

# Host PME over HTTP and HTTPS with SSL enabled

PME can be hosted using SSL for all web application. The following sections describe the required configuration changes

## Default PME SSL Certificate

Icon

PME will work over SSL by default without any changes to configuration (<https://SERVERNAME/Web>). It is STRONGLY recommended that a valid SSL Certificate (or at minimum a new Self-Signed Certificate) is added and configured and the described SSL configuration changes below are made. The pre PME 8.1 default certificate (IPKey) is not sufficient to enable HTTPS and disable HTTP (thus making it truly secure).

Internet Explorer Trusted Site

Icon

Over pure SSL, Silverlight applications (Alarms and Tables) may fail to load in Internet Explorer. Clients must add the domain as a Trusted site (Internet Options -> Security -> Sites -> Add)

## IIS Configuration

1. Open IIS
2. Install a valid SSL Certificate.
  - a. This document does not cover the scope of installing a valid SSL certificate!
  - b. There are many resources online, and godaddy has a very succinct overview found here: [Godaddy SSL overview](#).
  - c. If you want to create a self-signed SSL certificate, here are instructions from Microsoft: [Create a Self-Signed Server Certificate in IIS](#).
3. Right-click on the Sites -> **Default Web Site**, and select **Edit Bindings...**
4. Add a new **https** binding for the desired port, or edit the existing default port 443 binding
  - a. Select the desired SSL Certificate (the default IPKey certificate will not function for this purpose!)
  - b. You may add a Host name value to the binding, this value must match the associated certificates "Issued To" property.

## PME Configuration

Application Modules Database update

1. Open SQL Server Management Studio and connect to the PME SQL server.
2. Navigate to Databases -> **ApplicationModules** -> Views
3. Right-click on **Configuration.ConfigurationSettings** and select **Edit Top 200 Rows**
4. Find the entry in the result set with the following values:
  - a. ItemType = *Web Framework*
  - b. Item = *Server*
  - c. Key = *LocalServerAddress*
5. Update the Value column to include your new SSL host (and port if required).
  - a. Example: assuming the https binding certificate is for the host *mysampledomain.com* with the default port then the correct value would be: <https://mysampledomain.com/>
  - b. Example: assuming the https binding certificate is for the host *mysampledomain.com* with the port *567*, then the correct value would be: <https://mysampledomain.com:567/>

## ION\_Network Database update

1. Open SQL Server Management Studio and connect to the PME SQL server.
2. Navigate to Databases -> **ION\_Network** -> Views
3. Right-click on **dbo.vCFG\_ConfigItems** and select **Edit Top 200 Rows**
4. Find the entry in the result set with the following values:
  - a. Module = *Reporting*
  - b. Category = *General Settings*
  - c. Item = *WebServiceUrl*
5. Update the Value column to include your new SSL host (and port if required).
  - a. Example: assuming the https binding certificate is for the host *mysampledomain.com* with the default port then the correct value would be: <https://mysampledomain.com/ionreportdataservice/ReportDataService.asmx>
  - b. Example: assuming the https binding certificate is for the host *mysampledomain.com* with the port *567*, then the correct value would be: <https://mysampledomain.com:567/ionreportdataservice/ReportDataService.asmx>

## Application Framework Web.config update

Silverlight Dependant

Icon

If, in a future version of PME, there are no longer any Silverlight components then this step will not be required.

1. Open the following file in a text editor: [INSTALL FOLDER]\Applications\ApplicationFramework\Web.config
2. Navigate to the following section of the XML file:  
<Configuration> -> <system.serviceModel> -> <services>
3. You must "uncomment" the 2 endpoint https bindings (as we have enabled https bindings in IIS).

Before edits the section will look like this:

```
<services>
  <service name="UI.ApplicationFramework.Web.Services.RealTimeTablesService"
behaviorConfiguration="metadata">
  <!--<endpoint address="" binding="basicHttpBinding" bindingConfiguration="https"
bindingName="RealTimeTablesService"
contract="UI.ApplicationFramework.Web.Services.IRealTimeTablesService" />-->
  <endpoint address="" binding="basicHttpBinding" bindingConfiguration="http"
bindingName="RealTimeTablesService"
contract="UI.ApplicationFramework.Web.Services.IRealTimeTablesService" />
  <endpoint address="mex" binding="mexHttpBinding" contract="IMetadataExchange" />
</service>
  <service name="UI.ApplicationFramework.Web.Services.InternationalizationService"
behaviorConfiguration="metadata">
  <!--<endpoint address="" binding="basicHttpBinding" bindingConfiguration="https"
bindingName="InternationalizationService"
contract="UI.ApplicationFramework.Web.Services.IInternationalizationService" />-->
  <endpoint address="" binding="basicHttpBinding" bindingConfiguration="http"
bindingName="InternationalizationService"
contract="UI.ApplicationFramework.Web.Services.IInternationalizationService" />
  <endpoint address="mex" binding="mexHttpBinding" contract="IMetadataExchange" />
</service>
</services>
```

