

Apache Server Configuration for FLEXCUBE
Oracle FLEXCUBE Universal Banking
Release 14.6.2.0.0
Part No. F72916-01
[November] [2022]



Table of Contents

- 1. **INSTALLATION AND CONFIGURATION OF APACHE 2.2.25** 1-1
 - 1.1 **PURPOSE**..... 1-1
 - 1.2 **INTRODUCTION**..... 1-1
- 2. **INSTALLATION OF APACHE** 2-1
- 3. **CONFIGURE APACHE SERVER INFRONT OF WEBLOGIC SERVER**..... 3-1
- 4. **CONFIGURING SSL FOR APACHE SERVER**..... 4-1
 - 4.1 **SSL CONFIGURATION FOR INBOUND REQUEST TO APACHE** 4-1
 - 4.2 **CONFIGURING SSL BETWEEN APACHE AND ORACLE WEBLOGIC SERVER**..... 4-3
- 5. **STARTING, STOPPING, AND RESTARTING APACHE**..... 5-1
- 6. **DEBUGGING** 6-1

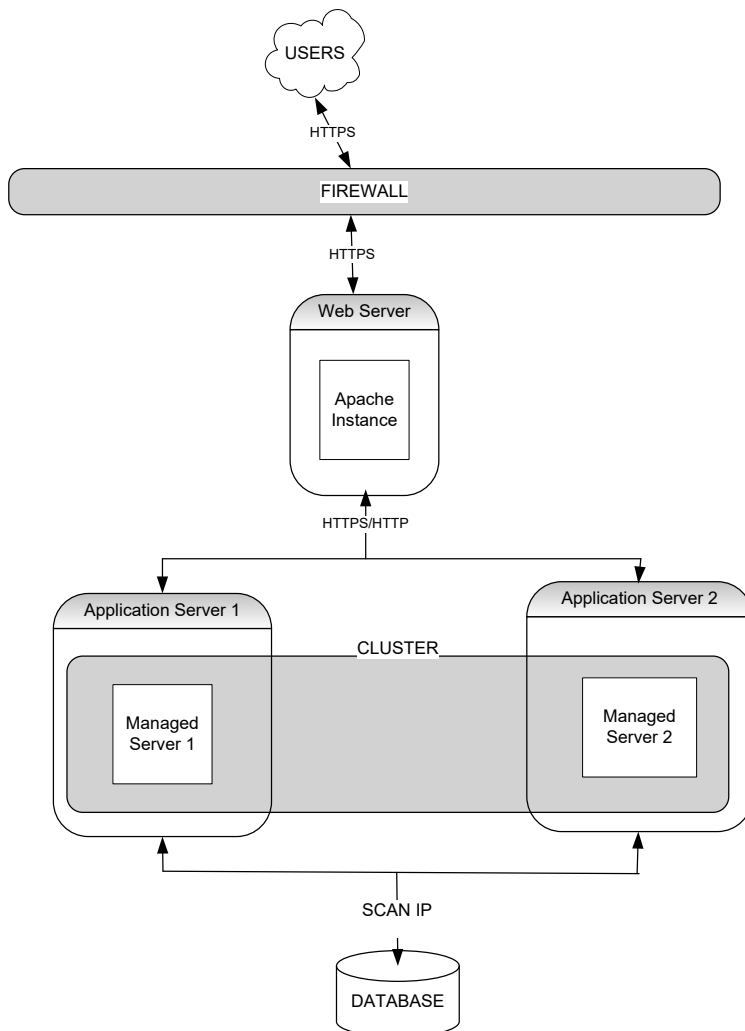
1. Installation and Configuration of Apache 2.2.25

1.1 Purpose

The objective of this document is to explain the installation and configuration of Apache 2.2.25. This includes setting up of server details and enabling SSL.

1.2 Introduction

Below is the typical deployment diagram and this document covers the setup for Apache Webserver Instance.



2. Installation of Apache

Unzip the Apache software, in this example the unzipped location is `/scratch/oracle/software/httpd-2.2.25`. Below steps should be executed from inside this directory location.

