WebServices Installation Guide

Oracle Financial Services Lending and Leasing

Release 14.3.0.1.0

Part No. E79024-01

September 2016

ORACLE

WebServices Installation Guide
September 2016
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/

Table of Contents

1. Preface

This document contains notes and installation steps needed to install WebServices. Oracle Financial Services Lending and Leasing relies on several pieces of Oracle software in order to run and this document is in no way meant to replace Oracle documentation supplied with the WebServices product or available via Oracle technical support. The purpose of this document is only meant to supplement the Oracle documentation and to provide Oracle Financial Services Lending and Leasing specific installation instructions.

For recommendations on security configuration, refer Security Configuration Guide.

It is assumed that anyone installing Oracle Financial Services Lending and Leasing will have a thorough knowledge and understanding of WebServices.

This guide covers the following processes.

- Installing WebServices Database Objects
- Deploying Application Interface WebServices
- Configuring Weblogic Policy on WebServices

1.1 Prerequisites

1. Download and Install the Oracle WebLogic Server 12c Version 12.2.1.0.0 from http://www.oracle.com/technetwork/middleware/weblogic/downloads/wls-main-097127.html). They are available from the following sources:
   - Oracle Software Delivery Cloud (http://edelivery.oracle.com/)
   - Oracle Technology Network (OTN)
2. It is assumed that the Oracle Financial Services Lending and Leasing DB is installed and configured, before running the WebServices installer.

1.2 Audience

This document is intended for system administrators or application developers who are installing Oracle Financial Services Lending and Leasing Application.

1.3 Conventions Used

Term	Refers to
Application	Oracle Financial Services Lending and Leasing

2. Installing WebServices Database Objects

Download and unzip the the WebServices database - ofsllxws.zip.

Run $ ./installofsllxwsdb.sh

```
-bash-4.1$ ./installofsllxwsdb.sh
```

This installer adds the required tables and packages to the apllication database.

```
Oracle Financial Services Lending and Leasing Webservices DB Installer

   Important Note:
   ---------------------------------------------------------------------------

   It is expected to have the OFSLL Application DB been installed and configured
   before running this installer.

   This installer adds the required tables and packages to the same OFSLL DB schema.


Continue? [y/n]: y
```

Enter ‘y’ when prompted to continue.

```
Oracle Financial Services Lending and Leasing Webservices DB Installer

   Important Note:
   ---------------------------------------------------------------------------

   It is expected to have the OFSLL Application DB been installed and configured
   before running this installer.

   This installer adds the required tables and packages to the same OFSLL DB schema.


Continue? [y/n]: y

Enter the Oracle Financial Services Lending and Leasing
Home Path? (usually /home/ofsll): /scratch/work_area/DEV/OFSLLREL

OFSLLHOME=/scratch/work_area/DEV/OFSLLREL
Okay? [y/n]: y
```

Script Prompts	Description and Action Required
Oracle Financial Services Lending and Leasing Home Path	Enter the path to the home directory. This is referred to as $OFSLL_HOME. Enter ‘y’ when prompted for.

```
Oracle Financial Services Lending and Leasing Webservices DB Installer

   Important Note:
   ---------------------------------------------------------------------------

   It is expected to have the OFSLL Application DB been installed and configured
   before running this installer.

   This installer adds the required tables and packages to the same OFSLL DB schema.


Continue? [y/n]: y

Enter the Oracle Financial Services Lending and Leasing
Home Path? (usually /home/ofsll): /scratch/work_area/DEV/OFSLLREL

OFSLLHOME=/scratch/work_area/DEV/OFSLLREL
Okay? [y/n]: y

Enter the Oracle DB Home Path? /scratch/app/db12c/product/12.1.0/dbhome_1

ORAHOME=/scratch/app/db12c/product/12.1.0/dbhome_1
Okay? [y/n]: y

Enter the Oracle SID? ORCL
INSTANCENAME=ORCL
Okay? [y/n]: y
```

Script Prompts	Description and Action Required
Oracle DB Home Path	Enter the path to the Oracle DB home directory. This is referred to as $ORACLE_HOME. Enter ‘y’ when prompted for.
Oracle SID	Enter the Name of Oracle Instance. Enter ‘y’ when prompted for.

```
   Important Note:
   ---------------------------------------------------------------------------

   Here is a list of CRITICAL environment variables and their settings:

   PATH=/usr/lib64/qt-3.3/bin:/usr/kerberos/sbin:/usr/kerberos/bin:/bin:/usr/bin:/usr/dev_infra/platform/bin:/usr/dev_infra/generic/bin:
11R6/bin:/usr/local/ade/bin:/scratch/app/db12c/product/12.1.0/dbhome_1/bin
   ORACLE_HOME=/scratch/app/db12c/product/12.1.0/dbhome_1
   ORACLE_SID=ORCL
   OFSLL_HOME=/scratch/work_area/DEV/OFSLLREL

   With the above environment, you should be able start SQLPlus and connect
   to the database.  If you cannot, correct the environment and restart the
   script to continue.


Continue? [y/n]: y
```

The script lists and sets the CRITICAL environment variables

```
   Important Note:
   ---------------------------------------------------------------------------

   Here is a list of CRITICAL environment variables and their settings:

   PATH=/usr/lib64/qt-3.3/bin:/usr/kerberos/sbin:/usr/kerberos/bin:/bin:/usr/bin:/usr/dev_infra/platform/bin:/usr/dev_infra/generic/bin
11R6/bin:/usr/local/ade/bin:/scratch/app/db12c/product/12.1.0/dbhome_1/bin
   ORACLE_HOME=/scratch/app/db12c/product/12.1.0/dbhome_1
   ORACLE_SID=ORCL
   OFSLL_HOME=/scratch/work_area/DEV/OFSLLREL

   With the above environment, you should be able start SQLPlus and connect
   to the database.  If you cannot, correct the environment and restart the
   script to continue.


Continue? [y/n]: y
.........
Oracle Financial Services Lending and Leasing Webservices Database Object Installation

The following items are available for installation:

                      1.  database types          (173)
                      2.  database tables         (47)
                      3.  database views          (157)
                      4.  database trigger        (4)
                      5.  database package specs  (254)
                      6.  database package bodies (262)
                      7.  database indexes        (30)
                      8.  System Seed Data        (0)


Continue with Installation? [y/n] : y
```

Enter ‘y’ when prompted to continue. A list of items available for installation are listed. Enter ‘y’ when prompted to ‘Continue with Installation’.

```
Oracle Financial Services Lending and Leasing Webservices Database Object Installation

The following items are available for installation:

                      1.  database types          (173)
                      2.  database tables         (47)
                      3.  database views          (157)
                      4.  database trigger        (4)
                      5.  database package specs  (254)
                      6.  database package bodies (262)
                      7.  database indexes        (30)
                      8.  System Seed Data        (0)


Continue with Installation? [y/n] : y

Log files will be located in /scratch/work_area/DEV/OFSLLREL/logs/ofsll_xws_install_logs
Press Enter to Continue...

Enter the Oracle userid (schema name) that will own the Oracle Financial Services Lending and Leasing
objects? (usually ofsllprd): OFSLLREL

Enter the password for this userid:
```

Sets the path for the location of log files. Press ‘Enter’ to continue.