After edits the section should look like this:

```
<services>
  <service name="UI.ApplicationFramework.Web.Services.RealTimeTablesService"
behaviorConfiguration="metadata">
  <endpoint address="" binding="basicHttpBinding" bindingConfiguration="https"
bindingName="RealTimeTablesService"
contract="UI.ApplicationFramework.Web.Services.IRealTimeTablesService" />
  <endpoint address="" binding="basicHttpBinding" bindingConfiguration="http"
bindingName="RealTimeTablesService"
contract="UI.ApplicationFramework.Web.Services.IRealTimeTablesService" />
  <endpoint address="mex" binding="mexHttpBinding" contract="IMetadataExchange" />
</service>
  <service name="UI.ApplicationFramework.Web.Services.InternationalizationService"
behaviorConfiguration="metadata">
  <endpoint address="" binding="basicHttpBinding" bindingConfiguration="https"
bindingName="InternationalizationService"
contract="UI.ApplicationFramework.Web.Services.IInternationalizationService" />
```

```

    <endpoint address="" binding="basicHttpBinding" bindingConfiguration="http"
bindingName="InternationalizationService"
contract="UI.ApplicationFramework.Web.Services.IInternationalizationService" />
    <endpoint address="mex" binding="mexHttpBinding" contract="IMetadataExchange" />
</service>
</services>

```

## WebServices Web.config update

1. Open the following file in a text editor: [INSTALL FOLDER]\system\WebServices\Web.config
2. Navigate to the following section of the XML file:  
<Configuration> -> <system.serviceModel> -> <services>
3. You must "uncomment" the 2 (or 5) endpoint https bindings (as we have enabled https bindings in IIS).

Before edits the section will look like this (if the 3 Power Quality services are not present, ignore them for update):

```

<services>
  <service name="WebServices.IONWebServices.Alarms">
    <endpoint binding="webHttpBinding" contract="WebServices.IONWebServices.Alarms"
behaviorConfiguration="RESTBehavior" bindingConfiguration="http" />
    <!--<endpoint binding="webHttpBinding" contract="WebServices.IONWebServices.Alarms"
behaviorConfiguration="RESTBehavior" bindingConfiguration="https" />-->
  </service>
  <service name="WebServices.IONWebServices.RealTime.WebRealTime">
    <endpoint binding="basicHttpBinding"
contract="WebServices.IONWebServices.RealTime.IWebRealTime"
behaviorConfiguration="SOAPBehavior" bindingConfiguration="http" />
    <!--<endpoint binding="basicHttpBinding"
contract="WebServices.IONWebServices.RealTime.IWebRealTime"
behaviorConfiguration="SOAPBehavior" bindingConfiguration="https" />-->
  </service>
  <service name="PowerQuality.Analytics">
    <endpoint binding="webHttpBinding" contract="PowerQuality.Analytics"
behaviorConfiguration="RESTBehavior" bindingConfiguration="http" />
    <!--<endpoint binding="webHttpBinding" contract="PowerQuality.Analytics"
behaviorConfiguration="RESTBehavior" bindingConfiguration="https" />-->
  </service>
  <service name="PowerQuality.EventProcessor">
    <endpoint binding="webHttpBinding" contract="PowerQuality.EventProcessor"
behaviorConfiguration="RESTBehavior" bindingConfiguration="http" />
    <!--<endpoint binding="webHttpBinding" contract="PowerQuality.EventProcessor"
behaviorConfiguration="RESTBehavior" bindingConfiguration="https" />-->
  </service>
  <service name="PowerQuality.DowntimeService.Impact">
    <endpoint binding="webHttpBinding" contract="PowerQuality.DowntimeService.Impact"
behaviorConfiguration="RESTBehavior" bindingConfiguration="http" />
    <!--<endpoint binding="webHttpBinding" contract="PowerQuality.DowntimeService.Impact"
behaviorConfiguration="RESTBehavior" bindingConfiguration="https" />-->
  </service>

