web Server. This function can be used with various types of forms.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

4 Summary of Application Development Using IceWall SSO

IceWall SSO relays information from the browser to Backend Web Servers. During this process, IceWall SSO adds to and changes a portion of the HTTP header information.
IceWall SSO converts the URL in the content sent from the Backend Web Server to the path of IceWall SSO and then returns it to the browser. In this way, the URL in the content received from a Backend Web Server can be displayed in the browser as a subdirectory of the IceWall server. IceWall SSO also adds to and changes a portion of the HTTP header information sent from the Backend Web Server.
When developing a system application that uses IceWall SSO, the following items must be taken into consideration.

(1) Review of the connection method from the IceWall server to Backend Web Servers
(2) Information transferred from the IceWall server to Backend Web Servers
(3) Method of using the information transferred to Backend Web Servers
(4) HTML syntax supported by IceWall SSO
(5) URL conversion function and keyword conversion function
(6) Host names or strings for use with these conversion functions in the item (5)
(7) Points to be aware of when using JavaScript
(8) IceWall SSO timeout and application timeout

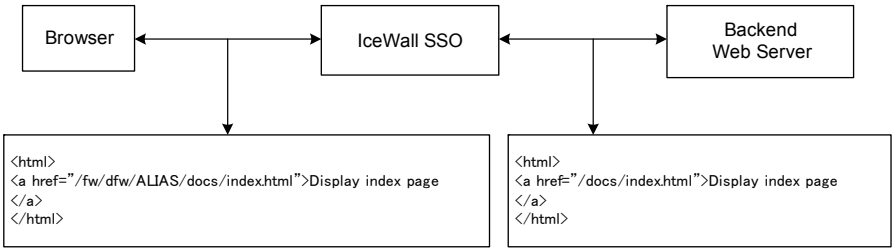
After considering the items above, the application developer should review the content as necessary and ask the IceWall SSO administrator to configure the IceWall SSO environment.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

4.1 Operating model

This manual uses the model below to describe IceWall SSO operations.

4.1.1 Displaying static content accessible via IceWall SSO



IceWall SSO automatically rewrites the URLs included in the HTML of a Backend Web Server into URLs accessible via IceWall SSO. In the example above, the Backend Web Server is defined as “ALIAS.”

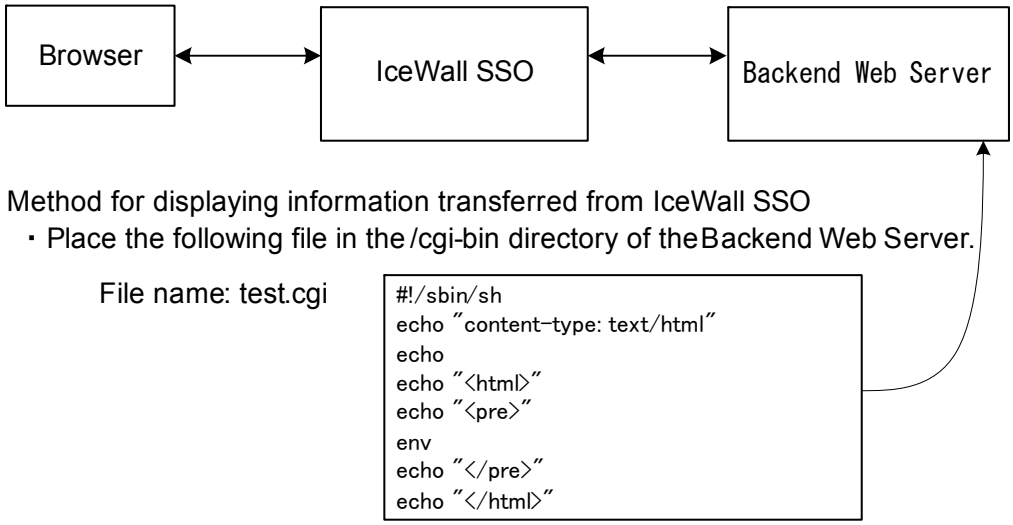
With the exception of certain scripts (JavaScript, etc.), HTML content can be provided on the Backend Web Server irrespective of IceWall SSO. (For details on JavaScript, see “10 JavaScript.”)

Example of HTML tags automatically rewritten by IceWall SSO

- <a href=
- <base href=
- <frame src=
- <form action=
- <base target=
- <img src=
- <script src=
- <body background=
- <td background=
- <tr background=
- <table background=
- <applet codebase=
- <input src=

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

4.1.2 Executing CGI via IceWall SSO



(1) Method for displaying information transferred from IceWall SSO:

- Place the following file in the /cgi-bin directory of the Backend Web Server.
  File name: test.cgi

- This CGI is invoked from the browser.
  URL=http://[IceWall SSO host name]/fw/dfw/ALIAS/cgi-bin/test.cgi

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

• The information usable by the CGI is displayed in the browser.

```
_=/usr/bin/env
LANG=C
HTTPS=OFF
HTTP_JYUUSYO=Tokyo-to
HTTP_NAME=John Smith
PATH=/sbin:/usr/sbin:/usr/bin
REMOTE_HOST=127.0.0.1
HTTP_HOST=[backend web server]
GATEWAY_INTERFACE=CGI/1.1
HTTP_ACCEPT_ENCODING=gzip
HTTP_SERVER_SOFTWARE=Netscape-Enterprise/3.6 SP3
HTTP_CONNECTION=close
SERVER_SOFTWARE=Netscape-Enterprise/3.6 SP3
HTTP_UID=user01
SERVER_URL=http://[backend web server]
REQUEST_METHOD=GET
HTTP_ACCEPT_CHARSET=Shift_JIS,*,utf-8
HTTP_USER_AGENT=Mozilla/4.7 [ja] (WinNT; I)
HTTP_ACCEPT=image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*
HTTP_ACCEPT_LANGUAGE=ja,en
SCRIPT_NAME=/cgi-bin/test.cgi
SERVER_PORT=80
SERVER_PROTOCOL=HTTP/1.0
REMOTE_ADDR=127.0.0.1
TZ=JST-9
HTTP_COOKIE=IW_INFO=458f8945e6d0edfbcbfc6f42a98a3ba8
```

(2) Method for changing displayed contents based on the information transferred from IceWall SSO:

• Place the following file in the /cgi-bin directory of the Backend Web Server. (The below is an example of when the name header is sent from IceWall SSO as user information)

File name: test2.cgi

```
#!/sbin/sh
echo "content-type: text/html"
echo
echo "<html>"
echo "<h1>Welcome "
echo $HTTP_NAME
echo "</h1>"
echo "</html>"
```

• This CGI is invoked from the browser.
URL=http://[IceWall SSO host name]/fw/dfw/ALIAS/cgi-bin/test2.cgi

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

- The user information is displayed in the browser.

Welcome John Smith

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

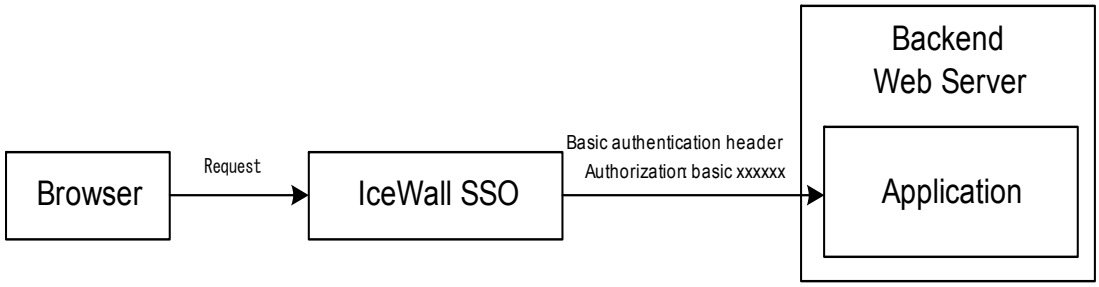
5 Method for Connecting to a Backend Web Server

5.1 Connecting to a Backend Web Server that does not require authentication

There are no particular problems to consider when connecting to a Backend Web Server that does not require authentication.