Script Prompts	**Description and Action Required**
Oracle User ID that will own the Oracle Financial Services Lending and Leasing objects	Valid User ID
Password for this User ID	Valid Password

The script installs the objects.

```
PL/SQL procedure successfully completed.


PL/SQL procedure successfully completed.


PL/SQL procedure successfully completed.


PL/SQL procedure successfully completed.


PL/SQL procedure successfully completed.


Commit complete.


Recompiling Invalid Objects...
mv: cannot stat `/tmp/recomp_obj..23135.log': No such file or directory

Oracle Financial Services Lending and Leasing Webservices DB Object Installation Com
```

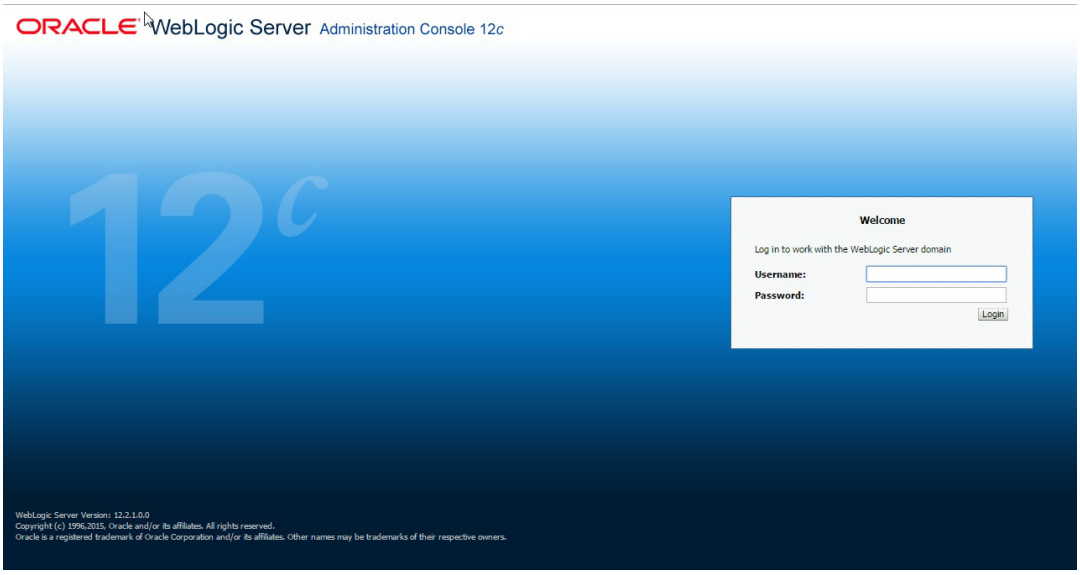
While installing, the script recompiles the invalid objects and completes the installation of DB objects.

3. Deploying Application Interface WebServices

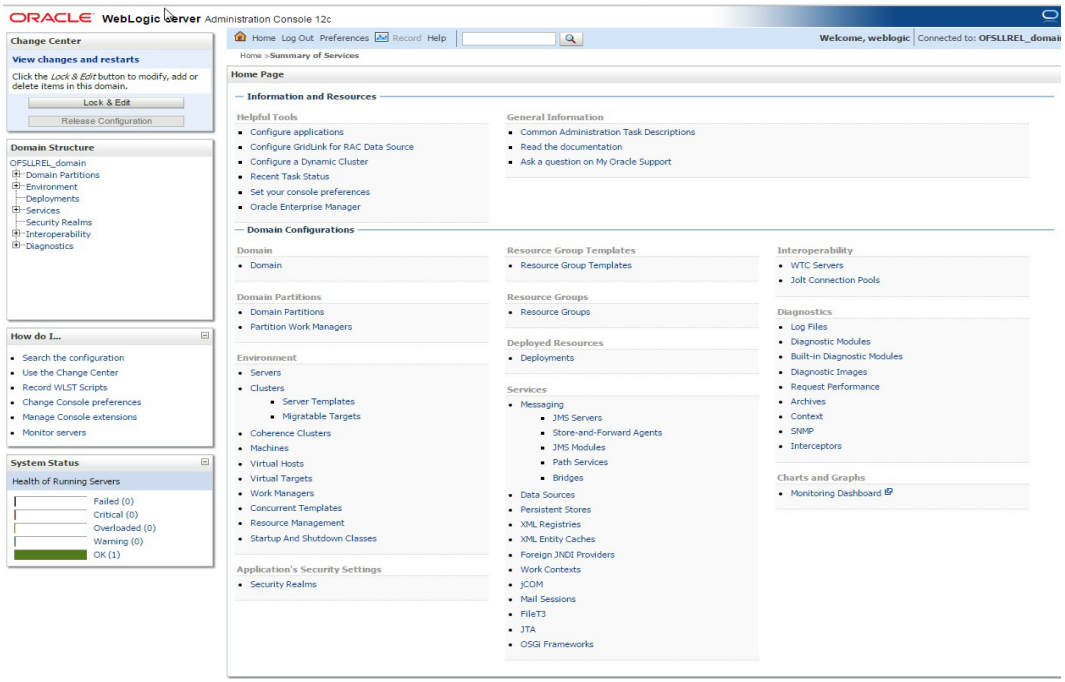
3.1 Creating Data Sources for WebServices

Create a new server on the console which will be specifically for WebServices.

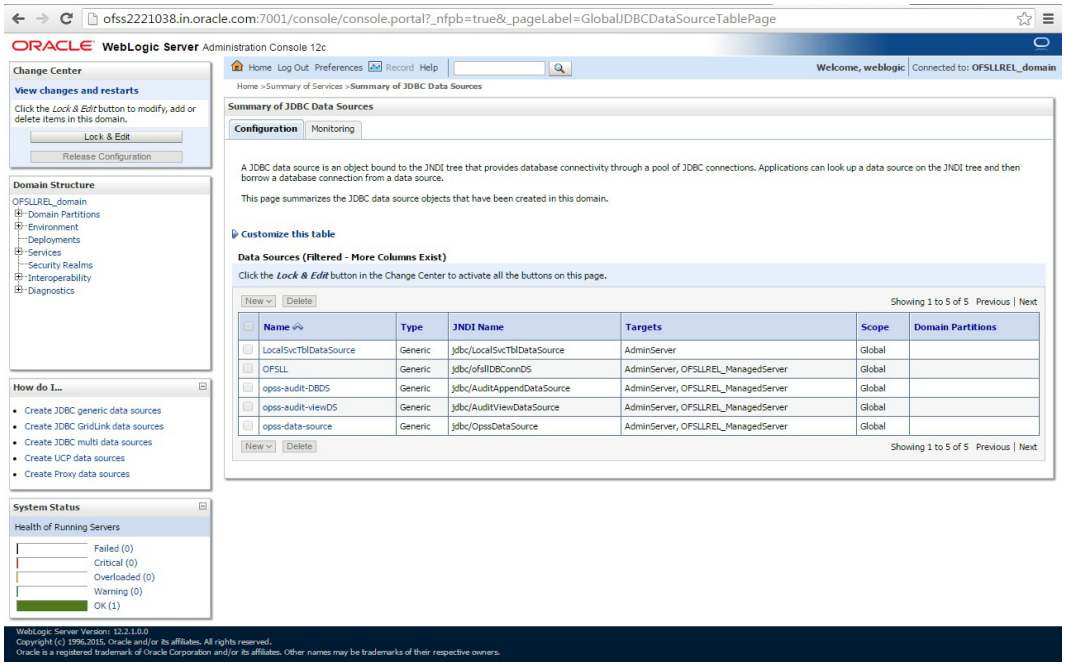
1. Login to WebLogic Server 12c console (http://hostname:port/console).



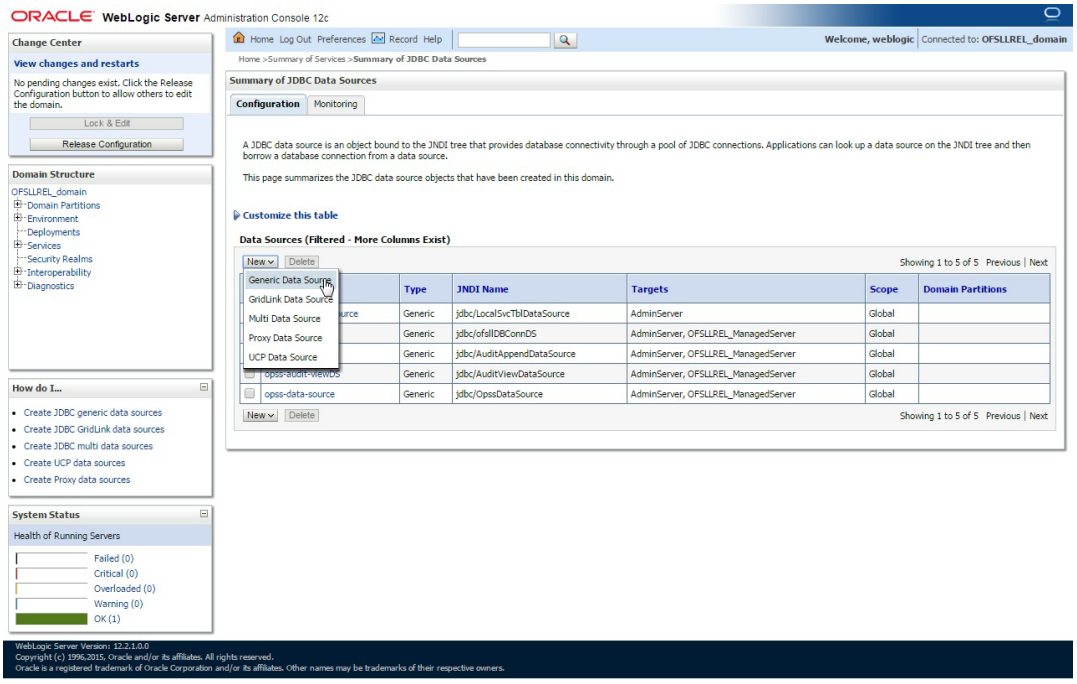
2. The following window is displayed.