Software Directory: `/scratch/oracle/software/httpd-2.2.25`

1. `$./configure --prefix=/scratch/oracle/apache --enable-ssl --with-included-apr`

Where `/scratch/oracle/apache` is the apache home directory

2. `$ make`
3. `$ make install`

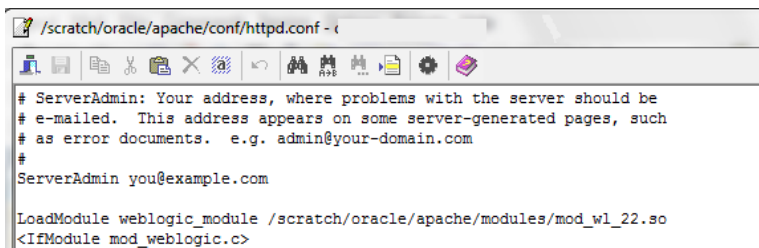
3. Configure Apache Server in front of Weblogic Server

1. Copy mod_wl_22.so from any weblogic Server (\$WLS_HOME/server/plugin/linux/x_86_64) to \$PREFIX/modules/ location in Apache Server.

NOTE: In this example the OS is linux, copy the file from appropriate folder according to the weblogic installed directory.

2. Edit http.conf located under folder \$PREFIX/conf to include mod_wl_22.conf file.

```
LoadModule weblogic_module /scratch/oracle/apache/modules/mod_wl_22.so
```

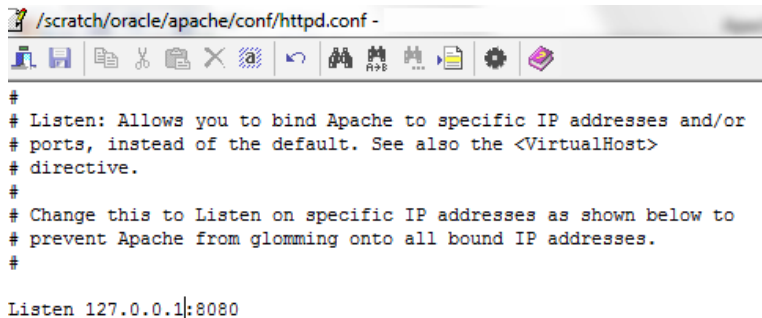


```
/scratch/oracle/apache/conf/httpd.conf - t
# ServerAdmin: Your address, where problems with the server should be
# e-mailed. This address appears on some server-generated pages, such
# as error documents. e.g. admin@your-domain.com
#
ServerAdmin you@example.com

LoadModule weblogic_module /scratch/oracle/apache/modules/mod_wl_22.so
<IfModule mod_weblogic.c>
```

3. Modify http.conf file for the required listen port

```
Listen <HOST NAME>:<PORT>
```



```
/scratch/oracle/apache/conf/httpd.conf -
#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
Listen 127.0.0.1:8080
```

NOTE: This port is for http protocol and not for https

4. Modify the http.conf file to include the weblogic server details

- a. For WebLogic in single instance

```
<Location /<<context/url>> >
```

```
SetHandler weblogic-handler
```

WebLogicHost <<server name>>

WeblogicPort <<port>>

</Location>

Example:

<Location /FCJNeoWeb>

SetHandler weblogic-handler

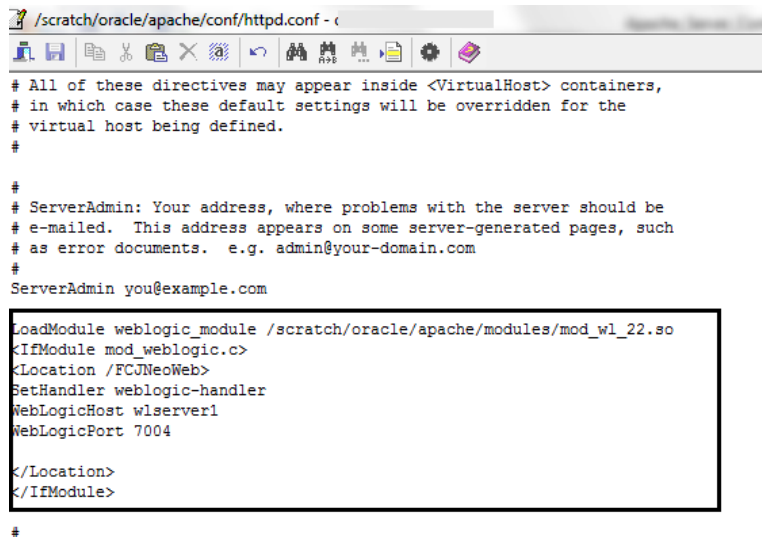
WebLogicHost wlsserver1

WeblogicPort 7004

</Location>

This will forward /FCJNeoWeb from Apache server to /FCJNeoWeb on WebLogic Server wlsserver1: 7004