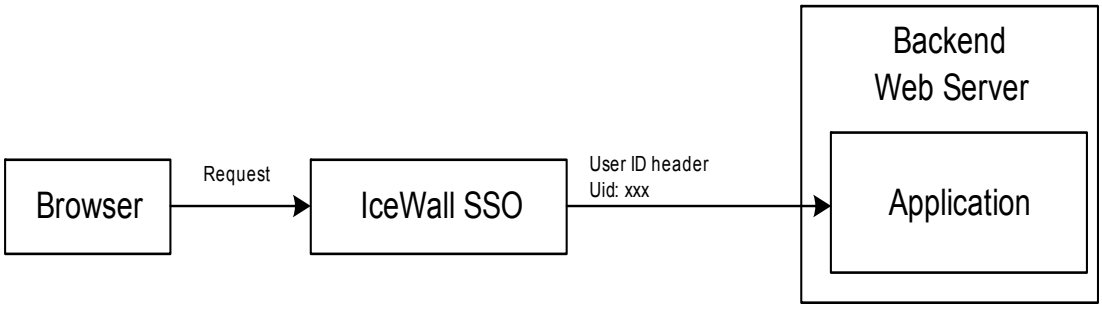
5.2 Backend Web Server that requires basic authentication

IceWall SSO can connect to a Backend Web Server by using basic authentication. By default, the user ID and password entered at login can be used as the basic authentication user ID and password.
Basic authentication for the Backend Web Server with a user ID and password that differ from the IceWall user ID and password is also possible by selecting the desired Authentication DB columns for the user ID and password to perform authentication on the Backend Web Server.



5.3 Applications that retrieve the user ID from the HTTP header

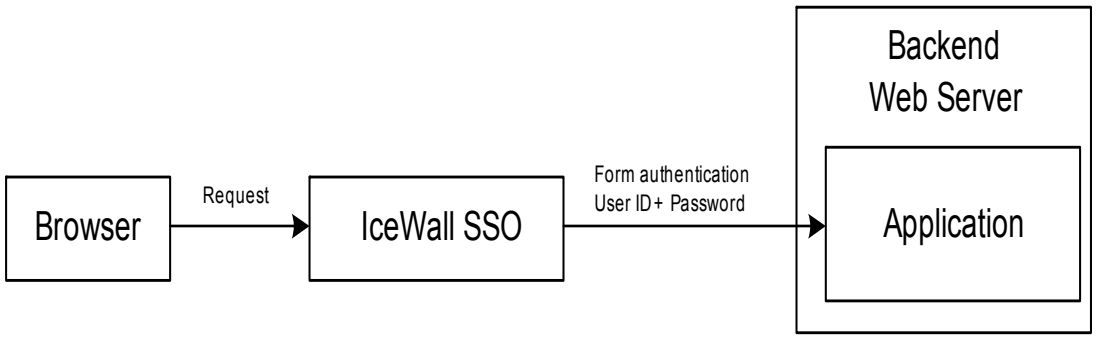
IceWall SSO notifies the Backend Web Server of the requesting user's user ID in the HTTP header.
An application on the Backend Web Server can retrieve the user ID as an environment variable or header information.



5.4 Applications that require form authentication

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

IceWall SSO can connect to a Backend Web Server by automatically logging in for form authentication on the Backend Web Server. The user ID and password can be a fixed value or arbitrarily selected from the Authentication DB user information. In addition, IceWall SSO also supports automatic login for applications that use cookies to notify the client of the session information after login.



(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

6 Restrictions on the Backend Web Server

Restrictions on the Backend Web Servers of IceWall SSO are as follows.

6.1 HTTP request headers (from Forwarder to a Backend Web Server)

The HTTP headers from the browser are basically sent from Forwarder to a Backend Web Server without being changed. HTTP header values can be received as HTTP header information by the Backend Web Server (for the CGI program to reference these values, it usually references an environment variable labeled “HTTP_[header name],” for example, HTTP_USER_AGENT).
Please be aware of the following points.

- There are headers requiring special attention. (Host name, etc.)
- Some headers are added by IceWall SSO.
- Information on the Authentication DB can be transferred to a Backend Web Server.
- To disallow the sending of certain headers, specify those headers in the Forwarder configuration file.
- If headers are sent from the browser in the format “xxx_xxx,” IceWall SSO converts them to “Xxx-Xxx” and then sends them. The conversion rules are as follows.
  (1) The first character of each word is capitalized
  (2) “_” is converted to “-.”
  (3) The next character after each converted “-” is capitalized.
  (4) All other characters are converted to lower case.

6.1.1 Headers requiring special attention

IceWall SSO controls the following headers. These headers are controlled in the same manner in reverse proxy mode.

Header name	Description
Referer	This is the original link source URL. This is not the path name accessible via the IceWall server.
Host	The Backend Web Server's host name and port number are sent.
Authorization	Content sent from the browser is not transferred.
Destination	This is the target URI for the COPY and MOVE methods. This is not the path name accessible via the IceWall server. (Primarily used with the WebDAV protocol)

6.1.2 Added headers

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

In addition to the headers sent from the browser, there are also headers that are added by the IceWall SSO server. These headers are added only in SSO mode.

Header name	Description
Authorization	Passes the data for basic authentication in the following format. Basic *[user ID]:[password]* (The italicized portion is encoded in base 64.) You can specify whether or not to pass this header in the configuration. (REMOTE_USER is referenced.)
Session ID name 10.0	Passes a unique (32-digit/64-digit) session ID issued each time the user logs in as a cookie with the name COOKIENAME. (This is retrieved from HTTP_COOKIE.) Default: IW_INFO
Session 10.0	Passes a unique (32-digit/64-digit)* session ID issued each time the user logs in. (HTTP_SESSION is referenced.) From version 10.0, the header name when sending the session ID can be freely changed with the HEADER_NAME_SID parameter.
Uid 10.0	Passes the user ID. (HTTP_UID is referenced.) From version 10.0, the header name when sending the user ID can be freely changed with the HEADER_NAME_UID parameter.
Transaction ID 10.0	Passes a unique transaction ID issued the first time the web browser accesses the server. The header name when sending the transaction ID can be freely changed with the HEADER_NAME_TID parameter.
Authentication DB column (attribute) name	Passes the column name (specified in the configuration file) and contents of that column. (“HTTP_[authentication DB column name]” is referenced.)
Description	Passes the contents of the column if the “DESCRIPTION” column name is specified in the configuration file. (“HTTP_DESCRIPTION” is referenced.)
DescriptionNN	Passes the contents of the “DESCRIPTION” column as multiple headers, separated by commas (“,”). “NN” in the header name is a two-digit sequential number. (For 1 to 9, this should be expressed with two digits as 01 to 09.) Example: When Description is data1, data2, and data3 Description01: data1 Description02: data2 Description03: data3 This is only valid if the “DESCRIPTION” column name is specified in the configuration file. (“HTTP_DESCRIPTIONNN” is referenced.)

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

Header name	Description
Arbitrary header name	Passes the specified value added to the host configuration file's (sample.conf) HEADER parameter. (“HTTP_[arbitrary header name]” is referenced.)

* To retrieve the session ID on the backend, please note that you must take into account whether the number of digits is 32 digits or 64 digits 10.0

6.1.3 Character encoding included in the HTTP headers

IceWall SSO does not convert the character encoding of the HTTP headers. Even when the Authentication DB user information is configured to be transferred to IceWall SSO by the HTTP header, this information is sent in the character encoding format retrieved from the Authentication DB by IceWall SSO (without conversion).

If the header values contain Japanese (double byte characters) such as kanji and katakana, this may be treated as an error when the request is received, depending on the Backend Web Server application. For this reason, it is important to be careful with character encoding when transferring user information to a Backend Web Server.

Authentication DB types	Character encoding
ORACLE	Character encoding differs depending on the ORACLE language setting.
LDAP,MSAD,NED	If data is saved as Unicode (UTF-8) on the database, it can be converted to an encoding such as Shift_JIS with the IceWall SSO function. However, if Microsoft Active Directory is used for the Authentication DB, the character encoding conversion function is not supported.
CSV	The character encoding saved in the CSV file is used as is.
MySQL	The character encoding saved on the database is used as is.

6.1.4 Encoding of special characters included in the URL

If the URL sent from the browser contains special characters, all the special characters **except those shown below** are converted to hexadecimal code and then sent to the Backend Web Server. The arguments appended to the end of the URL (the query string) are not converted.

```
- _ . ! ~ * ' ( ) /
```

Example: $ → %24 space → %20

Semicolon (;) conversion can be specified as on/off in the configuration file.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

6.2 HTTP response headers (from a Backend Web Server to Forwarder)

The HTTP headers sent from a Backend Web Server are basically sent to the client without being changed, but you can control sending headers and add arbitrary headers with the IceWall server configuration file. If the Content-length header is not sent, you can control whether or not to add it with the IceWall server configuration file.

6.2.1 Headers requiring special attention

The IceWall server controls the following headers. These headers are controlled in the same manner in reverse proxy mode.

Header name	Description
Last-Modified	This is the last modified date of the accessed content.
Location	This is the URL to connect to as the redirect location. This is converted to a URL accessible via the IceWall server.

6.2.2 Character encoding included in the HTTP headers

With regard to character encoding of HTTP headers sent from the Backend Web Server, IceWall SSO sends the headers to the browser without converting the character encoding.

If the header values contain Japanese (double byte characters) such as kanji and katakana, they are also sent to the browser without converting the character encoding. The behavior of the browser when using Japanese is dependent on the browser's specifications.