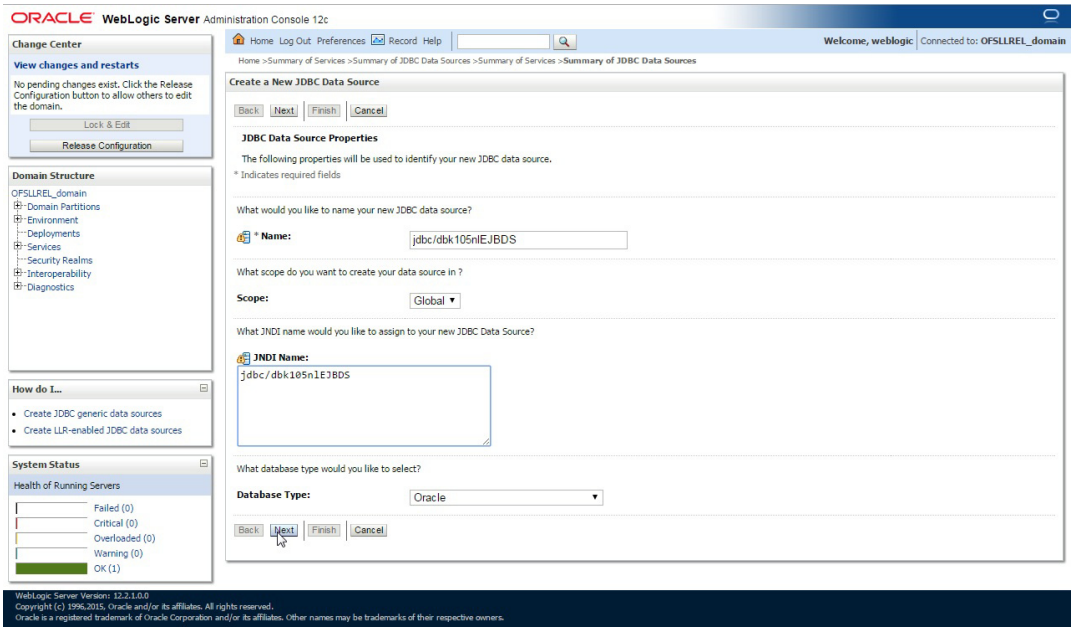
3. Click Domain Name > Services > Data Sources. The following window is displayed.



4. Click ‘Lock & Edit’ button on the left panel. Click ‘New’ on right panel and select ‘Generic Data Source’.

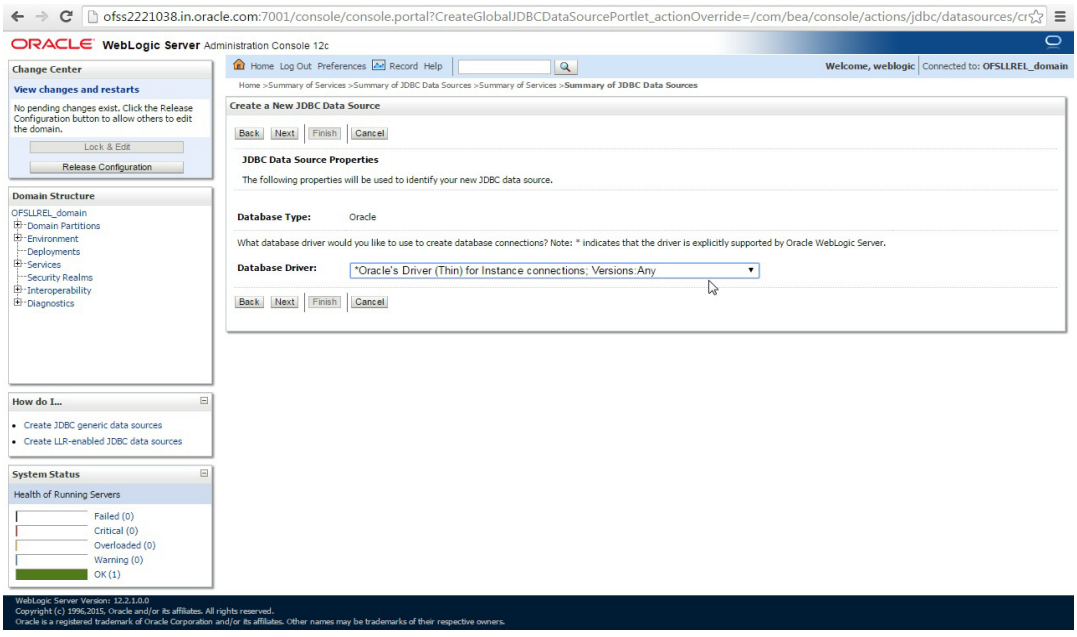


5. Specify the following details:

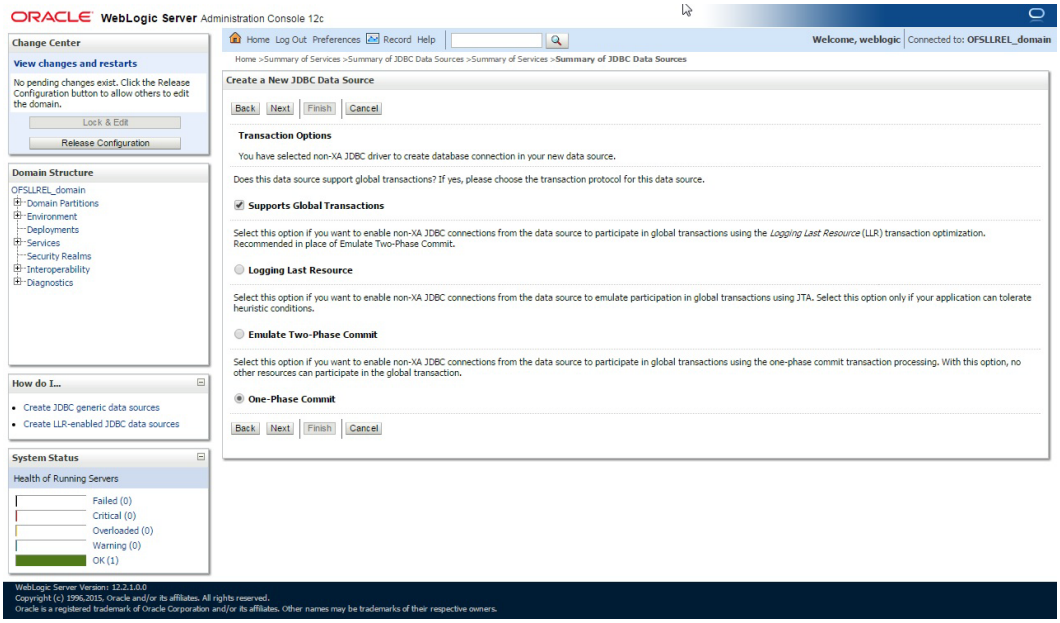


- Enter Data source Name.
- Enter JNDI Name as ‘jdbc/dbk105nlEJBDS’.
- Select ‘Oracle’ as Database Type.

