SSL and TLS Support in Uptime IM 7.6

In light of vulnerabilities found in SSL and TLS, many IT departments are abiding by the PCI Security Standard which requires secured communication using TLSv1.1 or TLSv1.2. This knowledge base article identifies the areas where TLSv1.1/v1.2 is already implemented or can be implemented in Uptime IM 7.6.

SSL/TLS Support for Uptime IM Services

Uptime Web Server

The Uptime Web Server (Apache) service is not configured by default to use HTTPS (SSL/TLS). By default the Uptime Web Server listens on port 9999 and uses regular HTTP; however, it is possible to configure the service to use HTTPS. The knowledge base article, Implementing HTTPS Browsing for the Web Interface, provides suggestions on how to accomplish this. The Uptime Web Server service can be configured to use all versions of SSL/TLS (SSLv2, SSLv3, TLSv1, TLSv1.1, and TLSv1.2).

Modifying SSL/TLS Settings

If you have followed the Implementing HTTPS Browsing for the Web Interface article to implement HTTPS browsing in Uptime IM, you will have noticed the following line in the <uptime_root>\apache\conf\httpd.conf file.

```
SSLProtocol ALL -SSLv2 -SSLv3
```

That line indicates that all versions of TLS will be accepted but SSLv2 and SSLv3 will be refused. If you only want to accept TLSv1.2, you can replace that line with this one. After making the change, restart the Uptime Web Server service.

```
SSLProtocol -ALL +TLSv1.2
```

Uptime Controller

The Uptime Controller (Jetty) service listens on port 9997 and uses TLS (TLSv1, TLSv1.1, and TLSv1.2) by default. If desired, it is possible to enable support for SSLv3 and it is also possible to remove support for TLSv1 (see “Enabling/Disabling SSL/TLS support for Uptime Controller” below).

Modifying SSL/TLS Settings

Out of the box, the Uptime Controller does not accept connections using SSLv3. The file where this configuration is maintained is <uptime_root>\controller\etc\jetty-ssl.xml.

If you want restrict SSL/TLS further and only accept TLSv1.2 connections, update the ExcludeProtocols section of jetty-ssl.xml with the entries below. After making the change, restart the Uptime Controller service for the change to take effect.

```
<Set name="ExcludeProtocols">
        <Array type="java.lang.String">
                <Item>SSLv3</Item>
                <Item>SSLv2Hello</Item>
                <Item>TLSv1</Item>
                <Item>TLSv1.1</Item>
        </Array>
</Set>
```

Uptime Data Collector

The Uptime Data Collector service listens on two ports, port 9996 for the Event Listener and port 9995 for the Data Collector Core. These ports are used internally by Uptime IM and are rarely/never called externally. The ports do not support SSL/TLS but a firewall restriction can be implemented to block access to these ports if desired.

Uptime Data Store

The Uptime Data Store (MySQL) service listens on port 3308. It is possible that this service has been disabled in favor of using MS SQL or Oracle instead. While it may be possible to configure any of these three databases to use SSL/TLS communication, the Uptime Data Collector service is not capable of connecting to those databases using SSL/TLS, so SSL/TLS is not supported for the Uptime Data Store at this time. To secure communication with the Uptime Datastore, you can restrict access to the DB via firewall rules.

Monitoring Web Services Using SSL/TLS

The HTTP (Web Services) service monitor is available in Uptime IM out of the box. It is capable of monitoring most webpages, including pages that use SSLv3 and TLSv1. The HTTP (Web Services) monitor does not support TLSv1.1 or TLSv1.2 at this time.

The HTTP / Web Services / SOAP (Advanced) plugin monitor is capable of monitoring all versions of SSL/TLS. Although the HTTP / Web Services / SOAP (Advanced) monitor is not available out of the box, it is free and easy to install using the Extension Manager. Details on installing plugins can be found in the Plugin Monitors portion of the online documentation.

Testing an SSL/TLS Connection

It is possible to test the SSL/TLS connection for a given port/service using the OpenSSL steps listed below. If you are using Windows, Shining Light Productions provides a Windows version of OpenSSL.

1. Open a command prompt.
2. Change to the bin directory of OpenSSL.
3. Run `openssl s_client -connect <hostname>:<port> -<ssl_version>` where <ssl_version> can be ssl2, ssl3, tls1, tls1_1, or tls1_2.

Examples

```
openssl s_client -connect uptime:443 -tls1_2

openssl s_client -connect uptime:9997 -ssl3
```