6.3 Communicating with a Backend Web Server

6.3.1 Version of the communication protocol

IceWall SSO uses HTTP/1.0 for communications with Backend Web Servers. Even if the browser connects to IceWall SSO with HTTP/1.1, communications between IceWall SSO and Backend Web Servers are performed unconditionally using HTTP/1.0.

6.3.2 Keep-Alive configuration

For communications between the Backend Web Servers from IceWall SSO, disable the Backend Web Server's Keep-Alive setting or set the KeepAliveTimeout setting to 0 (zero).
Depending on the configuration of the Backend Web Servers, the Keep-Alive configuration may cause a response delay.
For Backend Web Servers for which the configuration of the Keep-Alive settings cannot be changed, configure the “CLOSETIME” parameter in the host configuration file and adjust the waiting time of disconnection between Forwarder and the Backend Web Server.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

However, if the Backend Web Server is Microsoft Internet Information Services, we recommend using the web server with the web server instance's Keep-Alive configuration enabled. If used when disabled, communications from the web server may remain incomplete even when the content is displayed.

6.3.3 Sending basic authentication from the client

When a basic authentication request from a Backend Web Server is sent to the browser, even if the user ID and password are entered on the browser, that information is not transferred to the Backend Web Server. Similarly, this information is not transferred if Forwarder is operating in reverse proxy mode or for URLs that do not require authentication.

6.3.4 Basic authentication configuration for reverse proxy mode

Because IceWall SSO does not perform user authentication when operating in reverse proxy mode, basic authentication is not performed for Backend Web Servers. Similarly, basic authentication is not performed for URLs that do not require authentication.

6.3.5 Version of SSL connection and usable encryption methods

SSL 3.0 is the only version supported for SSL communication with Backend Web Servers when using the SSL Option. SSL 2.0 cannot be used for SSL communication.
You can check the encryption methods that can be used with IceWall SSO by executing the “openssl ciphers” command on the IceWall server.
Forwarder itself has no restrictions, and it is dependent on the IceWall server's operating environment.

6.3.6 SSL 3.0 cipher with Apache

When the Backend Web Servers use Apache, “TripleDES” must not be set as the SSL 3.0 cipher. This configuration setting may cause a significant degradation of communication performance.

6.3.7 Japanese folder and file names

Since requests for URLs are made with the characters URL-encoded, excluding the unreserved characters specified in RFC 2396 (URI), URLs can access Japanese folder and file names (not all Japanese characters are guaranteed to work).

6.4 Environment variables

Environment variables are referenced during the execution of a CGI program or SSI (Server Side Include). The values in some of these variables differ depending on whether access from a browser passed or did not pass through the IceWall server.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

6.4.1 REMOTE_ADDR and REMOTE_HOST

Normally, the IP address and host name of the client making the request are set in the environment variables REMOTE_ADDR and REMOTE_HOST (for CGI). However, when this request passes through Forwarder, the IP address and host name of the IceWall server are set in the environment variables.
The IP address and host name of the client can be retrieved by sending the environment variables retrievable on the IceWall server as HTTP headers. A sample configuration setting is shown below.

```
HEADER=REMOTE_ADDR,REMOTE_ADDR
HEADER=REMOTE_HOST,REMOTE_HOST
```

In addition, since this information is sent as HTTP header information, it must be retrieved as the environment variables HTTP_REMOTE_ADDR and HTTP_REMOTE_HOST on the Backend Web Server.

6.4.2 REMOTE_USER

IceWall SSO can send the authenticated user ID and password in the basic authentication format with the authorization header. The authenticated user ID can be referenced with the environment variable REMOTE_USER by configuring the basic authentication function on the Backend Web Server.

6.4.3 CLIENT_CERT (SSL_CLIENT_CERT when using Apache as the web server)

When using a Client Certificate between the browser and Forwarder, the certificate information is not transferred to the Backend Web Servers. This certificate information can be retrieved when necessary by configuring the IceWall server in the manner below.

- ○ If Forwarder's web server is Apache HTTP Server, to transfer the client certificate's subject DN:
  HEADER=SSL_CLIENT_S_DN,SSL_CLIENT_S_DN

This certificate information is sent as an HTTP header; therefore, it cannot be used for authentication between Forwarder and the Backend Web Servers. To retrieve as environment variables on the Backend Web Servers, set to “HTTP_SSL_CLIENT_S_DN”.

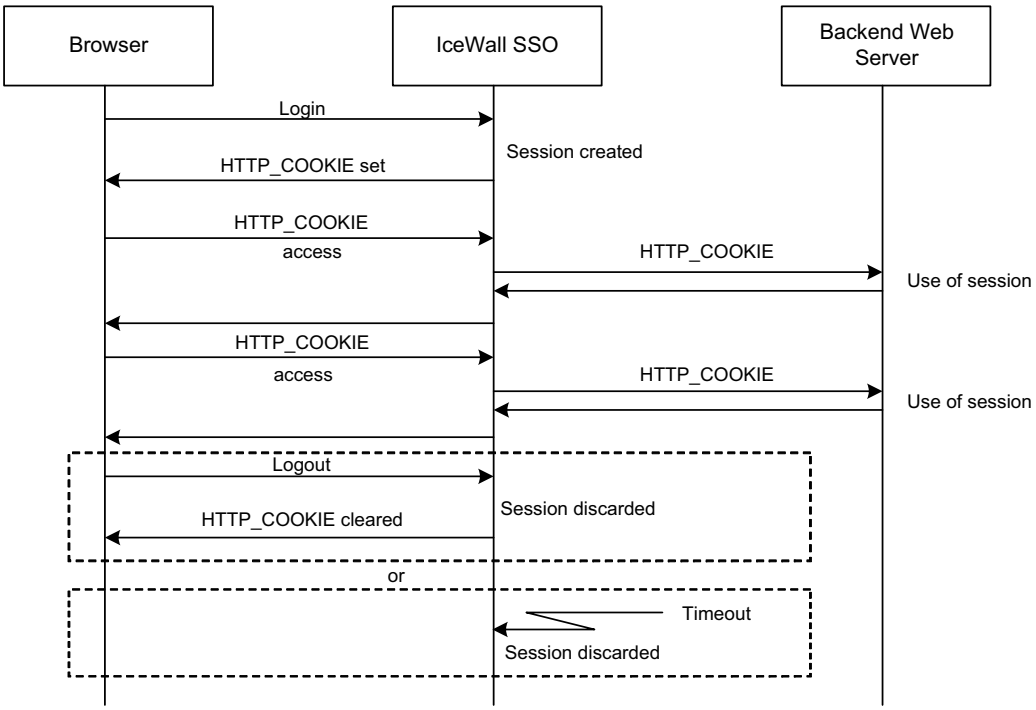
6.5 Session management

IceWall SSO issues a session ID to the logged in user. This session ID is valid from IceWall SSO login until logout. During this period, the same session ID is always sent to the Backend Web Servers. A Backend Web Server application can reference this session ID to determine that this is a process continuing from login. In addition, IceWall SSO also includes a function that automatically logs the user out depending on the time elapsed since last access in the case that a logout operation has not been performed and the user remains logged in.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

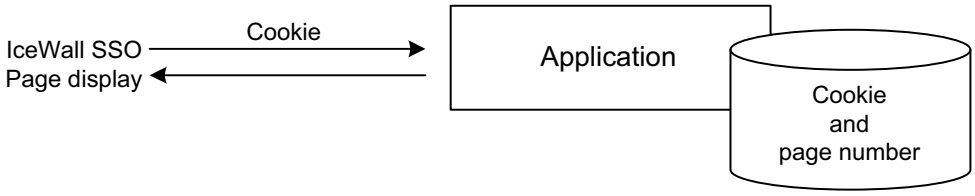
6.5.1 Examples of a session

An example of an IceWall SSO session is described below.



6.5.2 Session management usage example

This describes an example of page transitions using the session ID of IceWall SSO and the page numbers managed by the application. As a prerequisite condition, a table for the management of the pages is created in the database, and the application manages the page transitions.



By managing the IceWall SSO session and page numbers with the above configuration, it can be verified that the pages are flowing in the correct sequence. This can be controlled to display the next page if the pages are flowing in the correct order and return to the first page if they are in the incorrect order.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

6.5.3 Session timeout considerations

Please take into consideration the following points when using the application on the Backend Web Server to manage timeouts separately from IceWall SSO's automatic timeout.

IceWall SSO timeout and application timeout

(1) When managing a timeout in the application on the Backend Web Server, set the application's timeout longer than the IceWall SSO timeout.
(2) When application timeout is shorter than IceWall SSO timeout:
A timeout on the application may occur during IceWall SSO session.
(3) When application timeout is longer than IceWall SSO timeout:
Normally, IceWall SSO timeout occurs prior to application timeout, but depending on the usage, timeout on the application may occur first.