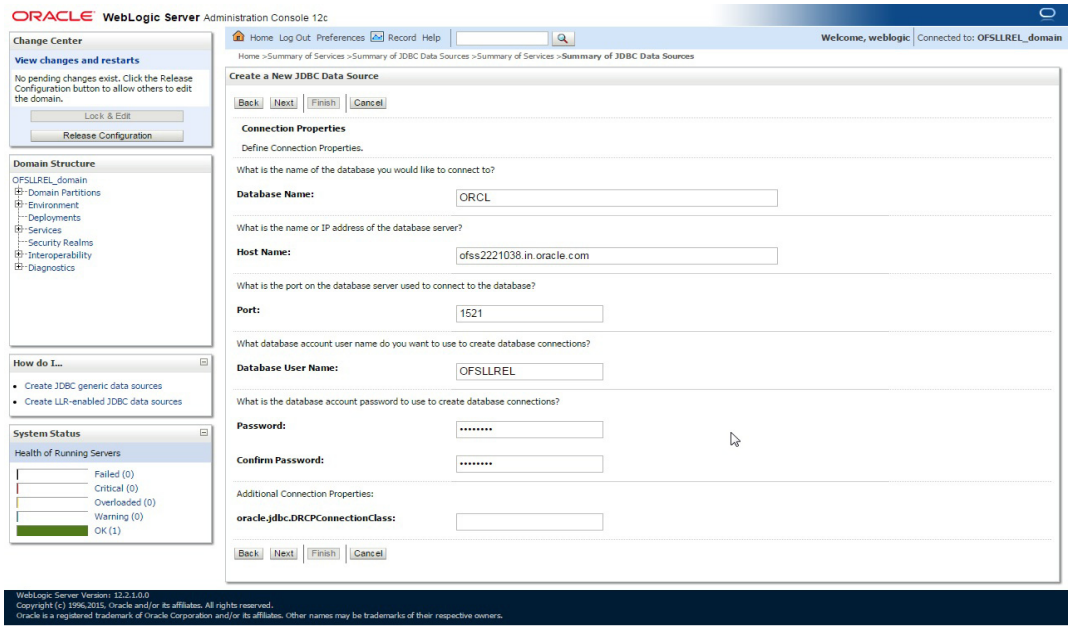
6. Click ‘Next’. The following window is displayed.
7. Select the Database Driver ‘Oracle's Driver(Thin) for Instance connections;Versions:Any’ as shown.



8. Click ‘Next’. The following window is displayed.

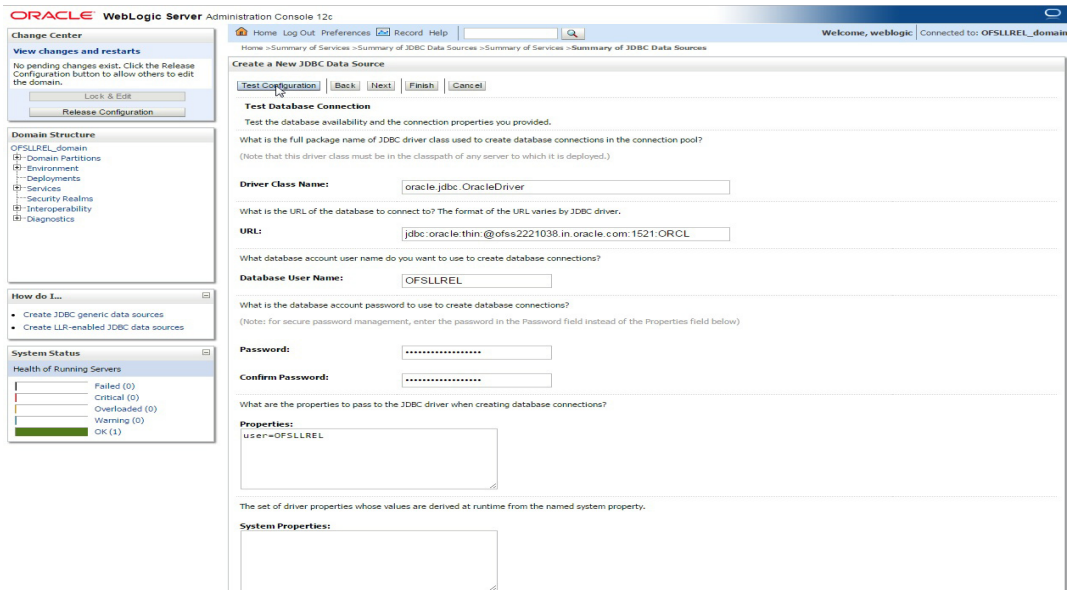


9. Click ‘Next’. The following window is displayed.

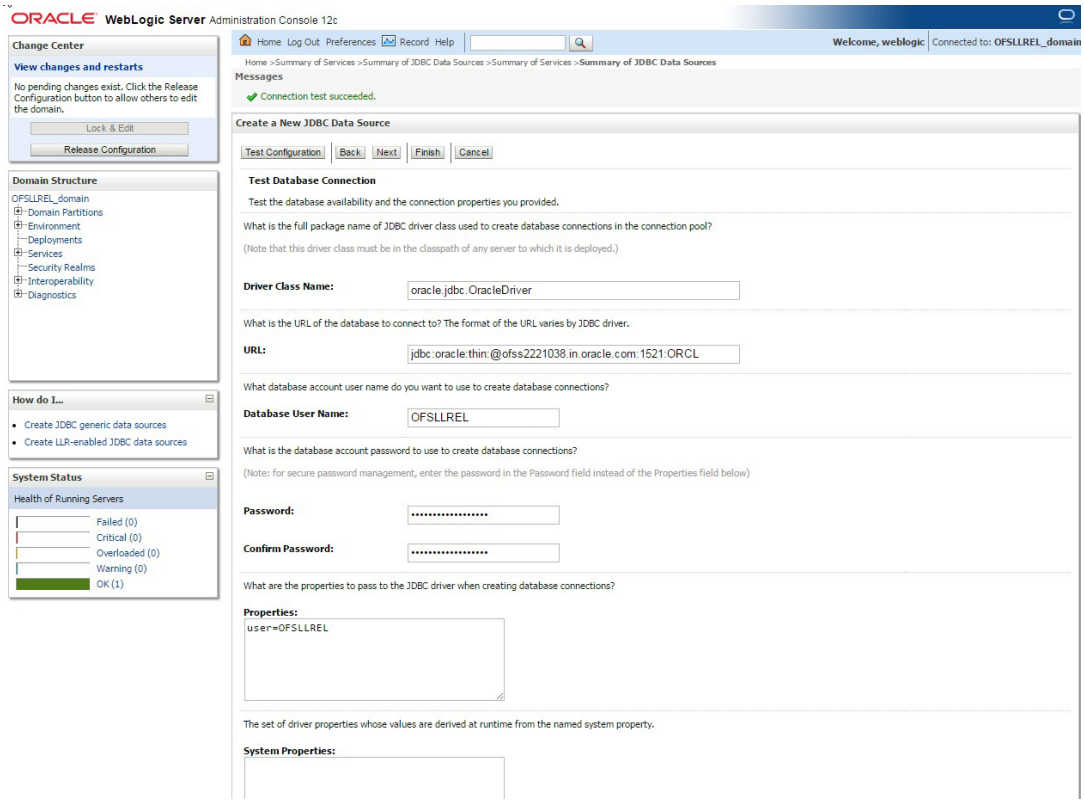


10. Enter the Database details.

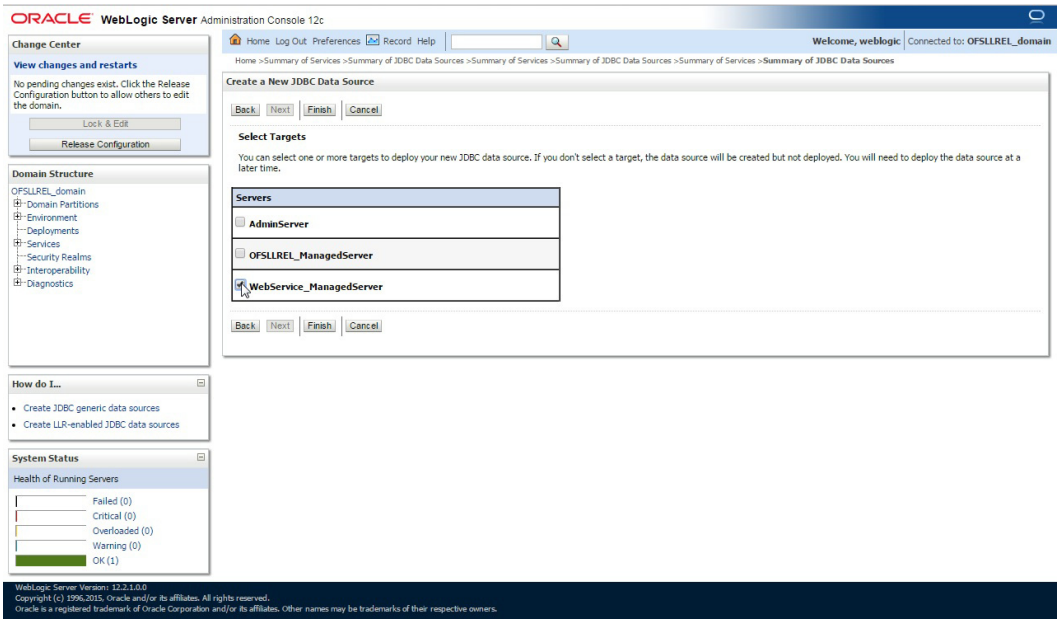
11. Click ‘Next’. The following window is displayed.



