

## HanWEB Fanjian Translation Server: Network Connectivity Overview

HanWEB Fanjian Translation Server is a versatile web-based translation server developed by KanHan Technologies. This document gives an overview of its network connectivity.

### TCP/IP Port Requirements

Like a typical web server, HanWEB requires one listening port for accepting incoming HTTP requests. It is usually port 80 but can be set to any other valid ports (0 through 65535) in the configuration file. To serve HTTP requests over SSL (HTTPS), an additional listening port is required, which is usually port 443.

HanWEB must also be able to connect to the HTTP and HTTPS ports of web servers so that it can retrieve web pages from them for translation.

### Secure Socket Layer (SSL) Connections

When an X.509 digital certificate is provided and appropriate settings are present in the configuration file, HanWEB can accept HTTP requests over SSL. To a web browser, HanWEB is just another secure web server. Similarly, HanWEB is just an SSL-enabled client to a web server. The following figure illustrates the relationship among the three parties:

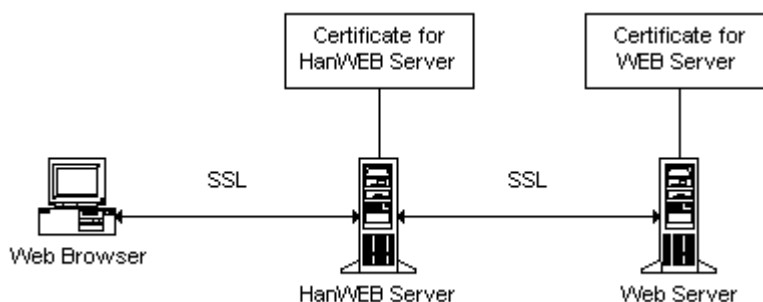


Figure 1. SSL connections

As shown in the figure, both connections are SSL connections. To ensure transparency and security, HanWEB will never make a non-SSL connection to the web server when that from the web browser is an SSL connection.

## Application Layer (HTTP) Connections

As mentioned earlier, HanWEB behaves like a web server to the web browser and behaves like a web browser to the web server. HTTP is the application layer protocol used for communications among the three parties:

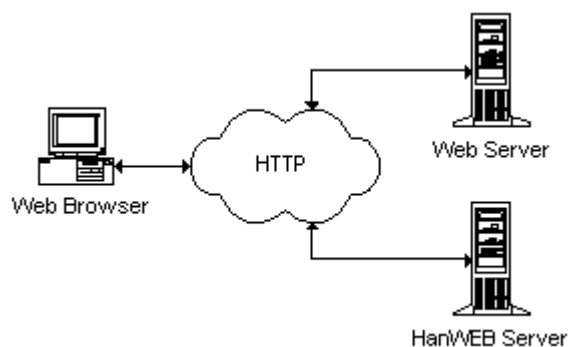


Figure 2. HTTP connections

The following is a typical sequence of events that happen at HanWEB when a HTTP request is received:

1. HTTP request received from the browser:

```
GET /gb/www.kanhan.com/ HTTP/1.1
Accept: */*
User-Agent: Dummy/1.0
Host: hanweb.kanhan.com
Connection: close
```

2. The same HTTP request is modified and sent to the web server:

```
GET / HTTP/1.1
Accept: */*
User-Agent: Dummy/1.0
Host: www.kanhan.com
Connection: close
```

3. The web server responds with the following (the response body is not shown):

```
HTTP/1.1 200 OK
Date: Fri, 22 Aug 2003 10:10:42 GMT
Server: Apache/1.3.28 (Unix)
X-Powered-By: PHP/4.3.2
Connection: close
Content-Type: text/html
```

4. HanWEB sends the modified response back to the browser.

```
HTTP/1.1 200 OK
Date: Fri, 22 Aug 2003 10:10:42 GMT
Server: Apache/1.3.28 (Unix)
X-Powered-By: PHP/4.3.2
Connection: close
Content-Type: text/html
Set-Cookie: kanhanBase=hanweb.kanhan.com/gb/ ,www.kanhan.com; path=/
```

By relaying the requests and responses between the web browser and the web server, HanWEB behaves much like a transparent proxy. During the process, HanWEB tries to keep as much information in the HTTP header and body intact as possible.