Handling timeouts

(1) When application timeout occurs prior to IceWall SSO timeout:
If a timeout occurs on the application, have the application perform error processing, such as displaying the timeout page.

(2) When IceWall SSO timeout occurs prior to application timeout:
If IceWall SSO timeout occurs, the login page for IceWall SSO authentication appears even if the application is mid-process. The sequence of operations of the application after authentication differs depending on whether the IceWall SSO session ID is referenced or not, as described below.

- When the IceWall SSO session ID is not referenced:
The process performed before authentication can be continued after authentication.
- When the IceWall SSO session ID is referenced:
Because a new session ID is created after authentication, a mismatch of the session ID occurs if continuation of the interrupted process is attempted. Either of the following two methods can be used to prevent this from occurring.
  Method 1: Perform a logout on the application for the old session ID. After the deletion of the old session ID, the top page appears, so as to begin a new session on the application with a new session ID.
  Method 2: Update the old session ID with a new session ID and display the top page. (This is virtually the same as not referencing the session ID.)

*** POST data during a timeout**

After processing the login, IceWall SSO uses the GET method to access the web application. Therefore, if IceWall SSO timeout occurs while processing POST data, the POST data that was being handled on the page immediately before the timeout is destroyed and the IceWall SSO data transmission error page is displayed.
In this situation, the form data must be recovered by entering it again.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

In addition, by using the “POST data inheritance function” with version 8.0 R3 or higher, the POST data sent during IceWall SSO timeout after login to IceWall SSO can be resent. However, please note the following restrictions.

- Content-type must be application/x-www-form-urlencoded.
- The POST data length must be within 1024 bytes.
- A browser that supports JavaScript must be used.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

7 HTML Syntax

Please note the following items when using HTML syntax with IceWall SSO.

7.1 Tag description methods

HTML syntax for use with IceWall SSO is based on the assumption that it conforms with a standard reference (HTML 4.01). There are no particular restrictions when this HTML syntax conforms to a standard reference.

7.1.1 Case sensitivity

There is no case sensitivity when describing tags. In addition, tag names are not case sensitive.

Examples of no case sensitivity

Description containing all upper case letters
```
<A HREF="http://www.hp.com/">Hewlett-Packard</A>
```

Description containing all lower case letters
```
<a href="http://www.hp.com/">Hewlett-Packard</a>
```

Description containing both upper and lower case letters
```
<A HrEf="http://www.hp.com/">Hewlett-Packard</a>
```

7.1.2 Attribute values

Tags operate regardless of whether there is a space or tab before and after the equal symbol (“=”) connecting the attribute name and value within the tag. In addition, double quotes (or single quotes) are not required for the attribute value, as shown in the following examples.

Examples of anchor tags

No space or tab before and after the equal symbol (“=”) connecting the attribute name and value
```
<a href="http://www.hp.com/">Hewlett-Packard</a>
```

A space/tab placed before/after the equal symbol (“=”) connecting the attribute name and value
```
<a href = "http://www.hp.com/">Hewlett-Packard</a>
<a href= "http://www.hp.com/">Hewlett-Packard</a>
<a href ="http://www.hp.com/">Hewlett-Packard</a>
```

The attribute value is not enclosed by double quotes
```
<a href=http://www.hp.com/>Hewlett-Packard</a>
```


(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

7.1.3 Character encoding

Shift_JIS, EUC, and Unicode (UTF-8) are supported as the character encoding for the content on the Backend Web Servers. However, when using content that uses non-ASCII character encodings such as Japanese, you must be aware of the following items.

(1) Tags must use ASCII characters.
(2) If double byte characters are to be excluded from keyword conversion and URL conversion, these characters can pass through IceWall SSO as is.
(3) Keyword conversion can also be used with multi-byte characters that are not supported. However, in this case, characters to be converted defined in the configuration file must be encoded with the target character code.
(4) Strings in which NULL characters appear cannot be used. For example, for character encodings such as Unicode in which NULL characters are automatically inserted when using single byte characters, single byte characters cannot be used.

7.2 Descriptions of unrecognizable tags

Some tags that are commonly recognized by the browser may not be recognized by IceWall SSO.

7.2.1 Tag names including spaces, tabs, or line feeds

Spaces, tabs, and line feeds within HTML tags are usually ignored (with the exception of <PRE> tags), but because IceWall SSO searches for and recognizes " '<' + 'tag name' " as a single string, it cannot recognize tag names including spaces, tabs, or line feeds.

1. Space, tab, or line feed in a tag name

```
<i mg src="title.gif">
<i
mg src="title.gif">
```

2. Space, tab, or line feed between "<" and the tag name

```
< a href="http://www.hp.com/">Hewlett-Packard</a>
<
a href="http://www.xxx.hp.com/">Hewlett-Packard</a>
```

7.2.2 Attribute names including spaces, tabs, or line feeds

Because IceWall SSO searches for and recognizes attribute names as single strings, it cannot recognize attribute names including spaces, tabs, or line feeds.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

Examples of attribute names including spaces, tabs, or line feeds

```
<a hr ef="http://www.hp.com/">Hewlett-Packard</a>
<a hr
ef="http://www.hp.com/">Hewlett-Packard</a>
<img sr c="title.gif">
<img s
rc="title.gif">
```

7.2.3 Missing space, tab, or line feed between a tag name and an attribute name

When there is no space, tab, or line feed between a tag name and an attribute name, that attribute name is not recognized.

Example of a connected tag name and attribute name

```
<a href="http://www.hp.com/">Hewlett-Packard</a>
<imgsrc="title.gif">
```

7.2.4 Missing space, tab, or line feed between an attribute value and an attribute name

When there is no space, tab or line feed between an attribute name and the value of another attribute, the attribute names are not recognized.

Example of a connected attribute value and attribute name

```
<body bgcolor="#000000"background="title.gif">
<img alt="Title Logo"src="logo.gif">
```


(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

8 URL Conversion

The IceWall SSO URL conversion function is described below.

8.1 Targets for URL conversion

IceWall SSO uses the URL conversion function to convert URLs described in HTML syntax tags received from a Backend Web Server into URLs accessible via IceWall SSO. IceWall SSO also converts the URLs in the HTTP Location header (see “6.2.1 Headers requiring special attention” for details) and Set-Cookie header (see “13.4 When the path attribute is specified in a cookie” for details) in the same way.

The content subject to conversion is configured by MIME type (example: text/html) in the IceWall server configuration file. Therefore, if the Content-type header is not sent, the content is not a target for conversion (the Location header and Set-Cookie header are converted).
Since URL conversion is performed by string substitution, if the types of content below are specified as a conversion target, the URLs may not be correctly converted or the content may be damaged.

- Content where text and binary are mixed (example: MIME multipart)
- Content that is HTTP compressed
- Binary files such as images and PDF files

As noted, URL conversion is a function which targets HTML and XHTML contents in text format. Further sections explain the conversion rules for URLs contained in HTML contents.

8.2 Conversion targets for URL conversion

The URL conversion function treats the value defined in the tag name and attribute name (URLKEY parameter) configured in the IceWall server configuration file as a URL and converts it. For tags including items other than URLs or URLs described outside of tags, the keyword conversion function described later must be used.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

Note: The URLKEY parameters initially configured in the IceWall server configuration file are as follows.

```
URLKEY=A,HREF
URLKEY=BASE,HREF
URLKEY=FRAME,SRC
URLKEY=FORM,ACTION
URLKEY=BASE,TARGET
URLKEY=IMG,SRC
URLKEY=SCRIPT,SRC
URLKEY=BODY,BACKGROUND
URLKEY=TD,BACKGROUND
URLKEY=TR,BACKGROUND
URLKEY=TABLE,BACKGROUND
URLKEY=APPLET,CODEBASE
URLKEY=INPUT,SRC
URLKEY=LINK,HREF
```

Complete matching is performed when searching for tag names in content, but the comparison method for tag attribute names differs according to the URLCONV_ATTR_FLG parameter setting. (10.0)

(1) When URLCONV_ATTR_FLG=1
- The tag attribute name comparison method is complete match.

(2) When URLCONV_ATTR_FLG=0
- The tag attribute name comparison method is prefix match.

8.2.1 URLs converted automatically

IceWall SSO automatically converts the two types of paths (URLs) below into URLs accessible via IceWall SSO using the URL conversion function.

(1) Absolute paths starting with “http” (including “https” when using the SSL Option)
Absolute path URLs including host names configured as Backend Web Server names in the IceWall server are automatically converted. Absolute paths to Backend Web Servers not configured are not converted.