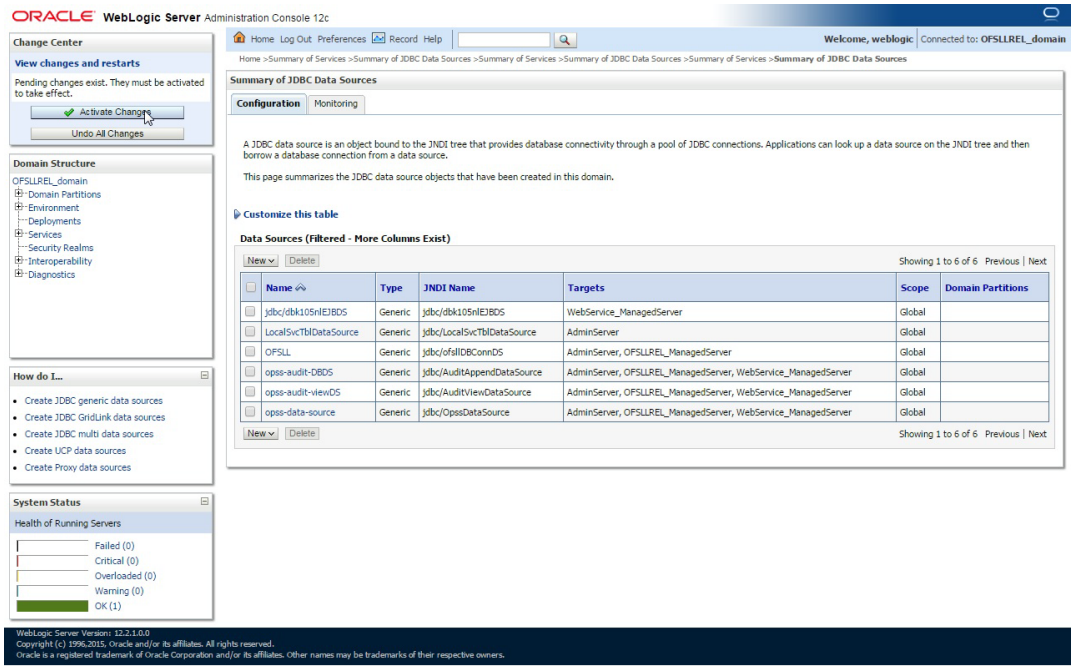
12. Click ‘Test Configuration’. The following window is displayed indicating a confirmation message as ‘Connection test succeeded’.



13. Click ‘Next’. The following window is displayed.



14. Select target WebServices Server and click ‘Finish’. The following window is displayed.

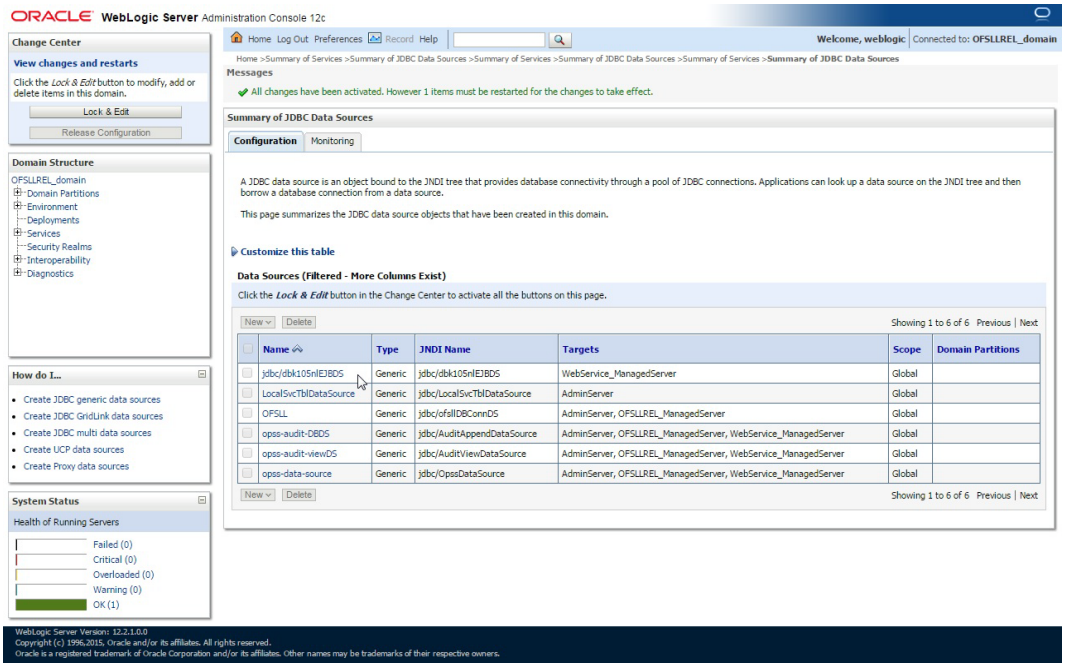


15. Click ‘Activate Changes’.

Similarly follow the above steps to create the following data sources:

– jdbc/dbkwsDS

– jdbc/IN1HukWznG0b4esj



3.2 Working with SSL

It is not recommended to run OFSLL WebServices with the test certificates in production. You have to get:

1. Vendor public production key/certificates for SSL handshake. These have to be imported into weblogic truststore.
2. RO public production key/certificates to validate digital signature in the RO inputs. This has to be imported into dls_cacerts keystore mentioned in the configuration file.
3. Generate production grade public/private key signed by appropriate CA. The public key has to be shared with RO so that they can validate digital signature in OFSLL requests. The corresponding private key should not be shared, should be imported into dls_cacerts keystore mentioned in the configuration file and used to add the digital signature in RO requests.

Additional Notes

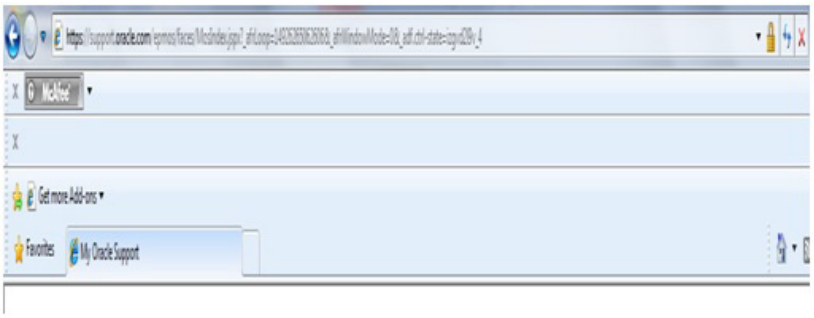
- While testing with the test certificates, you may get error: '**Signature verification failed because RSA key public exponent [3] is too small**'.
  - As a fix, you need to add the following in the start-up script: 'Dweblogic.security.SSL.allowSmallRSAExponent=true'
- You may encounter error: '**java.security.InvalidKeyException: Illegal key size or Cipher not initialize**'.
  - As a fix add the following in the start-up script: 'Dweblogic.security.SSL.nojce=true'
- You may encounter error: '**java.security.InvalidAlgorithmParameterException: the trustAnchors parameter must be non-empty**'.
  - As a fix remove the '-DUseSunHttpHandler=true' SSL option if any from the startup script.