NOTE: If you want to allow more than one context root, then either add different Location entries for each context root eg: <Location /FCJNeoWeb>, <Location /FCJWebServices>, etc or Add <Location /> which will all the context roots.



```
/scratch/oracle/apache/conf/httpd.conf - c
# All of these directives may appear inside <VirtualHost> containers,
# in which case these default settings will be overridden for the
# virtual host being defined.
#
#
# ServerAdmin: Your address, where problems with the server should be
# e-mailed. This address appears on some server-generated pages, such
# as error documents. e.g. admin@your-domain.com
#
ServerAdmin you@example.com

LoadModule weblogic_module /scratch/oracle/apache/modules/mod_wl_22.so
<IfModule mod_weblogic.c>
<Location /FCJNeoWeb>
SetHandler weblogic-handler
WebLogicHost wlsserver1
WebLogicPort 7004
</Location>
</IfModule>
#
```

b. For Weblogic instances in cluster

<Location /<<context/url>> >

```
SetHandler weblogic-handler
```

```
WebLogicCluster <server1>:<port1>,<server2>:<port2>
```

```
</Location>
```

Example

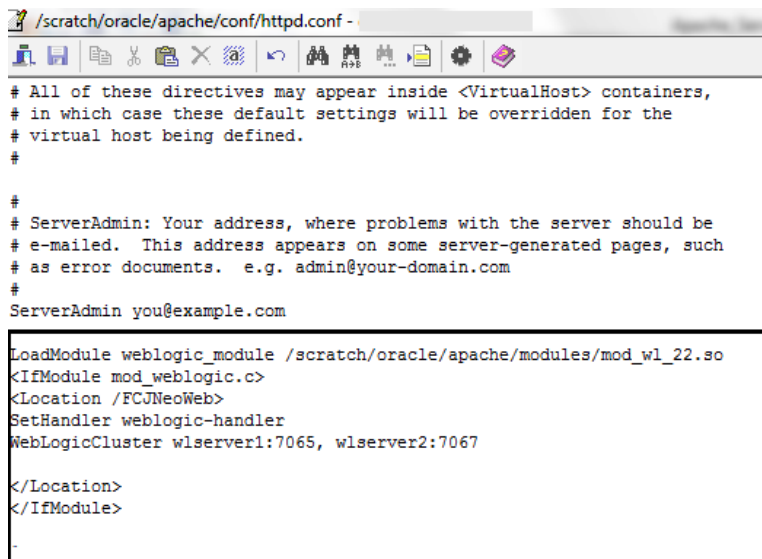
```
<Location / FCJNeoWeb >
```

```
SetHandler weblogic-handler
```

```
WebLogicCluster wls1:7065, wls2:7067
```

```
</Location>
```

This will forward /FCJNeoWeb from Apache server to /FCJNeoWeb on WebLogic Cluster wls1:7065 and wls2:7067



```
/scratch/oracle/apache/conf/httpd.conf -  
# All of these directives may appear inside <VirtualHost> containers,  
# in which case these default settings will be overridden for the  
# virtual host being defined.  
#  
#  
# ServerAdmin: Your address, where problems with the server should be  
# e-mailed. This address appears on some server-generated pages, such  
# as error documents. e.g. admin@your-domain.com  
#  
ServerAdmin you@example.com  
  
LoadModule weblogic_module /scratch/oracle/apache/modules/mod_wl_22.so  
<IfModule mod_weblogic.c>  
<Location /FCJNeoWeb>  
SetHandler weblogic-handler  
WebLogicCluster wls1:7065, wls2:7067  
  
</Location>  
</IfModule>  
-
```

5. Enable “WebLogic Plug-In Enabled” flag in weblogic

This flag needs to be enabled in weblogic if it is accessed through proxy plugins. When the WebLogic plugin is enabled, a call to getRemoteAddr will return the address of the browser client from the proprietary WL-Proxy-Client-IP header instead of the web server.

a. Plugin flag at managed server level

- i. Click on ‘Environment’ -> ‘Servers’ -> ‘<ManagedServer>’ -> ‘General’ -> ‘Advanced’
- ii. Check the ‘WebLogic Plug-In Enabled’ box.