Example of when the HOST parameter of dfw.conf is set to HP=www.hp.com
(IceWall SSO Forwarder path: /fw/dfw)

When the content of the backend web server contains the tag
link to Hewlett-Packard
the URL conversion function is used to automatically convert this to the following URL:
link to Hewlett-Packard

In the following example, the HOST parameter is not configured and, therefore, the URL is not converted.
link to XYZ Corp
In this case, by adding a setting such as XYZCorp=www.xyz.co.jp to the HOST parameter, the URL is automatically converted.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

(2) Absolute paths from the document root starting with "/"
URLs specified with the absolute path from the document root are automatically converted to URLs for Backend Web Servers with content including those URLs.

Example of when the HOST parameter of dfw.conf is set to HP=www.hp.com
(IceWall SSO Forwarder path: /fw/dfw)

When www.hp.com is accessed via IceWall SSO and the content retrieved contains the tag
Notice
the URL conversion function is used to automatically convert this to the following URL:
Notice

8.2.2 URLs not requiring conversion

Depending on how the URL is described, conversion via the URL conversion function may not be necessary. Below are the two types of paths (URLs) that do not require conversion.

(1) Relative paths from the current directory (../, etc.)
When a link to a URL described by the relative path from the current directory is selected in the browser, a URL is created and accessed on the browser based on the position of the content that the browser is currently referencing. Therefore, when displaying content accessed via IceWall SSO, a URL accessible via IceWall SSO is created on the browser and the target link is accessed.

Example of when the browser is referencing the URL http://host.co.jp/fw/dfw/HP/data1/

The following type of URL is described in the displayed content:
Data2 notice
When this link is then selected in the browser, the URL is created on the browser, as shown below, and accessed.
http://host.co.jp/fw/dfw/HP/data2/info/index.html

(2) File name only (index.html, etc.)
When there is a link only to the file name, as described in "(1) Relative paths from the current directory (../, etc.)" above, no URL conversion is necessary because, when that link is selected, a path is created on the browser that should provide access.

8.2.3 URLs not converted even if configured for conversion

URLs that cause a malfunction if converted by the URL conversion function are not converted. Please note that if URL conversion is not performed then the actual URL may be displayed in the browser.
Unconverted paths (URLs) are as follows.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

(1) URLs already converted (/fw/dfw/sys/index.html, etc.)
When a URL that is accessible via IceWall SSO is included in the content retrieved from the Backend Web Server, URL conversion is not performed. However, note that conversion is performed when the Forwarder path differs from the actual operating Forwarder path.

Example of when the HOST parameter of dfw.conf is set to HP=www.hp.com
(IceWall SSO: host.co.jp, Forwarder path: /fw/dfw)

When the content of the backend web server contains the tag
Data1 notice
URL conversion is not necessary because it is already determined that this is the URL accessible via IceWall SSO.
However, if part of the Forwarder path differs from the actual operating path, as in
Data1 notice
the URL is converted to an unintended URL, as in
Data1 notice

When a URL accessible via IceWall SSO is described as a specified absolute path, as shown below, the URL is not converted because the URL host name is not defined in the HOST parameter of dfw.conf.
Data1 notice

(2) URLs in query strings
URL conversion is not performed even if a Backend Web Server's URL is present inside the query string.
If you need to convert these URLs, you must use keyword conversion.

8.2.4 URLs that cannot be converted

Depending on the description method of the URL, automatic conversion via URL conversion may not work. In this case, convert the URL using the keyword conversion function described later.

- JavaScript event handler specification (OnClick="location.replace('/index.html')", etc.)
  The URL conversion function automatically converts absolute paths starting with “http://” or relative paths from the document root starting with “/”. However, all paths (URLs) that start with other characters are recognized as relative paths from the current directory or paths specified by file name.
- Inconsistency between setting and actual URL
  When the HOST parameter in the Forwarder configuration file (dfw.conf) is defined as an IP address and the URL in the content is written as a host name, URL conversion is not performed because Forwarder cannot resolve the name.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

8.2.5 URLs for extraction

The following types of URLs are extracted as targets for URL conversion in IceWall SSO according to the URLCONV_FLG parameter and ATTRQUOT_FLG parameter settings.

(1) When URLCONV_FLG=0

- URLs enclosed with double quotes
  (Example: <a href="www.svr.com/">)
- URL with double quotes either before or after
  (Example: <a href="www.svr.com/> <a href=www.svr.com/">)
- URLs enclosed with single quotes
  (Example: <a href='www.svr.com/'>)
- URL with single quotes either before or after
  (Example: <a href='www.svr.com/> <a href=www.svr.com/'>)
- URL enclosed with a combination of double quotes and single quotes
  (Example: <a href="www.svr.com/'> <a href='www.svr.com/">)
- URLs without double quotes or single quotes
  (Example: <a href=www.svr.com/>)

(2) When URLCONV_FLG=1, ATTRQUOT_FLG=0

- URLs enclosed with double quotes
  (Example: <a href="www.svr.com/">)
- URLs enclosed with single quotes
  (Example: <a href='www.svr.com/'>)
- URL enclosed with a combination of double quotes and single quotes
  (Example: <a href="www.svr.com/'> <a href='www.svr.com/">)
- URLs without double quotes or single quotes
  (Example: <a href=www.svr.com/>)
- URLs defined with brackets ("<" and ">") in the attribute value
  (Example: <a href="/index.cgi?name=aaaaa>bbbbb">)

(3) When URLCONV_FLG=1, ATTQUOT_FLG=1

- URLs enclosed with double quotes
  (Example: <a href="www.svr.com/">)
- URLs enclosed with single quotes
  (Example: <a href='www.svr.com/'>)
- URLs without double quotes or single quotes
  (Example: <a href=www.svr.com/>)
- URLs defined with brackets ("<" and ">") in the attribute value
  (Example: <a href="/index.cgi?name=aaaaa>bbbbb">)

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

8.2.6 URL conversion of multiple attributes with different attribute names in a single tag 10.0

The URLs in multiple attributes with different attribute names in a single tag can be converted by setting the URLCONV_ATTR_FLG parameter to 1.

Example: To convert multiple attributes with different attribute names in a single tag
(Forwarder path: /fw/dfw)

Perform the following setting.
Forwarder configuration file
HOST=HP=www.hp.com
Host configuration file
URLCONV_ATTR_FLG=1
URLKEY=INPUT,VALUE1
URLKEY=INPUT,VALUE2

When the content of the backend web server contains the tag
<input value1="http://www.hp.com/index.html" value2="http://www.hp.com/index.html">
the value1 and value2 are converted to the following URL:
<input value1="/fw/dfw/HP/index.html" value2="/fw/dfw/HP/index.html">

8.3 URL description restrictions

The following are the restrictions on the descriptions of URLs to be converted by IceWall SSO.

(1) Description for an identical host name must be the same.
(2) A path tracing back the document root must not be described.
(3) The maximum number of characters in a URL (including query strings) is 512 bytes.
(4) URLs containing only a host name must end with a slash (“/”).
(5) When HTML syntax is expressed with a percent sign (“%”), that URL is not subject to URL conversion.
Example: When <a href=... > is expressed as <a%20href=... >

8.3.1 Standardization of host name descriptions

As part of the IceWall SSO specification, URL conversion of host names uses string pattern matching. When descriptions exist for both the full domain name and host name only, or if the host name is described in both upper and lower case letters, the host name is recognized as a separate host. Each of the host names must be set in the configuration file.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

1. Converting a host name with both upper and lower case letters to an alias name

When the configuration file definition is:

HOST=HP=www.hp.com
(host name: www.hp.com, alias name: HP)

Then:

(1) "http://www.hp.com/" can be used as an alias.
(2) "http://WWW.HP.COM/" cannot be used as an alias.

To set (2) as the same alias name as (1), perform the two-line configuration below.

HOST=HP=www.hp.com
HOST=HP=WWW.HP.COM
(host name: www.hp.com, alias name: HP
host name: www.hp.com, alias name: HP)

2. Converting a host name with only itself or a full domain name to an alias name

When the configuration file definition is:

HOST=HP=www.HP.com
(host name: www.hp.com, alias name: HP)

Then:

(1) "http://www.hp.com/" can be used as an alias.
(2) "http://www/" cannot be used as an alias.

To set (2) as the same alias name as (1), perform the two-line configuration below.

HOST=HP=www.HP.com
HOST=HP=www
(host name: www.hp.com, alias name: XXXCorp
host name: www, alias name: XXXCorp)

If both a description using the host name and a description using the IP address are used, the host information must be additionally defined in the same manner.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

8.3.2 Path tracing back the document root

When the relative path from the current directory is output to the browser via IceWall SSO, a URL accessible via IceWall SSO is automatically created on the browser based on the accessed URL. However, in the case a path is tracing back the document root, the path to the host alias name or to IceWall SSO is not correctly created.

Example of when a path tracing back the document root is specified in the URL
(Forwarder path: /fw/dfw, alias name: HP)

When an image file (../../../image/title.gif) is specified in "/data/info1/index.html," this can be accessed via the IceWall server as follows.
/fw/dfw/HP/data/info1/../../../image/title.gif

However, on the browser, this is recognized as a relative path and accessed with the following path.
/fw/dfw/image/title.gif
As a result, "title.gif" does not appear.