3.2.1 Steps to Import Certificates to keystore

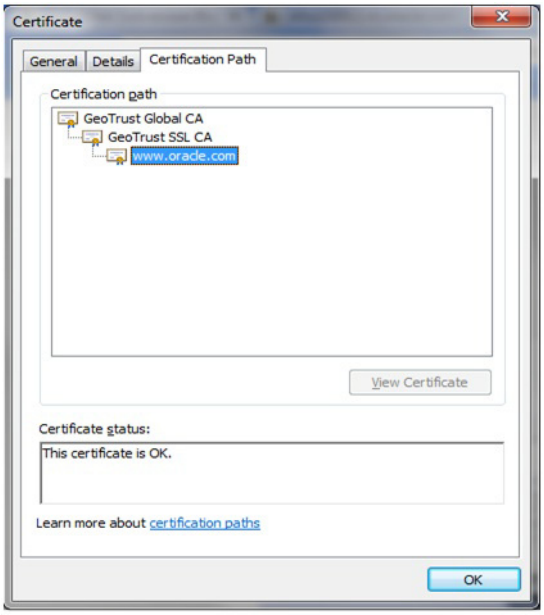
1. The JKS(dls_cacerts) should be available under /WEB-INF/classes/config.

2. Save all the certificates from the vendor website. Note to save the certificates in "Base-64 encoded X.509(.CER)" and with extn .cer. Steps for saving certificates from the vendor website:

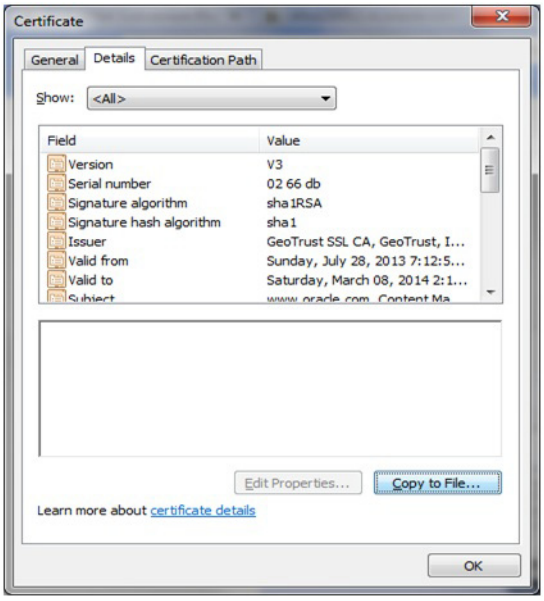
- Click on the padlock and choose ‘View Certificates’.



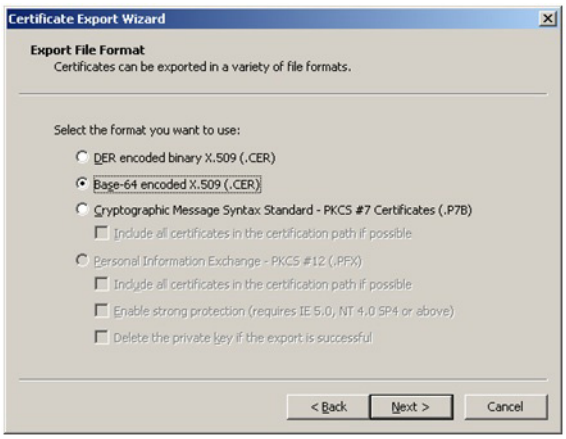
- Click on the Certification Path tab and select the certificate with the ‘Name’ and ‘Issued To’ the same as the user-defined ID.



- Go to the Details tab.



- Click on the ‘Copy to File’ button, and click ‘Next’. Select ‘Base-64 encoding’ and save the file somewhere on your local computer.



3. Run following command to import certificate into JKS(dls_cacerts)
   – "keytool -keystore <Key store Path> /dls_cacerts -import -trustcacerts -file <Certificate location in file system>\xxx.cer -alias <alias as mentioned in config file>
4. Run following command to view details of certificate like expiration date of certificate etc.
   – "keytool -list -rfc -alias alias <alias as mentioned in config file> -keystore <Key store Path> /dls_cacerts
   – "keytool -list -v -alias alias <alias as mentioned in config file> -keystore <Key store Path> /dls_cacerts

For more details on keytool refer the link: http://docs.oracle.com/javase/7/docs/technotes/tools/windows/keytool.html

3.2.2 Enable SSL Debugging

SSL debugging can be enabled by adding the following to managed server start-up script: ‘Dssl.debug=true’

3.2.3 Connecting to service supporting only TLS protocol

For WLS 12c by default (acting as a client) will send sslv2 hello for the SSL handshake to TLS service. The TLS service will not respond to SSLv2 hello and the connection will be dropped.

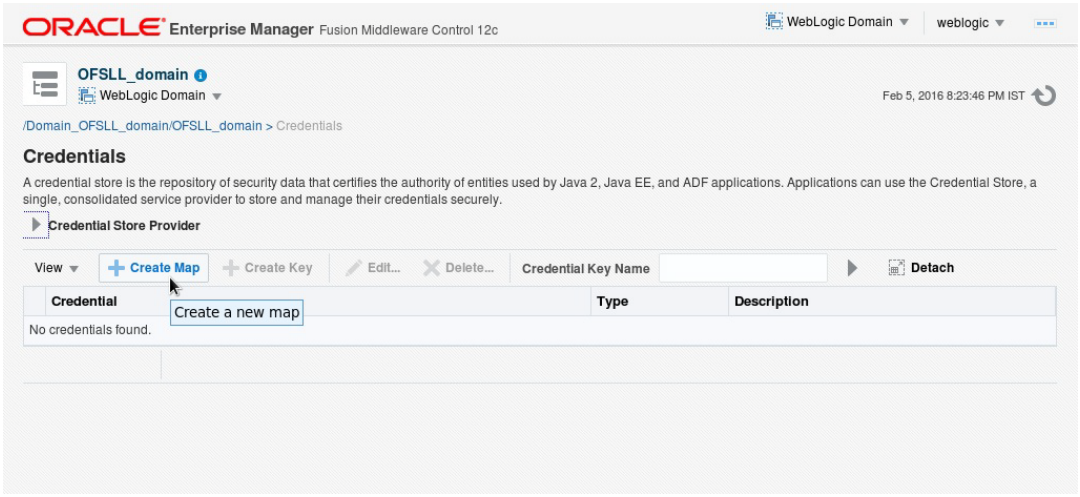
The fix for the problem is to set Dweblogic.security.SSL.protocolVersion=TLS1 at the Managed server level in which the WebServices have been deployed.

3.3 Create RouteOne Credentials and System Policies

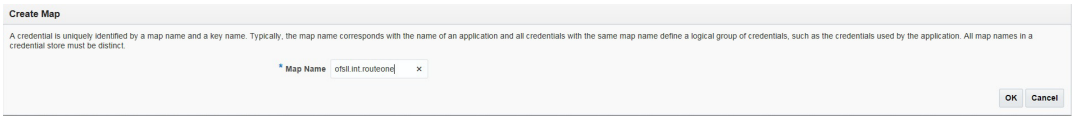
In order Configure RouteOne, you need to create credentials and system policies.