- iii. Click 'Save'
 - iv. Restart the Server.
- b. Plugin flag at domain level
- i. Click on <Domain> -> 'Web Applications'
 - ii. Check the 'WebLogic Plug-In Enabled' box.
 - iii. Click 'Save'
 - iv. Restart the server.
6. Restart the apache server and application can be accessed using link

<http://<hostname>:<port>/FCJNeoWeb/>

4. Configuring SSL for Apache Server

Secure Sockets Layer (SSL) is required to run any Web site securely. Secure Sockets Layer (SSL) is an encrypted communication protocol that is designed to securely send messages across the Internet.

Reading of “**SSL_Configuration on Weblogic**” document provided as part of FCUBS installation is recommended before proceeding with further setup.

SSL configuration can be done between

1. Browser to Apache Server(Mandatory)
2. Apache to Oracle Weblogic Server(If required)

4.1 SSL configuration for Inbound Request to Apache

Perform these tasks to enable and configure SSL between browser and Apache Server.

1. In httpd.conf, uncomment the below line

Include conf/extra/httpd-ssl.conf

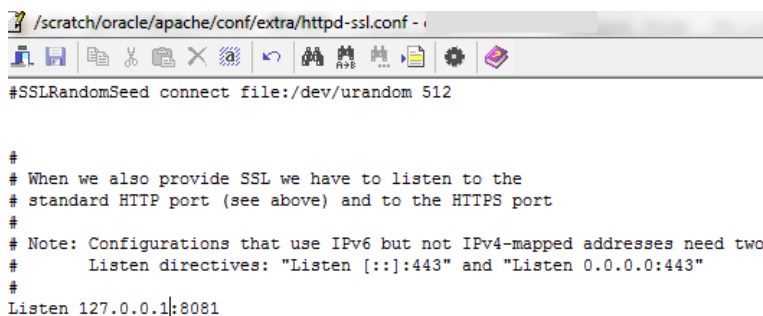


```
/scratch/oracle/apache/conf/httpd.conf - 1
# Various default settings
#Include conf/extra/httpd-default.conf

# Secure (SSL/TLS) connections
Include conf/extra/httpd-ssl.conf
#
```

2. Edit httpd-ssl.conf file located in \$PREFIX/conf/extra/ to give the server address and port as below

Listen <hostname>:<port>



```
/scratch/oracle/apache/conf/extra/httpd-ssl.conf - 1
#SSLRandomSeed connect file:/dev/urandom 512

#
# When we also provide SSL we have to listen to the
# standard HTTP port (see above) and to the HTTPS port
#
# Note: Configurations that use IPv6 but not IPv4-mapped addresses need two
#       Listen directives: "Listen [::]:443" and "Listen 0.0.0.0:443"
#
Listen 127.0.0.1:8081
```

NOTE: This is the https port and not for http

3. Obtain a certificate from CA or create a self signed certificate using openssl

a. Steps to create certificate using openssl

```
openssl genrsa -des3 -out test.key 1024
```

```
openssl req -new -key test.key -out test.csr
```

```
openssl x509 -req -days 365 -in test.csr -signkey test.key -out test.crt
```

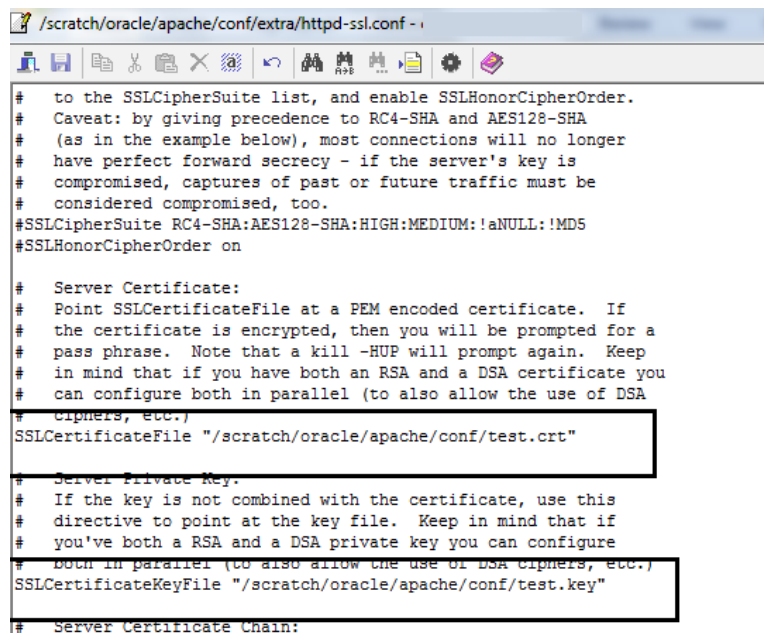
This will generate 3 files named test.key, test.csr, test.crt