When creating new content, describe the URL without specifying the path tracing back the document root. When existing content cannot be edited, use the keyword conversion function, described later, to convert the path.

8.3.3 URLs not starting with "http" or "/"

When a URL starting with "http" (or "https") or "/" directly after the attribute name is not specified and URL conversion is necessary, the keyword conversion settings, described later, must be performed.

Example of when a URL not starting with "http" or "/" is defined
(Forwarder path: /fw/dfw, alias name: HP)

In the tag
<metat http-equiv="Refresh" content="10;url=http://www.hp.com">
"10;http://www.hp.com" is not recognized as a URL, but this can be converted as an exception with the following setting.
URLKEY=META,CONTENT

Keyword conversion is necessary for URLs described above, with the exception of <meta> tags.

8.3.4 URLs described in comments

Even when a tag is described in a comment tag, if that tag is subject to conversion, IceWall SSO performs URL conversion.

Example of when a URL is described in a comment tag
(Forwarder path: /fw/dfw, alias name: HP)

If the URL is described as

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

<!--<a href="http://www.hp.com/">-->
conversion is performed regardless of whether there is a comment.
<!-- <a href="/fw/dfw/HP/">-->
The URL is converted as shown above and treated as a comment by the browser.

8.3.5 URL conversion when using BASE tags

When a <base> tag is described in the web content, all URLs in the content are regarded as existing on the Backend Web Server specified with <base> tags, and IceWall SSO converts the URLs accordingly. However, URLs described as absolute paths including host names are not affected.

Example of URL conversion of content in which a <BASE> tag is described
(Forwarder path: /fw/dfw, alias name: HP, IceWall server name: icewall.hp.com)

URLs in content existing on the backend web server [www.hp.com] are normally converted as follows:
<a href="http://www.hp.com/index.html">
After conversion: <a href="/fw/dfw/HP/index.html">
<a href="/index.html">
After conversion: <a href="/fw/dfw/HP/index.html">

When the description <base href="http://www.xyz.co.jp/"> (alias name is XYZCorp) is added to this content, all URLs are converted as follows:
<base href="http://www.xyz.co.jp/">
After conversion: <base href="http://icewall.hp.com/fw/dfw/XYZCorp/">
<a href="http://www.hp.com/index.html">
After conversion: <a href="/fw/dfw/HP/index.html">(not affected)
<a href="/index.html">
After conversion: <a href="/fw/dfw/XYZCorp/index.html">(affected)

When a <BASE> tag is described in a comment tag, as described in “8.3.4 URLs described in comments,” conversion is performed in the same manner as above.

8.3.6 Using a separator between the URL and argument (query string)

Use a question mark (“?”) as a separator between the URL and query string.
In the case a web application server as a Backend Web Server supports “;” as the separator (such as a servlet session ID), because IceWall SSO does not recognize “;” as the separator, it is URL-encoded and then sent. For a web application server which cannot recognize a URL-encoded “;” as a separator, configure (URL_SCOLON=1) so as not to convert “;” on the IceWall server.

8.3.7 Restrictions when writing absolute URLs in content on the Backend Web Server

Forwarder converts absolute URLs included in content with the URL conversion function, but the function has the following restrictions when writing absolute URLs in IPv6.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

- Square brackets are required “[~]” when writing the host as an IPv6-formatted IP address.

http://**[2001:0db8:0:0:0:0:0:1]**/index.html
http://**[2001:0db8:0:0:0:0:0:1]**:81/index.html

The IP addresses below are IPv6-formatted, but they cannot be compared normally since they have no square brackets “[~]” and URL conversion is not performed.

http://**2001:0db8:0:0:0:0:0:1**/index.html
http://**2001:0db8:0:0:0:0:0:1**:81/index.html

- Square brackets are not required “[~]” when writing the host as an IPv6 host name.

  Square brackets "[~]" are not added to IPv6 host names, the same as IPv4. If square brackets "[~]" are added to the host name, the host name cannot be compared normally and URL conversion is not performed.

http://**www.iwdfw.com**/index.html
http://**www.iwdfw.com**:81/index.html

8.4 Java applets

8.4.1 Downloading an applet

Java applets can be downloaded via IceWall SSO and operated without any problems, but the URLs held in the applets are not modified. If the URLs held in the applets are absolute paths, they require conversion. You must correct the URLs in the applets so they are URLs accessible via IceWall SSO.

8.4.2 Communication from an applet via IceWall SSO

Other documents can be accessed via IceWall SSO by creating a URL that combines “getDocumentBase()” and a relative path.

Example:
① URL url = new URL(getDocumentBase(), "sample.htm");
② URL url = new URL(getDocumentBase(), "../test/sample.htm");

8.5 ActiveX

8.5.1 Downloading ActiveX

ActiveX can be downloaded via IceWall SSO and operated without any problems, but the URLs held in the controls are not modified. If the URLs held in the controls are

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

absolute paths, they require conversion. You must correct the URLs in the controls so they are URLs accessible via IceWall SSO.

8.6 Other files

8.6.1 Downloading other files

Files other than Java applets and ActiveX can also be downloaded via IceWall and operated without any problems, but please be aware of the following points.

- Just as with Java applets and ActiveX controls, if rich media such as Flash contains embedded URLs with absolute paths, then you must change them to URLs accessible via IceWall SSO.
- The MIME configuration of the target file must be set up on the Backend Web Server where the downloaded file is installed. (The MIME setting of files for which MIME configuration is not set up is usually recognized as text/plain.)

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

9 Keyword Conversion

The IceWall SSO keyword conversion function is described below.

9.1 Remarks on the keyword conversion function

There are several important points to keep in mind when using the keyword conversion function.

9.1.1 Conversion target

The only target for keyword conversion is the response from the Backend Web Server. Keyword conversion cannot be used on requests sent to the Backend Web Server. As with URL conversion, the subject to conversion is only the content of the MIME type specified in the IceWall server configuration file. Content is not subject to conversion when the Content-type header is not sent, as well as for content compressed by zip, PDF files, binary files such as image files, and MIME multipart content.

The keyword conversion function can also be used for the entire response subject to conversion (both the HTTP headers and content). The conversion configuration can be specified for all Backend Web Servers or for individual Backend Web Servers.

9.1.2 Case sensitivity

The keyword conversion function uses case sensitivity in search strings. For this reason, when defining keywords, the search string must be configured so that the case matches that of the original string.

9.1.3 Using unique search strings

When defining search strings in the content or HTTP headers, search strings must be made unique for each Backend Web Server so that keyword conversion is performed only for the appropriate locations. When only the smallest part of a search string is defined, the keyword conversion function may also affect unexpected areas of the search string.

Example of the definition of a search string
(Forwarder path: /fw/dfw, alias name: HP)

When the content is as follows:

```
<html>
<head>
<title>This is http://www.hp.com</title>
</head>
<body>
<meta http-equiv="Refresh" content="10;URL=http://www.hp.com/index2.html">
</body>
```


(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

</html>
and the definition for keyword conversion is as follows:
REPKEY=http://www.hp.com,/fw/dfw/HP/
(Search string: http://www.hp.com
Substitution string: /fw/dfw/HP/)
the text string in the <meta> tag as well as the text string in the <title> tag is converted.
To prevent this from occurring, define a unique search string while including text before and after the keyword, as shown below.
REPKEY=10;url=http://www.hp.com, 10;url=/fw/dfw/HP/
(Search string: 10;URL=http://www.hp.com
substitution string: 10;URL=/fw/dfw/HP/)

9.1.4 Creating reserved words from search strings

Search strings defined for keyword conversion are treated as reserved words in each Backend Web Server. For this reason, when creating new content, it is important that the content does not contain those strings, else the search strings must be redefined.

9.1.5 Character encoding

Shift_JIS, EUC, and Unicode (UTF-8) are supported as the character encoding of content on the Backend Web Servers. However, when using keyword conversion to convert double byte characters such as Japanese characters, the character encoding of the content must be set.

9.1.6 Unconvertible strings 10.0

In keyword conversion, the comma "," is reserved as the REPKEY parameter separator, so it cannot be set. The control codes "\r", "\n", and "\t" also cannot be set. When you wish to set these characters as keywords, set each as a character string "\,", "\r", "\n", and "\t" in the REPKEY_EXT parameter.

9.2 Specific keyword conversion function

Keyword conversion has two types of conversion functions: normal keyword conversion, which defines the desired keywords to convert in the configuration file, and specific keyword conversion.

9.2.1 Specific keywords

These are keywords that have been reserved in advance by IceWall SSO and are converted at execution to the IceWall SSO operating environment or the information of the user logging in.

