Implementing HTTPS Browsing for the Web Interface with Apache 2.4.x

Contents

This article provides a process to configure secure browsing (HTTPS) to the Uptime web interface over SSL. The steps are guaranteed to work with Uptime IM 7.7 and later. If you are looking for a similar solution for an earlier version of Uptime IM, please see Implementing HTTPS Browsing for the Web Interface with Apache 2.2.

Note

Upgrading the Uptime Monitoring Station will overwrite the changes to httpd.conf, so when the upgrade is complete, be sure to update the httpd.conf file again.

Configuring SSL

To configure SSL browsing in the Uptime web interface, you must generate a server certificate, which identifies that server is using SSL for security, and perform some platform-specific configuration. The following steps will cover this process.

Generate or obtain a server certificate

You can purchase a recognized certificate from a vendor such as Verisign or Thawte.

Alternately, you can generate your own non-recognized certificate. A non-recognized certificate is one that does not come from a certificate-issuing authority. To generate a non-recognized certificate, download and install the OpenSSL software. OpenSSL binaries for Windows can be obtained from Shining Light Productions.

Once OpenSSL is installed, enter the following commands (changing <openssl_dir> to the proper path for the OpenSSL installation directory) at the command line to generate the certificate key.

```
cd <openssl_dir>/bin
openssl req -new -x509 -newkey rsa:4096 -nodes -out uptime_ssl_server.crt -keyout uptime_ssl_server.key
```

Working with wildcard certs / pfx certs

You'll need to pull key and crt files from the pfx first. To do this:

Take the file you exported (e.g. certname.pfx) and copy it to your Uptime server, or somewhere you have openSSL installed. You’ll need to supply your password the pfx file was created with in the steps that follow.

1. Run the following command to export the private key:
   openssl pkcs12 -in certname.pfx -nocerts -out uptime_ssl_key.pem –nodes
2. Run the following command to export the certificate:
   openssl pkcs12 -in certname.pfx -nokeys -out uptime_ssl_cert.pem
3. Run the following command to remove the passphrase from the private key:
   openssl rsa -in uptime_ssl_key.pem -out uptime_ssl_server.key
4. Run the following command to produce the cert file
   openssl pkcs12 -in certname.pfx -clcerts -nokeys -out uptime_ssl_server.crt

Move the files to the Uptime Infrastructure Monitor directory

Copy the following files to the <uptime_dir>/apache/conf directory where <uptime_dir> is the installation directory of Uptime (the default installation directory is C:\Program Files\uptime software\uptime on Windows and /usr/local/uptime on Linux).

- uptime_ssl_server.key
- uptime_ssl_server.crt

Update *httpd.conf*

The following changes to the web server configuration file (httpd.conf) will allow it to use SSL.

Open <uptime_dir>/apache/conf/httpd.conf for editing. Where <uptime_dir> appears below, change it to reflect the directory where you have Uptime installed (ex. c:/Program Files/uptime software/uptime). All path slashes in httpd.conf need to be forward slashes (rather than the usual backslash that is used in Windows).

To make browsing to the Uptime UI easy for users, have it listen on the default Uptime UI port, 9999, as well as the typical HTTP and HTTPS ports, 80 and 443.
Above the line "Listen 9999", add the following two lines:

```
Listen 80
Listen 443
```

To handle requests on each of these ports, 80, 443, and 9999, and redirect (actually rewrite) them properly, we will leverage the mod_rewrite.so module, so we need to enable it. In the httpd.conf file, uncomment the following line.

```
LoadModule rewrite_module modules/mod_rewrite.so
```

Finally, the last part is to add entries in httpd.conf that will rewrite the requests as HTTPS. At the bottom of the httpd.conf file, add these lines, changing <uptime_dir> to the directory of your Uptime installation. Please note that the following example uses a specific list of ciphers. You can change the list of ciphers according to your security requirements.

```
SSLProtocol -all +TLSv1 +TLSv1.1 +TLSv1.2
SSLCipherSuite ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES256-SHA256:DHE-RSA-AES256-SHA:!RC4:!LOW:!MD5:!aNULL:!eNULL:!3DES:!EXP:!PSK:!SRP:!DSS
SSLHonorCipherOrder On
SSLSessionCache none

<VirtualHost *:80>
 RewriteEngine on
 RewriteCond %{SERVER_PORT} !^443$
 RewriteRule ^/(.*) https://%{SERVER_NAME}/$1 [NC,R,L]
</VirtualHost>

<VirtualHost *:443>
 SSLEngine on
 DocumentRoot "<uptime_dir>/GUI"
 SSLCertificateFile "<uptime_dir>/apache/conf/uptime_ssl_server.crt"
 SSLCertificateKeyFile "<uptime_dir>/apache/conf/uptime_ssl_server.key"
</VirtualHost>

<VirtualHost *:9999>
 RewriteEngine on
 RewriteCond %{SERVER_PORT} !^443$
 RewriteRule ^/(.*) https://%{SERVER_NAME}/$1 [NC,R,L]
</VirtualHost>
```

Update *uptime.conf*

Open the <uptime_dir>/uptime.conf file for editing and change the httpContext parameter (which begins with "httpContext=http://") to reflect the use of SSL:

```
httpContext=https://<Server_Hostname>:443
```

Restart the Uptime Web Server

For the changes to take effect, restart the Uptime Web Server on Windows or uptime_httpd on Linux.

Starting (or restarting) and Stopping Uptime Infrastructure Monitor