4. Copy the files with extension.crt and .key to \$PREFIX/conf/ folder.

5. Edit httpd-ssl.conf to give the location and name for .crt and .key as below

```
SSLCertificateFile "/scratch/oracle/apache/conf/test.crt"
```

```
SSLCertificateKeyFile "/scratch/oracle/apache/conf/test.key"
```



6. Restart apache and application can be accessed using URL

<https://<hostname>:<port>/FCJNeoWeb/>

NOTE: When apache is started with SSL enabled it will ask for pass phrase: here enter the pass phrase used during creation of certificate

4.2 Configuring SSL between Apache and Oracle Weblogic Server

1. Obtain a certificate from CA or create a self signed certificate using Keytool

```
keytool -genkeypair -alias testselfcert -keyalg RSA -keypass admin123 -validity 365 -keystore testidentity.jks
```

```
keytool -export -alias testselfcert -file test.cer -keystore testidentity.jks
```

2. Configure in weblogic to use the generated keyStore
 - a. Enable the enable SSL port for the managed server

Home > Summary of Environment > Summary of Servers > ManagedServer_1

Settings for ManagedServer_1

Configuration Protocols Logging Debug Monitoring Control Deployments Services Security Note

General Cluster Services Keystores SSL Federation Services Deployment Migration Tuning Over

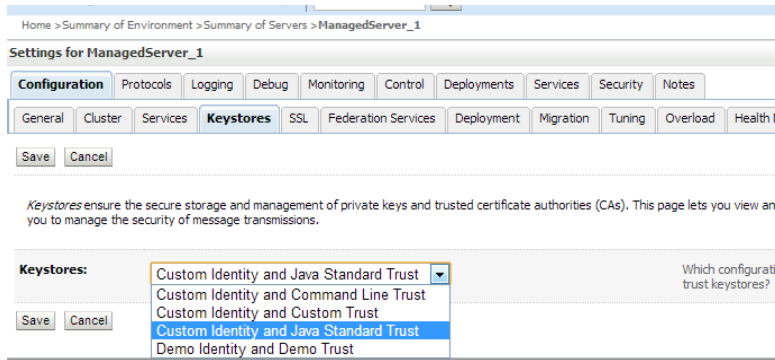
Save

Use this page to configure general features of this server such as default network communications.

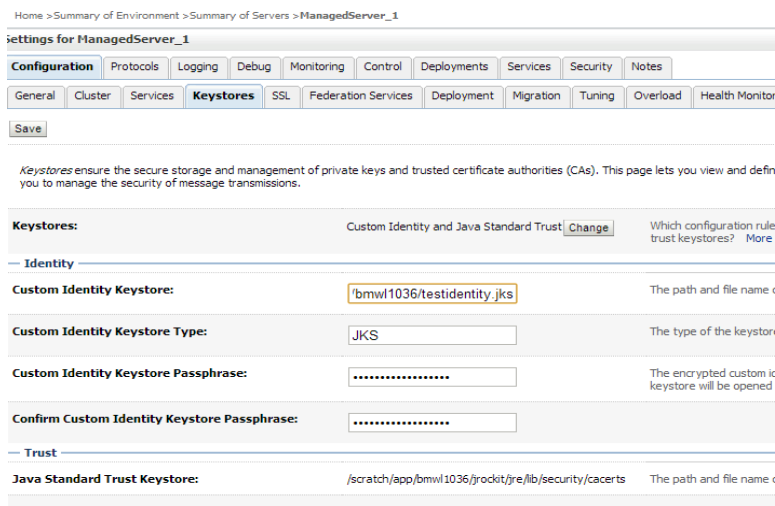
[View JNDI Tree](#)

Name:	ManagedServer_1	An
Machine:	Machine_1	The to r
Cluster:	BCLUSTER	The bel
Listen Address:	<input type="text"/>	The cor
<input checked="" type="checkbox"/> Listen Port Enabled		Spe (no
Listen Port:	<input type="text" value="7078"/>	The incr
<input checked="" type="checkbox"/> SSL Listen Port Enabled		Ind por
SSL Listen Port:	<input type="text" value="7002"/>	The req