1. Login to Oracle Enterprise Manager 12c (http://hostname:port/em).

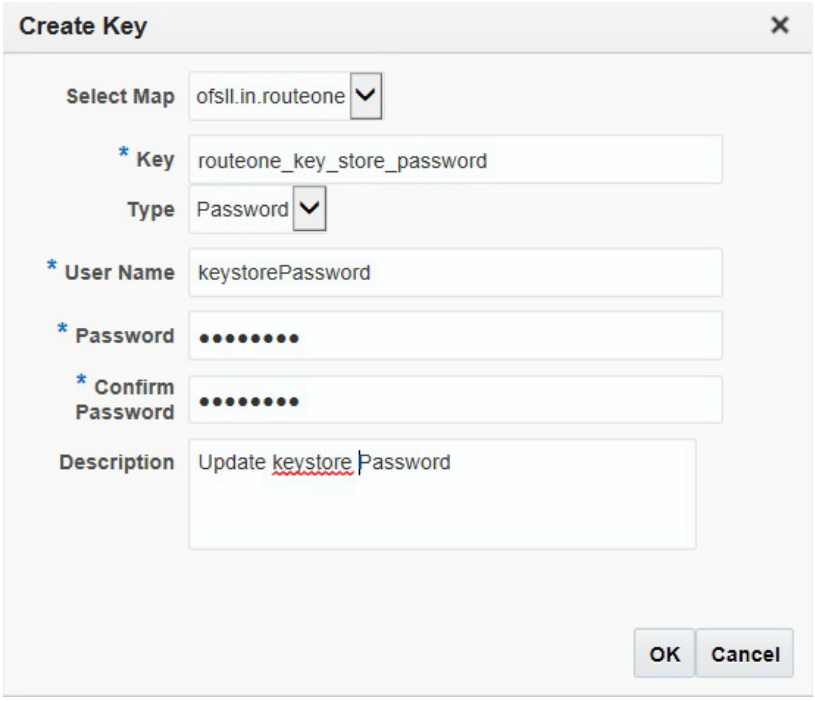
2. On the left panel, right click on OFSLL_domain and select Security > System Policies > Credentials. The following window is displayed.



3. Click ‘Create Map’. The following window is displayed.



4. Enter Map Name as ‘ofsll.int.routeone’ and click ‘OK’.
5. Click ‘Create Key’. The following window is displayed.



6. Click ‘OK’.

Similarly you need to create the following Maps and corresponding keys as indicated in following table.

Maps	Key Name	Username	Password	Description

ofsll.int.routeone	routeone_key_store_password	keystore-Password	changeit	The keystore password
	routeone_key_alias	roKeyAlias	routeone_pub_tst	RouteOne's public key alias name.The public key is needed to verify the xml signature of the request.
	routeone_sscro_key_alias	sscroKeyAlias	ssc_routeone	OFSLL's private key alias name. The private key is used to sign xml response to RouteOne
	routeone_sscro_key_passsword	sscroKey-Password	demotestSSCR1	OFSLL's private key password

You need to provide access permission for the below mapping. For details on how to set the access permission, refer to point 9 in 'Create Credentials and System Policies' section.

Permission Class	Resource Name	Permission Actions
oracle.security.jps.service.credstore.CredentialAccessPermission	context=SYSTEM,mapName=ofsll.int.routeone,keyName=*	read

3.4 Deploying Webservices

3.4.1 Deployment of WebServices in Enterprise Manager

1. Download and unzip the WebServices - ofsllxws.zip.
2. Following is a mapping of which EAR is needed for which service:

dbkls-xxws.ws.app.ear	**dbkls-xxws.ws.app-dt.ear**	**dbkls-xxws.ws.app-ds.ear**	**dbkls-xxws.ws.app-ro.ear**
DialerIntegrator: Service to integrate with third party dialer systems	**DT Interface**: To receive loan application from dealer track	**DS Interface**: To receive edocs application update	**RO Interface**: To receive loan application from route one
CSSAccountDetailsRequestService: Service to fetch Account Details by account number	**DT Interface**: To receive loan application/deal update from dealer track	**DS Interface**: To receive edocs comment update	**RO Interface**: To receive loan application/deal refresh from route one

CSSAccount-SearchRequest-Service: Service to fetch Account Details by attributes such as AccountNumber, CustomerFirst-Name, Customer-LastName, SSN	**DT Interface**: To receive comments from dealer track	**DS Interface**: To receive edocs location update	**RO Interface**: To receive comments from route one
CSSPostTransactionsRequestService: To post different transactions such as update address, update phone number, update ACH details, etc	**LOSPostStatusRequestService**: To post comments to dealer track		**LOSPostStatus-RequestService**: To post comments to route one
CSSAccountPayOffQuoteRequest-Service: To get the account payoff quote	**LOSPostStatusRequestService**: To post application status to dealer track		**LOSPostStatus-RequestService**: To post application status to route one
LOSeApplication-RequestService:Edocs application update service	**ILOSPostDealerDetailsService**: To post dealer details to dealer track		**ILOSPostDealer-DetailsService**: To post dealer details to dealer track
LOSApplication-RequestService: Loan application entry service			
LOSApplication-SearchRequest-Service: Application Search service			
LOSApplication-SearchRequest-Service: Application Search service			
LOSApplication-CommentUpdate-Service: Edocs application comment update service			

LOSPostStatusRequestService: Service to post loan application status			
LOSApplicationUpdateService: Service to update an existing application			
LOSCheckStatusRequestService: Service to check application status			
LOSApplicationAccountWebService: Service to create Account number in OFSLL			
			LOSEContractService: To recieve contract information from RouteOne

3. Unzip all the ear files present:
   - dbkls-xxws.ws.app.ear
   - dbkls-xxws.ws.app-dt.ear
   - dbkls-xxws.ws.app-ro.ear
   - dbkls-xxws.ws.app-ds.ear

```
-bash-4.1$ unzip ofsllxws.zip -d ofsllxws
Archive:  ofsllxws.zip
  inflating: ofsllxws/dbkls-xxws.ws.app-ds.ear
  inflating: ofsllxws/dbkls-xxws.ws.app-dt.ear
  inflating: ofsllxws/dbkls-xxws.ws.app-ro.ear
  inflating: ofsllxws/dbkls-xxws.ws.app.ear
-bash-4.1$ cd ofsllxws
-bash-4.1$ ls
dbkls-xxws.ws.app-ds.ear  dbkls-xxws.ws.app-dt.ear  dbkls-xxws.ws.app.ear  dbkls-xxws.ws.app-ro.ear
-bash-4.1$ unzip dbkls-xxws.ws.app-ds.ear -d dbkls-xxws.ws.app-ds
Archive:  dbkls-xxws.ws.app-ds.ear
   creating: dbkls-xxws.ws.app-ds/META-INF/
  inflating: dbkls-xxws.ws.app-ds/META-INF/application.xml
  inflating: dbkls-xxws.ws.app-ds/dbkls-xws-web-ds.war
-bash-4.1$ cd dbkls-xxws.ws.app-ds
-bash-4.1$ ls
dbkls-xws-web-ds.war  META-INF
-bash-4.1$ unzip dbkls-xws-web-ds.war -d dbkls-xws-web-ds
Archive:  dbkls-xws-web-ds.war
   creating: dbkls-xws-web-ds/WEB-INF/
   creating: dbkls-xws-web-ds/WEB-INF/classes/
   creating: dbkls-xws-web-ds/WEB-INF/classes/com/
   creating: dbkls-xws-web-ds/WEB-INF/classes/com/ofss/
   creating: dbkls-xws-web-ds/WEB-INF/classes/com/ofss/fll/
   creating: dbkls-xws-web-ds/WEB-INF/classes/com/ofss/fll/utils/
   creating: dbkls-xws-web-ds/WEB-INF/classes/com/ofss/fll/xws/
   creating: dbkls-xws-web-ds/WEB-INF/classes/com/ofss/fll/xws/dialerintegration/
   creating: dbkls-xws-web-ds/WEB-INF/classes/com/ofss/fll/xws/exception/
   creating: dbkls-xws-web-ds/WEB-INF/classes/com/ofss/fll/xws/lookup/
   creating: dbkls-xws-web-ds/WEB-INF/classes/com/ofss/fll/xws/xae/
   creating: dbkls-xws-web-ds/WEB-INF/classes/com/ofss/fll/xws/xcl/
   creating: dbkls-xws-web-ds/WEB-INF/classes/com/ofss/fll/xws/xcs/
```

