**APPLIES TO:**

Oracle HTTP Server - Version 12.1.2.0.0 to 12.1.3.0.0 [Release 12c]  
Information in this document applies to any platform.  
This is for OHS 12c release 1 (12.1.2/12.1.3). The same concepts apply for Release 2 (12.2.x) with some minor changes, upgraded SSL features, and different options. It is encouraged to use the new Oracle Documentation to get started with the configuration of SSL with Oracle HTTP Server 12c Release 2 (12.2.x). Obtain the specific version from the doc library at [https://docs.oracle.com/en/middleware/middleware.htm](https://docs.oracle.com/en/middleware/middleware.html) looking for FMW and OHS Administrator's Guides.

**PURPOSE**

**How to Configure Oracle HTTP Server to use SSL in Fusion Middleware 12c (12.1.X)**

To list the steps needed to configure Oracle HTTP Server (OHS) to use the Secure Sockets Layer (SSL) when installed with Oracle Fusion Middleware 12c.

Steps provided show how to configure new server certificates using FMW Control, orapki and WLST tools.

**Note:**  
  
This is part of a number of articles written for SSL Configuration in FMW 12c. Please read [Note:1628909.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&sourceId=1662675.1&id=1628909.1) Primary Note for SSL Configuration in Fusion Middleware

This article is to be used after reviewing the following Oracle Documentation:  
[Administering Oracle Fusion Middleware 12c - 6.4 Configuring SSL for the Web Tier](http://docs.oracle.com/middleware/1212/core/ASADM/sslconfig.htm#CBDEEDCF)

**SCOPE**

**DETAILS**

There are two major steps needed to configure SSL in Fusion Middleware 12c (12.1.X)

I. Create an Oracle Wallet which contains an SSL Certificate.  
II. Configure ssl.conf directives to enable SSL with OHS.

**Step I: Creating an Oracle Wallet**

As outlined in some of the referenced notes, there are several ways to create an Oracle Wallet in Fusion Middleware 12c. To summarize the methods are as follows:

***Fusion Middleware Control  
ORAPKI  
WLST***

* The one you choose to use depends on your circumstances. Please read [Note 1218603.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&sourceId=1662675.1&id=1218603.1) Understanding Wallets and Keystores in Fusion Middleware 11g/12c
* Generally speaking the recommendation is to use FMW Control, however remember that FMW Control can **\*only\*** be used if using a **Collocated OHS Domain**. For an**OHS Standalone Domain**, use ORAPKI.
* Choose whichever method suits and follow the relevant "*How to Create a Wallet Via* ...." Note in [Note 1628909.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&sourceId=1662675.1&id=1628909.1) Primary Note for SSL Configuration in Fusion Middleware 12c Section II: Wallets and Keystores in FMW 12c.

When the Wallet is complete and contains a valid certificate move to Step II.

**Step II: Configuring HTTP Server for SSL**

In a **Collocated OHS Domain**, OHS can be configured using either **FMW Control** or by **editing the configuration manually**.

In a **Standalone OHS Domain**, OHS can only be configured by **editing the configuration manually**.

**Configuring SSL in a Collocated OHS Domain**

As already referenced, in a Collocated OHS Domain, OHS can be configured using either FMW Control or by editing the configuration manually.  
This section will concentrate on configuring via **Fusion Middleware Control.**  
  
If you want to edit the files manually then follow the **"Configuring SSL in a Standalone OHS Domain**", however when configuring, the AdminServer and NodeManager should be **\*SHUTDOWN\*** and only brought back up once the configuration changes are fully made!  
  
The OHS out of the box install creates an SSL Virtual Host for you, which is configured to use a dummy certificate. The steps below will show how to edit the existing SSL Virtual Host. If you choose to create a new Virtual Host see the later section.