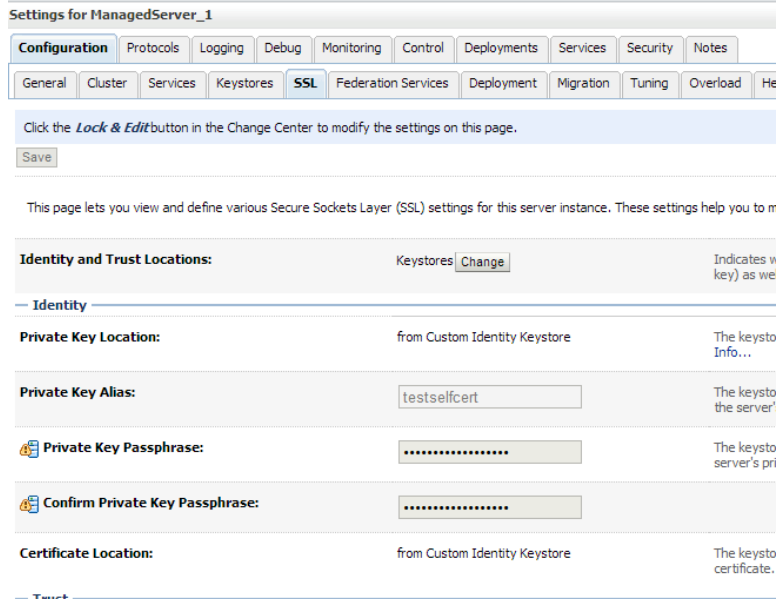
- b. In Keystores tab select Custom Identity and Java Standard Trust



c. Enter the Path where KeyStore generated is stored and then Enter the passphrase



d. Under SSL tab Enter the Alias and the key Passphrase



Certificate displayed by Weblogic needs to be copied to apache, below steps needs to be followed

- e. Execute the openssl command, `openssl s_client -connect <WL host: Port>`

Eg: `openssl s_client -connect wlserver1:7004.`

It will give output as below

```
-bash-4.1$ openssl s_client -connect wlserver1:7004
CONNECTED(00000003)
depth=0 C = IN, ST = IN, L = IN, O = IN, OU = IN, CN = IN
verify error:num=18:self signed certificate
verify return:1
depth=0 C = IN, ST = IN, L = IN, O = IN, OU = IN, CN = IN
verify return:1
---
Certificate chain
 0 s:/C=IN/ST=IN/L=IN/O=IN/OU=IN/CN=IN
  i:/C=IN/ST=IN/L=IN/O=IN/OU=IN/CN=IN
---
Server certificate
-----BEGIN CERTIFICATE-----
MIICEzCCAXygAwIBAgIEUigrQTANBgkqhkiG9w0BAQUFADBOMQswCQYDVQQGEwJJ
TjELMAkGA1UECBMCSU4xCzAJBgNVBACsTAKIOMQswCQYDVQQKEwJJTjELMAkGA1UE
CxMCSU4xCzAJBgNVBAMTAkiOMB4XDTEzMDkwNTA2NTcwNVVoXDTE0MDkwNTA2NTcw
NVowTjELMAkGA1UEBhMCSU4xCzAJBgNVBAGsTAKIOMQswCQYDVQQHEwJJTjELMAkG
A1UEChMCSU4xCzAJBgNVBAsTAKIOMQswCQYDVQQDEwJJTjCBnzANBgkqhkiG9w0B
AQEFAAOBjQAwgYkCgYEAita2L8q7cA+aMbnCIWljXO+2lf+1/Oz3tiLJiC2ulhD/
kd7Q+TxjPS5qsGDhK4jgLOTk4wwNzjawvQVUBY+s5XGwSQatujkgk9kmu+d1zP29
5Hwer0wvA/mRH3k4tj9Os1ueJaHgldl1eS16bhadMV1C7Z9Tr1M+2fdjzGWSI0UC
AwEAATANBgkqhkiG9w0BAQUFAAOBgQA62+BFGN5CMQJkX3YUf1I0KhLJveWQwwGI
OHnW2lchW3YK4YyKsl5b4HdNuBeGjINn47wnujhqPjh6BI8pqbHOYIPjKpNwjTVn
5qkxKZhC5WCMdA3lyyGSQrmUlxJBatw2fVhGaMxdQRy7WujPL5Vf5N+TpedTwXWY
ampTtc+4cw==
```

-----END CERTIFICATE-----

subject=/C=IN/ST=IN/L=IN/O=IN/OU=IN/CN=IN
issuer=/C=IN/ST=IN/L=IN/O=IN/OU=IN/CN=IN