9.2.2 Conversion sequence

The conversion sequence of the conversion functions available with IceWall SSO are as follows:

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

(1) Conversion of URLs in HTTP headers
(2) Keyword conversion function (REPKEY parameters of the host configuration file)
(3) Extended keyword conversion function (REPKEY_EXIT parameters of the host configuration file)
(4) URL conversion function (URLKEY parameters of the host configuration file)
(5) Specific keyword conversion function

9.2.3 List of specific keywords 10.0

The specific keywords that can be defined are as follows:

Keyword	Description of conversion
$USER_ID	This is replaced by the user ID of the user logging in. When using the Client Certificates Option, this is replaced, even before login, with the user ID of the user that accessed IceWall SSO.
$PASSWORD	This is replaced by the password of the user logging in. This is the value entered at the time of login, not the encrypted password stored in the Authentication DB.
$DFW	This is replaced by the Forwarder path. Example: /fw/dfw However, for the session management using URL-Cookie, it is replaced as follows. At password change: Example: /fw/dfw/xxx–xxx/[password change alias] At logout: Example: /fw/dfw/xxx–xxx/[logout alias]
$REQUEST_URL	This is replaced by the path requested by the browser. Example of the session management using Cookie: /fw/dfw/ALIAS/index.html Example of the session management using URL-Cookie: /fw/dfw/xxx–xxx/ALIAS/index.html However, arguments added to the URL's end (query string) are not included.
$HIDEURL	This is replaced by the path requested by the browser but with the Forwarder part omitted. Example of the session management using Cookie: /ALIAS/index.html Example of the session management using URL-Cookie: /xxx–xxx/ALIAS/index.html This keyword includes arguments added to the URL's end (query string).
$ALIAS	This is replaced by the alias name of the Backend Web Server from which the content was retrieved.
$IW_INFO	This is replaced by the session ID of the logged in user. This is only the authentication information part included in the authentication cookie.
$KEY_LOGIN	This is replaced with a string that is the key for determining login processing.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

Keyword	Description of conversion
$KEY_LOGOUT	This is replaced with a string that is the key for determining logout processing.
$KEY_PWDCHG	This is replaced with a string that is the key for determining password change processing.
$PWDCHG_URL	This is replaced with a URL for displaying the password change page.
$LOGOUT_URL	This is replaced with a URL for displaying the logout page.
$TOPPAGE_URL 10.0	This is replaced by the URL to display the top page.
$PWDHIDEURL 10.0	This is replaced by the URL to take over the URL information to return to Agent when changing passwords linked with Agent.
$IWTID 10.0	This is replaced by the transaction ID for the user that accessed the server. This keyword is only replaced when the TRANSID parameter is ON.

9.2.4 Usage examples of specific keywords

The following is an example of using the keyword conversion function with the session management using URL-Cookie (SESSION=1) to convert to URLs accessible via the IceWall server.

Examples of the keyword conversion settings for the session management using URL-Cookie

Keyword conversion is necessary when JavaScript contains the following:

```
location.url = 'http://www.hp.com/info/index.html';
```

The following setting is used for normal Cookie session management (SESSION=0 in dfw.conf):
(When the Forwarder path is /fw/dfw and the alias name of www.hp.com is HP)

```
REPKEY='http://www.hp.com, '/fw/dfw/HP
```

In the case of the session management using URL-Cookie, the session ID is included in the URL. Because the value of the session ID differs for each login and a fixed value cannot be specified, a specific keyword is used.

```
REPKEY='http://www.hp.com, '/fw/dfw/$IW_INFO/HP
```

With this setting, the $IW_INFO part is replaced with the session ID of the user logging in at execution.

The following is an example of conversion to a URL accessible via the IceWall server and not dependent on the IceWall SSO operating environment (for example, when the CGI prefix differs for the development environment and the actual environment).

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

Examples of the keyword conversion settings not dependent on the operating environment

Keyword conversion is necessary when JavaScript contains the following:

```
var url = '/mail/list.html';
```

Normally, when configuring the keyword settings, the Forwarder path name and alias name are directly specified as follows, depending on the operating environment: (When the Forwarder path is /fw/dfw and the alias name is HP)

REPKEY=var url = '/,var url = **'/fw/dfw/HP/**

When the Forwarder path name and alias name are changed, this keyword setting must also be changed. A universal keyword setting can be made by using the specific keywords $DFW and $ALIAS here.

REPKEY=var url = '/,var url = **'$DFW/$ALIAS/**

With this setting, the keywords are replaced with the value of the environment running $DFW and $ALIAS.
When using these specific keywords with the session management using URL-Cookie, the setting is as follows:

REPKEY=var url = '/,var url = **'$DFW/$IW_INFO/$ALIAS/**

9.2.5 Restrictions

When a setting is made to convert a text string reserved as a specific keyword to another value with the keyword conversion function, that keyword no longer functions as a specific keyword.

When the following setting is made, the keyword no longer functions as a specific keyword.
Example: REPKEY=$USER_ID,XXX

A function that prevents the misuse of specific keywords has been included in IceWall SSO since version 8.0 as one of the cross site scripting functions. Please be aware that when this function is used, the keyword is not converted to a value and remains in the content unchanged.

Example:
To disallow all specific keyword conversions in the backend web server's content:
CTRL_SPKEY=

To allow only the conversion of $USER_ID and $ALIAS in the backend web server's content:
CTRL_SPKEY=$USER_ID,$ALIAS

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

10 JavaScript

Please note the following items when using JavaScript with IceWall SSO.

10.1 Client Side JavaScript

When using Client Side JavaScript, there are no particular restrictions if the syntax conforms to the standard reference. However, if there is a need to control the URLs within the script, these URLs must be converted to URLs accessible via IceWall SSO by using the URL conversion function or the keyword conversion function.

There are two types of converted URLs, depending on the description method within the script.

(1) URLs in tags output by scripts
(2) URLs used as arguments of a function

10.1.1 URLs in tags output by script

When content is dynamically output from a script, the content is treated as being subject to the URL conversion function if the output is in tag format.
However, it is important to note that if the attribute value of the output tag starts with an escape character (\), that attribute value is determined to be the relative path from the current directory and is not subject to conversion.

Examples of tags output by script
(Forwarder path: /fw/dfw, alias name: HP)

If, for example, the script in the document contains
```
document.write('<a href="http://www.hp.com/">Hewlett-Packard</a>');
```
the output tag part is subject to URL conversion and is converted as follows before the script is executed on the client side.
```
document.write('<a href="/fw/dfw/HP/">Hewlett-Packard</a>');
```

If, for example, the attribute value of the tag uses escape characters, as in
```
document.write('<a href=\"http://www.hp.com/\">Hewlett-Packard</a>');
```
this is not converted because it is determined to be a relative path from the current directory.

10.1.2 URLs used as arguments of a function

When URLs are used as arguments of a function, these URLs are not subject to conversion with the URL conversion function. Therefore, the keyword conversion function must be used.

Examples of URLs specified as arguments of a function
(Forwarder path: /fw/dfw, alias name: HP)

If, for example, a tag (e.g., anchor) contains
```
<a href="JavaScript:Func01('/index.html')">Hewlett-Packard</a>
```


(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

the following additional settings must be made as keyword conversion parameters.
REPKEY=Func01('/index.html'),Func01('/fw/dfw/HP/index.html')
(Search string: Func01('/index.html')
Substitution string: Func01('/fw/dfw/HP/index.html'))

Further, if OnClick(), etc. is used to separately create a function to be called and an argument is hard-coded within that function, the keyword conversion function must be used as described above.
When the description is as follows:

```
<script language=JavaScript>
  function Func01() {
  location.replace('/index.html');

</SCRIPT>
```

the following additional settings must be made as keyword conversion parameters.
(Search string: location.replace('/index.html')
Substitution string: location.replace('/fw/dfw/HP/index.html'))

10.1.3 Notes when using the cross site scripting prevention filter

When using the cross site scripting prevention filter for content from the Backend Web Server, the start/end of the tag may be falsely recognized because of inequality signs written in the script. In this kind of situation, please add the opposite inequality sign for those inequality signs to the script in a comment so that the numbers of "<" and ">" are made equal.

Example of when the number of inequality signs is unequal

For the following script:

```
<script language=JavaScript>
  •
  for( i = 0; i < 10; i++ ){
    •
  }
</script>
```

Since the numbers of "<" and ">" are not equal, the tags after the for() statement may not be correctly recognized. Add the following comment in this kind of situation.

```
<script language=JavaScript>
  •
  for( i = 0; i < 10; i++ ){
  •
  }
  <!-- > -->
</script>
```

10.2 Server Side JavaScript

When using Server Side JavaScript, the script syntax itself is not affected by IceWall SSO. If the HTML syntax of the content output from the script and Client Side JavaScript syntax conform with a reference, the parameters to be checked are only those parameters described in each mentioned chapter.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

10.2.1 Server objects

The name of the web server operating Server Side JavaScript is usually stored in the host name properties of the server objects. However, if the content is accessed via IceWall SSO, the host name of the Backend Web Server (IP address) and the port name are stored in the format “[host name (IP address)]:[port number].”