***Editing an Existing SSL Virtual Host***

1. Follow Steps 1-8 in [Section 6.4.2.1 Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using Fusion Middleware Control](http://docs.oracle.com/middleware/1212/core/ASADM/sslconfig.htm#CBDJJDCI) with the following caveats:

* Ignore Step 2
* At Step 5  for "*Server Wallet Name*" select the Wallet you created in Step I.  
  Note if you created your Wallet ORAPKI, then you need to import it into FMW Control by following [Section 8.4.4.9. Importing a Wallet Using Fusion Middleware Control](http://docs.oracle.com/middleware/1212/core/ASADM/wallets.htm#CIHBBFFF)
* At Step 5  for "*SSL Authentication*" select "*Server Authentication*". It is not recommended to set "No Authentication" for security reasons, and "Mutual" and "Optional Client Authentication" are used  if client certificates are required.    
  For client authentication see [Note 1665237.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&sourceId=1662675.1&id=1665237.1) Configuring SSL Client Authentication with Oracle HTTP Server in Fusion Middleware 12c (12.1.X)
* At Step 5 select the "CipherSuites" you wish to use, or leave All as the default.
* At Step 5 select the "SSL Protocol" version you wish to use. In FMW 12c, "v1" refers to SSL V3.0, v3\_v2Hello refers to SSLV3 with SSLv2 Client Hello, and "v3" refers to to TLS 1.0  - Be aware that there are a couple of bugs here in that in  it allows you to set unsupported ciphersuites/protocols (12.1.2), and FMW Control does not allow you to set TLS 1.1 or 1.2 (12.1.2 and 12.1.3). If you need to set specific ciphersuites and protocols do this manually by following the "Configuring SSL Via Manual Configuration". See the SSLProtocol and SSLCipherSuite sections in [G.2 mod\_ossl](http://docs.oracle.com/middleware/1212/webtier/HSADM/directives.htm#CIHJAHID)
* At Step 6 check the staging directory  *$ORACLE\_HOME/user\_projects/domains/<domain\_name>/config/fmwconfig/components/OHS/ohs1/ssl.conf*, and make sure the changes to the SSLWallet directive etc.. were made. If the changes were not made  then you are hitting [Bug 18288409](https://support.oracle.com/epmos/faces/BugDisplay?parent=DOCUMENT&sourceId=1662675.1&id=18288409) EM AUTO COMMIT MODE IS NOT BEING HONORED. If hitting this bug then follow the Solution in [Note 1632717.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&sourceId=1662675.1&id=1632717.1) Fusion Middleware Control Operations Not Committing OHS Changes To File System in FMW 12.1.2. Once the changes are made to the file system, also be aware of the fact, that at this point you may see an incorrect entry as per the following section in the Release Notes:[1.25 SSLWallet Directive in ssl.conf Might be Saved Incorrectly.](http://docs.oracle.com/middleware/1212/webtier/OHSRN/known_issues.htm#CHDJABBF)

e.g:

SSLWallet  "${ORACLE\_INSTANCE}/config/fmwconfig/components/${COMPONENT\_TYPE}/instances/${COMPONENT\_NAME}/keystores//OHS/user\_projects/domains/base\_domain/config/fmwconfig/components/OHS/ohs1/keystores/<wallet>"

 If this happens then follow the advice in the release notes and go to '*Webtier*' -> *'ohs1'* -> '*Administration'* -> *'Advanced Server Configuiration'* -> '*ssl.conf*' -> *'Go'*  
 and change the directive to:

SSLWallet "${ORACLE\_INSTANCE}/config/fmwconfig/components/${COMPONENT\_TYPE}/instances/${COMPONENT\_NAME}/keystore/<wallet>"

 Again make sure the file has been updated properly following [Note 1632717.1](https://support.oracle.com/epmos/faces/DocumentDisplay?parent=DOCUMENT&sourceId=1662675.1&id=1632717.1) Fusion Middleware Control Operations Not Committing OHS Changes To File System in FMW 12.1.2 if it hasn't.  
   
 Once the file is updated properly proceed at Step 8 and restart OHS    
   
  
2. Test you can connect from a browser to your Virtual Host: via *https://<HOSTNAME>:<PORT>* e.g *https://host.domain:4443*

***To Create A New Virtual Host***

1. Select "*Web Tier*" -> "*ohs1*" -> Right click "*Administration*" -> "*Virtual Hosts*"  
2. Click the "*Create*" button  
3. In the Virtual Host Name add a new listen address, for example: \*:4449

If you are creating an SSL port on any port <1024, please see [Enabling Oracle HTTP Server to Run as Root for Ports Set to Less Than 1024 (UNIX Only)](http://docs.oracle.com/middleware/1212/core/ASADM/ports.htm#BABFHACI)