No client certificate CA names sent

SSL handshake has read 665 bytes and written 295 bytes

New, TLSv1/SSLv3, Cipher is RC4-MD5

Server public key is 1024 bit

Secure Renegotiation IS NOT supported

Compression: NONE

Expansion: NONE

SSL-Session:

Protocol : TLSv1

Cipher : RC4-MD5

Session-ID: EF4D92AFB99069ADF35DB9B16E8B36C9

Session-ID-ctx:

Master-Key:

3B9AD7E1F72C0A50FF4C10458F392C763331CDD5313832A8BF28EDEBFFCD8E6D928944D
4698FC302F3A490116DC6E320

Key-Arg : None

Krb5 Principal: None

PSK identity: None

PSK identity hint: None

Start Time: 1378442902

Timeout : 300 (sec)

Verify return code: 18 (self signed certificate)

- f. The section highlighted in red above is the certificate presented by weblogic. This needs to be copied to new file and name this file as server.pem under location \$PREFIX/conf/

3. Apache to be configured to use the SSL

In httpd.conf add the below directive

```
<Location /FCJNeoWeb>
```

```
SetHandler weblogic-handler
```

```
SecureProxy ON
```

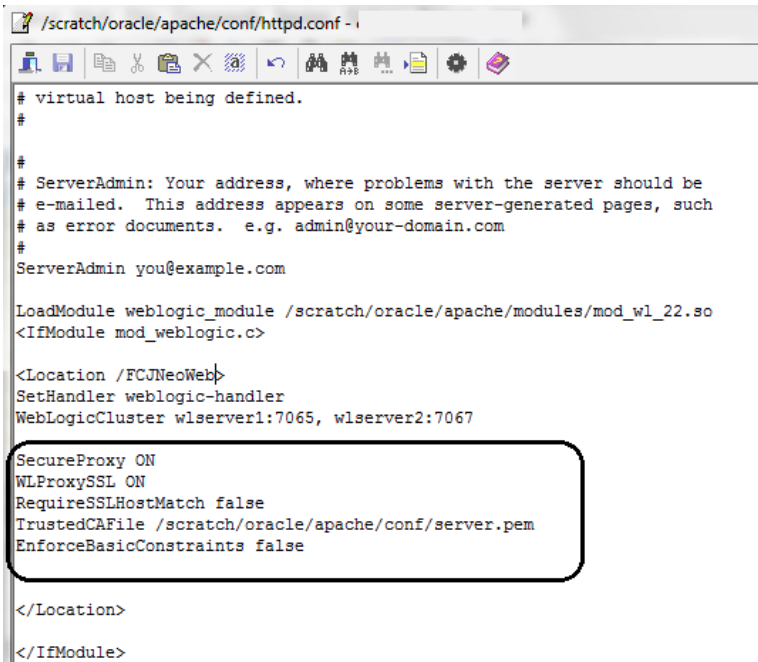
```
WLProxySSL ON
```

```
RequireSSLHostMatch false
```

```
TrustedCAFile /scratch/oracle/apache/conf/server.pem
```

```
EnforceBasicConstraints false
```

```
</Location>
```



```
/scratch/oracle/apache/conf/httpd.conf - i
# virtual host being defined.
#
#
# ServerAdmin: Your address, where problems with the server should be
# e-mailed. This address appears on some server-generated pages, such
# as error documents.  e.g. admin@your-domain.com
#
ServerAdmin you@example.com

LoadModule weblogic_module /scratch/oracle/apache/modules/mod_wl_22.so
<IfModule mod_weblogic.c>

<Location /FCJNeoWeb>
SetHandler weblogic-handler
WebLogicCluster wlserver1:7065, wlserver2:7067

SecureProxy ON
WLProxySSL ON
RequireSSLHostMatch false
TrustedCAFile /scratch/oracle/apache/conf/server.pem
EnforceBasicConstraints false

</Location>
</IfModule>
```