```

```
</service>
</services>
```

After edits the section should look like this:

```
<services>
  <service name="WebServices.IONWebServices.Alarms">
    <endpoint binding="webHttpBinding" contract="WebServices.IONWebServices.Alarms"
behaviorConfiguration="RESTBehavior" bindingConfiguration="http" />
    <endpoint binding="webHttpBinding" contract="WebServices.IONWebServices.Alarms"
behaviorConfiguration="RESTBehavior" bindingConfiguration="https" />
  </service>
  <service name="WebServices.IONWebServices.RealTime.WebRealTime">
    <endpoint binding="basicHttpBinding"
contract="WebServices.IONWebServices.RealTime.IWebRealTime"
behaviorConfiguration="SOAPBehavior" bindingConfiguration="http" />
    <endpoint binding="basicHttpBinding"
contract="WebServices.IONWebServices.RealTime.IWebRealTime"
behaviorConfiguration="SOAPBehavior" bindingConfiguration="https" />
  </service>
  <service name="PowerQuality.Analytics">
    <endpoint binding="webHttpBinding" contract="PowerQuality.Analytics"
behaviorConfiguration="RESTBehavior" bindingConfiguration="http" />
    <endpoint binding="webHttpBinding" contract="PowerQuality.Analytics"
behaviorConfiguration="RESTBehavior" bindingConfiguration="https" />
  </service>
  <service name="PowerQuality.EventProcessor">
    <endpoint binding="webHttpBinding" contract="PowerQuality.EventProcessor"
behaviorConfiguration="RESTBehavior" bindingConfiguration="http" />
    <endpoint binding="webHttpBinding" contract="PowerQuality.EventProcessor"
behaviorConfiguration="RESTBehavior" bindingConfiguration="https" />
  </service>
  <service name="PowerQuality.DowntimeService.Impact">
    <endpoint binding="webHttpBinding" contract="PowerQuality.DowntimeService.Impact"
behaviorConfiguration="RESTBehavior" bindingConfiguration="http" />
    <endpoint binding="webHttpBinding" contract="PowerQuality.DowntimeService.Impact"
behaviorConfiguration="RESTBehavior" bindingConfiguration="https" />
  </service>
</services>
```

4. Navigate to the following section of the XML file: <Configuration> -> <system.webServer> -> <security>

5. You must add a new rule to allow connections to this server via the SSL certificate (as this service is only accessible locally)

Before edits the section will look like this:

```
<security>
  <ipSecurity allowUnlisted="false">
    <add ipAddress="127.0.0.1" allowed="true" />
  </ipSecurity>
</security>
```

After edits the section should look like this (assuming the https binding certificate is for the host [mysampledomain.com](http://mysampledomain.com)):

```
<security>
  <ipSecurity allowUnlisted="false">
    <add ipAddress="127.0.0.1" allowed="true" />
    <add domainName="mysampledomain.com" allowed="true"/>
  </ipSecurity>
</security>
```

See "Validating the configuration" Icon

There is a potential that this file may require additional configuration steps. Please review the [Validating the Configuration](#) section below once you have applied all settings and are in the process of validating the configuration is working as intended. The Web applications may "seem" to be working, but there are some special edge cases to look out for below.

### Hosts file update

It is very important the the server can identify itself via the domain name registered in the SSL certificate.

1. Navigate to: C:\Windows\System32\drivers\etc
2. Open the file *hosts* in a Notepad
3. Add an entry for your domain mapped to 127.0.0.1
  - a. Example: assuming the https binding certificate is for the host [mysampledomain.com](https://mysampledomain.com) then the correct value would be: 127.0.0.1 [mysampledomain.com](https://mysampledomain.com)
4. If this entry was not present, you must flush the local DNS cache. This may be done in 2 ways:
  - a. In a command window, run this command: **ipconfig /flushdns**
  - b. Alternately, you may reboot the server

### Default Web Application link update

1. Open a Windows explorer window and navigate to: Desktop -> **StruxureWare Power Monitoring Expert**
2. Right-click on **Web Applications** and select **Properties**
3. Update the **URL** value with the updated url
  - a. Example: assuming the https binding certificate is for the host *mysampledomain.com* with the default port then the correct value would be: <https://mysampledomain.com/Web>
  - b. Example: assuming the https binding certificate is for the host *mysampledomain.com* with the port 567, then the correct value would be: <https://mysampledomain.com:567/Web>

## Final Steps

1. Open a windows Command window (cmd.exe) and run **iisreset**
2. Open the windows Services console, and restart the following service: **ApplicationModules**  
**CoreServicesHost**
  - a. Click Yes to restart ApplicationModules services
3. See the section below "Validating the Configuration" to ensure all features are working as intended.