Since this is part of the IceWall SSO specification, avoid writing scripts that are dependent on this property.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

11 VBScript

Please note the following items when using VBScript with IceWall SSO.

11.1 Remarks about using VBScript

11.1.1 URLs in tags output by script

When content is dynamically output from a script, the content is treated as being subject to URL conversion if the output is in tag format.
However, it is important to note that if the attribute value of the output tag starts with an escape character (\), that attribute value is determined to be the relative path from the current directory and is not subject to conversion.

11.1.2 Notes when using the cross site scripting prevention filter

When using the cross site scripting prevention filter for content from the Backend Web Server, the start/end of the tag may be falsely recognized because of inequality signs written in the script. In this kind of situation, please add the opposite inequality sign for those inequality signs to the script in a comment so that the numbers of "<" and ">" are made equal.

Example of when the number of inequality signs is unequal

For the following script:

```
<script language=VBScript>
    •
    for (i = 0;i<10; i++){
        •
    }
</script>
```

Since the numbers of "<" and ">" are not equal, the tags after the for() statement may not be correctly recognized. Add the following comment in this kind of situation.

```
<script language=VBScript>
    •
    for (i = 0;i<10; i++){
        •
    }
    <!-- > -->
</script>
```

11.2 Active Server Pages (ASP)

11.2.1 Request object

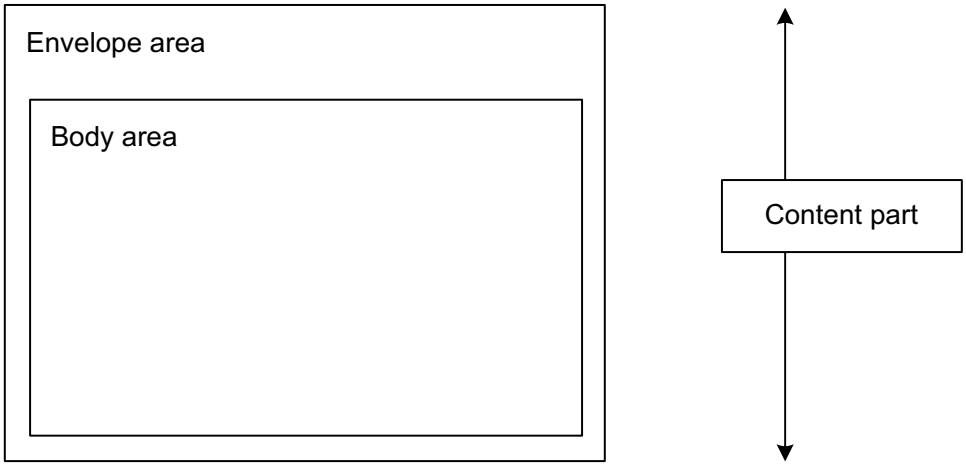
The name of the Backend Web Server operating the Request object is usually stored in the ServerVariables collection of the Request object. However, if the content is accessed via IceWall SSO, the host name of the Backend Web Server (IP address) and the port name are stored in the format "[host name (IP address)]:[port number]." Avoid including scripts that are dependent on this collection.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

12 XML

IceWall SSO treats XML data as binary data like that of images, etc.

XML data format
Content-type: text/xml



The encoder URL described in the envelope area is not subject to the IceWall SSO URL conversion and, therefore, is transferred as is. Even when a URL is included in the body data, the URL is not subject to conversion and, therefore, is transferred as is.

However, if the XML file used is entirely text, you can convert URLs using the keyword conversion function by configuring the XML file MIME type in the IceWall server configuration file.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

13 Using Cookies Configured by a Backend Web Server

In IceWall SSO, cookies configured by a Backend Web Server are handled as follows.

13.1 When the domain attribute is not specified in a cookie

When the domain attribute is not specified, the browser regards the “path under the IceWall SSO alias name” as the effective range of the cookie. Therefore, the cookie is only sent to the Backend Web Server that issued it. (It is not sent to other Backend Web Servers.)

13.2 When the domain attribute is specified in a cookie

IceWall SSO does not include a function to automatically convert a cookie's domain attribute. Keyword conversion must be used to convert the domain attribute to the IceWall SSO domain in order to send the cookie to the Backend Web Server.

13.3 When the path attribute is not specified in a cookie

When the path attribute is not specified in a cookie, the browser stores the path at the time the cookie is set and sends the cookie only when accessing paths specified under that path.

Example of operations when the path attribute of a cookie is not specified
(Forwarder path: /fw/dfw, alias name: HP)

The backend web server program that attempts to set the cookie uses the following URL:

/foobar/set_cookie.cgi

When this path is accessed and the cookie is set, the browser recognizes that the cookie is set with the following path.

/fw/dfw/HP/foobar/set_cookie.cgi

This cookie is sent to the web server only when “/fw/dfw/HP/foobar/” is included at the beginning of the path. When the accessed path is as follows, the issued cookie is not sent to the web server.

/fw/dfw/HP/cgi-bin/... (the area after “/fw/dfw/HP/” does not match)
/fw/dfw/XYZCorp/ (the area after “/fw/dfw/” does not match)

13.4 When the path attribute is specified in a cookie

When the path attribute is specified in a cookie, IceWall SSO automatically converts this to a path on IceWall SSO.
For example, if path=/foobar is set, IceWall SSO converts this as follows and sends it to the browser.

path=/fw/dfw/HP/foobar

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

However, if the attribute value is enclosed in double quotes, it is converted as follows.
Example when set to path="/foobar"
path=/fw/dfw/HP"/foobar"
In this kind of situation, handle the conversion using the keyword conversion function.

13.5 When the secure attribute is specified in a cookie

IceWall SSO does not have a function to automatically convert a cookie's secure attribute. For this reason, the keyword conversion function must be used to delete the secure attribute when communicating between the browser and Forwarder via the HTTP protocol. Keyword conversion is not necessary when communicating between the browser and Forwarder via the HTTPS (SSL) protocol.

13.6 Limits of usable cookies

The maximum number of cookies a single browser can save differs depending on the type of browser. IceWall SSO uses one cookie for session management, so the number of usable cookies in the application is the browser's maximum number - 1. If a cookie over the maximum number is used, the oldest cookie is deleted, which means the IceWall SSO session management cookie is deleted, and the user is requested to login again.
Therefore, when using cookies in an application, measures are necessary such as reducing the number of cookies as much as possible or deleting unnecessary cookies.

In addition, the IceWall server is also equipped with a function to retain the IceWall SSO session management cookie. If maintaining the cookie in the application is difficult to implement, use this function to prevent the cookie from being deleted. However, please note that by using this function, the IceWall SSO session management cookie is sent to the browser for each access and the oldest cookie used in the application is deleted.

13.7 Notes on the Set-Cookie header format

Please be aware of the following points regarding the Set-Cookie header format.
- The only character that can be used as the Set-Cookie header separator is semicolon (";").
- Multiple cookies cannot be written in the Set-Cookie header. If you wish to set multiple cookies, write each Set-Cookie header on a separate line.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

Improper description method
Set-Cookie: [Name 1]=[Value 1]; [Name 2]=[Value 2]; expires=[Date and time]; path=[Send path]; domain=[Domain name]
Set-Cookie: [Name 1]=[Value 1]: expires=[Date and time]: path=[Send path]: domain=[Domain name]

Proper description method
Set-Cookie: [Name 1]=[Value 1]; expires=[Date and time]; path=[Send path]; domain=[Domain name]
Set-Cookie: [Name 2]=[Value 2]; expires=[Date and time]; path=[Send path]; domain=[Domain name]

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.

14 Restrictions on Agent Installation

The following are the restrictions for installing an agent on a web server.

14.1 Eliminating Authentication Functions

If basic authentication or other authentication functions are configured on the existing web server, eliminate those authentication functions or disable those functions. Because the agent is directly installed on the existing web server, basic authentication and form authentication cannot be controlled as with a Forwarder.

14.2 URL Conversion and Keyword Conversion

Because the agent is directly installed on the existing web server as “part of the web server,” it is not necessary for the URLs within the content to be converted.
Therefore, a URL conversion function is not provided with this installation.
Similarly, a keyword conversion function is not provided.
If, after installing the agent, changes need to be made to the content, the content must be edited directly.

14.3 Access Logs

The agent does not output access logs. If access logs are required, see the access logs output by the web server.

14.4 Using Client Certificates

Note that, on the Forwarder, the source of the client certificate can be specified as an option; however, on the agent, the certificate is retrieved with the standard Web server name.

(c) Copyright 2001-2010 Hewlett-Packard Development Company, L.P.