4. Set "*Type*" as "*IP-based*". SSL does not support Name Based Virtual hosts hence why IP-based is required.  
5. Fill in any of the other parameters as required and hit OK  
6. Highlight the new Virtual Host and select "*Configure*" -> "*SSL Configuration*"  
7. Follow the Steps 4-8 in [Section 6.4.2.1 Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using Fusion Middleware Control](http://docs.oracle.com/middleware/1212/core/ASADM/sslconfig.htm#CBDJJDCI) with the same caveats outlined in the "**Editing an existing SSL Virtual Hos**t" section  
8. Test you can connect from a browser to your Virtual Host: via https://<HOSTNAME>:<PORT> e.g https://host.domain:4449  
  
Note: This will add the new virtualHost to the httpd.conf instead of the ssl.conf

**Configuring SSL in a Collocated or Standalone OHS Domain**

***Using Manual Configuration***

If using a**Standalone OHS Domain** it is necessary to configure SSL manually.  
You can also do it this way in a **Collocated OHS Domain** if you prefer not to use FMW Control, with the caveat that the AdminServer and NodeManager should be \***SHUTDOWN**\* and only brought back up once the configuration changes are fully made. Failure to do this can cause consequences further down the line!  
  
  
As mentioned above, out of the box OHS ships with a default SSL VirtualHost. The easiest way to get SSL up and running quickly is to modify this VirtualHost.  
  
1. Backup the \*staging\* *$ORACLE\_HOME/user\_projects/domains/<domain\_name>/config/fmwconfig/components/OHS/ohs1/ssl.conf*  
2. Edit the ssl.conf and locate the VirtualHost section:

<VirtualHost \*:4448> #OHS\_SSL\_VH  
  <IfModule ossl\_module>  
   #  SSL Engine Switch:  
   #  Enable/Disable SSL for this virtual host.  
   SSLEngine on  
  
   #  Client Authentication (Type):  
   #  Client certificate verification type and depth.  Types are  
   #  none, optional and require.  
   SSLVerifyClient none  
  
   #  SSL Protocol Support:  
   #  List the supported protocols.  
   SSLProtocol nzos\_Version\_1\_0 nzos\_Version\_3\_0  
  
  
    # SSL Cipher Suite:  
    # List the ciphers that the client is permitted to negotiate.  
       SSLCipherSuite SSL\_RSA\_WITH\_RC4\_128\_MD5,SSL\_RSA\_WITH\_RC4\_128\_SHA,SSL\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA,SSL\_RSA\_WITH\_DES\_CBC\_SHA,TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA,TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA  
  
   # SSL Certificate Revocation List Check  
   # Valid values are On and Off  
   SSLCRLCheck Off  
  
   #Path to the wallet  
   SSLWallet "${ORACLE\_INSTANCE}/config/fmwconfig/components/${COMPONENT\_TYPE}/instances/${COMPONENT\_NAME}/keystores/<wallet>"  
  
  
    <FilesMatch "\.(cgi|shtml|phtml|php)$">  
    SSLOptions +StdEnvVars  
   </FilesMatch>  
  
   <Directory "${ORACLE\_INSTANCE}/config/fmwconfig/components/${COMPONENT\_TYPE}/instances/${COMPONENT\_NAME}/cgi-bin">  
    SSLOptions +StdEnvVars  
   </Directory>  
  
   BrowserMatch ".\*MSIE.\*" \  
   nokeepalive ssl-unclean-shutdown \  
   downgrade-1.0 force-response-1.0  
  
  </IfModule>  
 </VirtualHost>

3. Change the SSLWallet directive to point at the Wallet created in Step I e.g:

SSLWallet "${ORACLE\_INSTANCE}/config/fmwconfig/components/${COMPONENT\_TYPE}/instances/${COMPONENT\_NAME}/keystore/<wallet>"

4. Save the file  
5. Start or restart HTTP Server  
  
a) Make sure NodeManager is running

$ORACLE\_HOME/user\_projects/domains/<DOMAIN\_NAME>/bin/startNodeManager.sh

b) Restart OHS :

$ORACLE\_HOME/user\_projects/domains/<DOMAIN\_NAME>/bin/stopComponent.sh ohs1  
$ORACLE\_HOME/user\_projects/domains/<DOMAIN\_NAME>/bin/startComponent.sh ohs1

6. Make sure the SSLWallet changes have been synchronised to the \*runtime\* directory *$ORACLE\_HOME/user\_projects/domains/<DOMAIN\_NAME>/config/fmwconfig/components/OHS/instances/ohs1/ssl.conf*  
  
7. Test you can connect from a browser to your Virtual Host:  
  
https://<HOSTNAME>:<PORT> e.g https://host.domain:4448