4. Restart Apache and access using the URL

<https://<hostname>:<port>/FCJNeoWeb/>

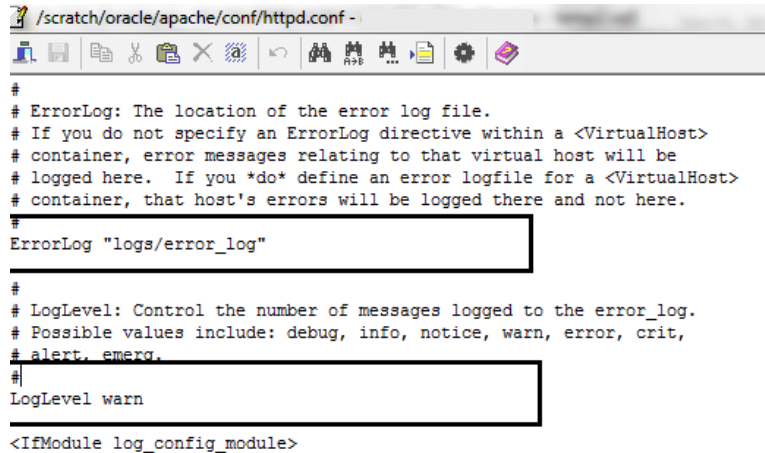
5. Starting, Stopping, and Restarting Apache

Run below commands

- `$PREFIX/bin/apachectl start`
- `$PREFIX/bin/apachectl stop`
- `$PREFIX/bin/apachectl restart`

6. Debugging

1. LogLevel and ErrorLog directives in httpd.conf file control the location and log file severity.



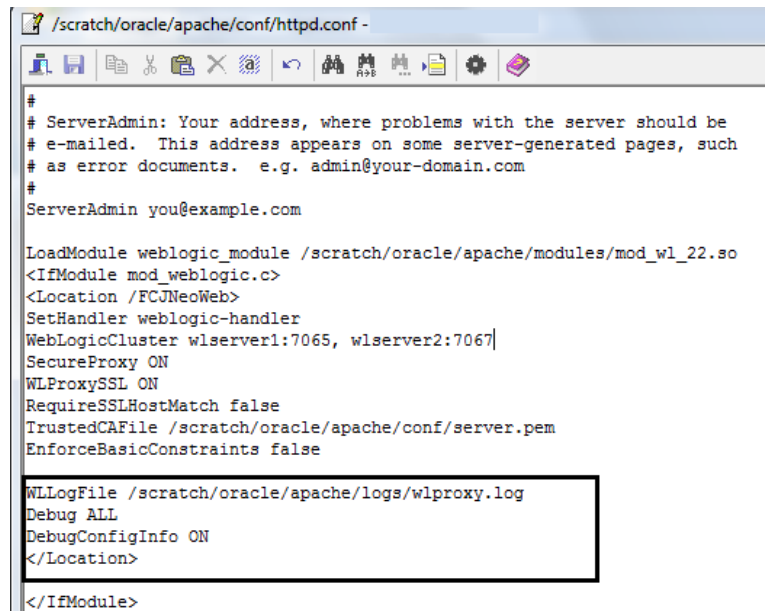
```
#  
# ErrorLog: The location of the error log file.  
# If you do not specify an ErrorLog directive within a <VirtualHost>  
# container, error messages relating to that virtual host will be  
# logged here.  If you *do* define an error logfile for a <VirtualHost>  
# container, that host's errors will be logged there and not here.  
#  
ErrorLog "logs/error_log"  
#  
# LogLevel: Control the number of messages logged to the error_log.  
# Possible values include: debug, info, notice, warn, error, crit,  
# alert, emerg.  
#  
LogLevel warn  
  
<IfModule log_config_module>
```

2. To enable debugging for communication between apache and weblogic add the following directives under Location tag

WLLogFile <File path>

Debug ALL

DebugConfigInfo ON



```
#  
# ServerAdmin: Your address, where problems with the server should be  
# e-mailed.  This address appears on some server-generated pages, such  
# as error documents.  e.g. admin@your-domain.com  
#  
ServerAdmin you@example.com  
  
LoadModule weblogic_module /scratch/oracle/apache/modules/mod_wl_22.so  
<IfModule mod_weblogic.c>  
<Location /FCJNeoWeb>  
SetHandler weblogic-handler  
WebLogicCluster wlserver1:7065, wlserver2:7067|  
SecureProxy ON  
WLLogFile /scratch/oracle/apache/logs/wlproxy.log  
Debug ALL  
DebugConfigInfo ON  
</Location>  
  
</IfModule>
```



Apache Server Configuration for FLEXCUBE
[November] [2022]
Version 14.6.2.0.0

Oracle Financial Services Software Limited
Oracle Park
Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India

Worldwide Inquiries:
Phone: +91 22 6718 3000
Fax: +91 22 6718 3001
www.oracle.com/financialservices/

Copyright © 2007, 2022, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.