4. Open each of the unzipped files and unzip the war file.

```
-bash-4.1$ls -lart
total 7920
-rw-r--r-- 1 orafmw oinstall 1074660 Apr 10 08:47 dbkls-xxws.ws.app.ear
-rw-r--r-- 1 orafmw oinstall  996280 Apr 10 08:47 dbkls-xxws.ws.app-dt.ear
-rw-r--r-- 1 orafmw oinstall  991947 Apr 10 08:47 dbkls-xxws.ws.app-ro.ear
-rw-r--r-- 1 orafmw oinstall  978700 Apr 10 08:47 dbkls-xxws.ws.app-ds.ear
-rw-r----- 1 orafmw oinstall 4032431 Apr 10 15:09 ofsllxws.zip
drwxr----- 8 orafmw oinstall    4096 Apr 10 15:09 ..
drwxr-xr-x 3 orafmw oinstall    4096 Apr 10 16:26 dbkls-xxws.ws.app-ds
drwxr-xr-x 3 orafmw oinstall    4096 Apr 10 16:26 dbkls-xxws.ws.app-dt
drwxr-xr-x 3 orafmw oinstall    4096 Apr 10 16:26 dbkls-xxws.ws.app-ro
drwxr----- 6 orafmw oinstall    4096 Apr 10 16:26 .
drwxr-xr-x 3 orafmw oinstall    4096 Apr 10 16:26 dbkls-xxws.ws.app
-bash-4.1$cd dbkls-xxws.ws.app
-bash-4.1$ls

dbkls-xws-web.war  META-INF
-bash-4.1$unzip dbkls-xws-web.war
Archive:  dbkls-xws-web.war
   creating: WEB-INF/
   creating: WEB-INF/classes/
   creating: WEB-INF/classes/com/
   creating: WEB-INF/classes/com/ofss/
   creating: WEB-INF/classes/com/ofss/fll/
   creating: WEB-INF/classes/com/ofss/fll/utils/
   creating: WEB-INF/classes/com/ofss/fll/xws/
   creating: WEB-INF/classes/com/ofss/fll/xws/dialerintegration/
   creating: WEB-INF/classes/com/ofss/fll/xws/exception/
   creating: WEB-INF/classes/com/ofss/fll/xws/lookup/
```

5. It creates a WEB-INF file

```
-bash-4.1$ls
dbkls-xws-web.war  META-INF  WEB-INF
```

After unzipping the war file to directory, the directory will have the following structure

- dbkls-xxws.ws.app / dbkls-xxws.ws.app-ds / dbkls-xxws.ws.app-dt / dbklsxxws.ws.app-ro
- /dbkls-xxws.ws.app/dbkls-xws-web

WEB-INF (directory)

---->classes (directory)

- config

- class files (in package folders)

-----> lib (directory)

- jdom-2.0.6.jar

- OfsllCommonCSF.jar

- soap-2.3.1.jar

------> wsdl (directory)

- *.wsdl

-----> *-java-wsdl-mapping.xml

-----> web.xml

-----> weblogic.xml

-----> weblogic-webservices.xml

-----> weblogic-webservices-policy.xml

-----> webservices.xml

6. To edit the web interface config files, navigate to the above WEB-INF > classes > config. Edit the following configuration files with the application url and port.

   For complete details on configuration parameters, refer to "Appendix - Configuration parameters" section in Appendix chapter.

- ds_servlet_init.conf-This is the configuration file only for edocs servlet interface

```
#### XML schema to use when validating incoming application update
#### (comments) messages

#### URL for OFSLL eDocs web service

LOSeApplicationRequestServiceURL = http://<localhost>:<port>/dbkls-xws-app-ds/LOSeApplicationRequestService

LOSApplicationCommentUpdateServiceURL = http://<localhost>:<port>/dbkls-xws-app-ds/LOSApplicationCommentUpdateService

LOSApplicationLocationUpdateServiceURL = http://<localhost>:<port>/dbkls-xws-app-ds/LOSApplicationLocationUpdateService

#### URL for OFSLL application update web service
```

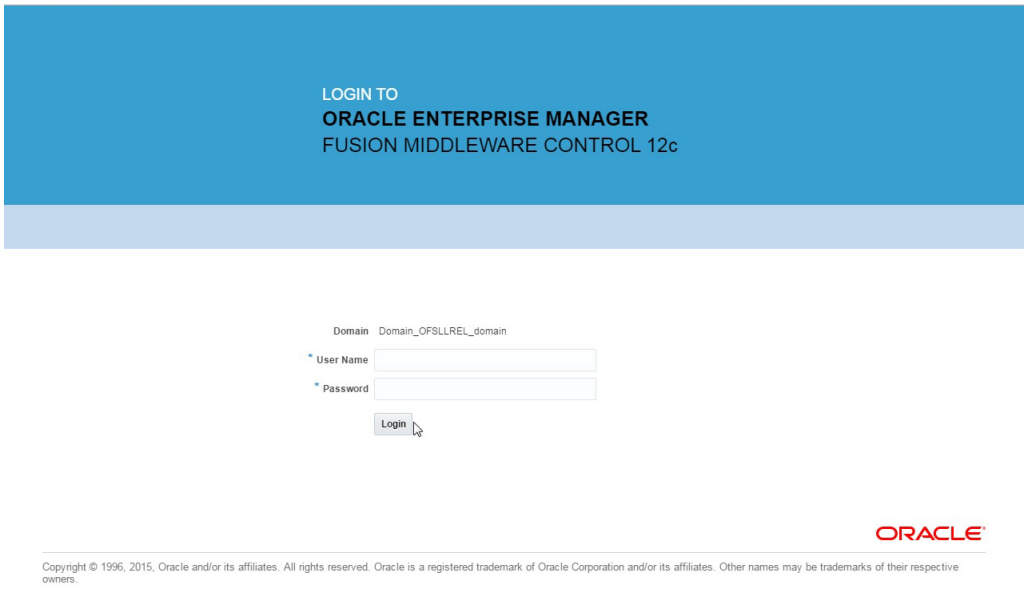
- dt_servlet_init.conf- This is the configuration file only for dealer track servlet interface

```
#### URL for OFSLL new application web service

LOSApplicationRequestServiceURL = http://<localhost>:<port>/dbkls-xws-app-dt/LOSApplicationRequestService


#### URL for OFSLL application update web service

LOSApplicationUpdateServiceURL = http://<localhost>:<port>/dbkls-xws-app-dt/LOSApplicationUpdateService
```

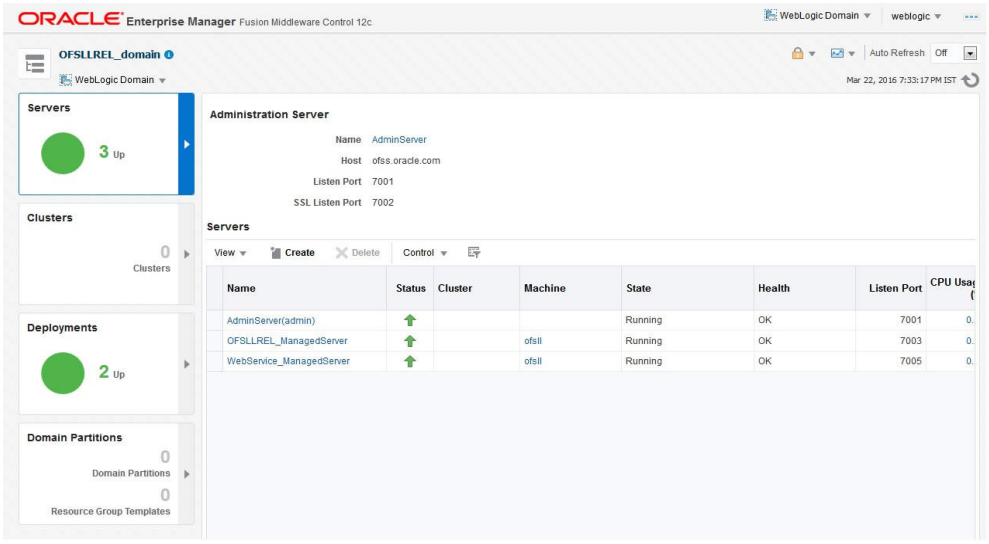
- ro_servlet_init.conf- This is the configuration file only for route one servlet interface

```
#### URL for OFSLL new application web service

LOSApplicationRequestServiceURL = http://<localhost>:<port>/dbkls-xws-app-ro/LOSApplicationRequestService

#### URL for OFSLL application update web service

LOSApplicationUpdateServiceURL = http://<localhost>:<port>/dbkls-xws-app-ro/LOSApplicationUpdateService

#### URL for OFSLL e contract web service

LOSeContractWebServiceServiceURL = http://<localhost>:<port>/dbkls-xws-app-ro/LOSEContractService
```

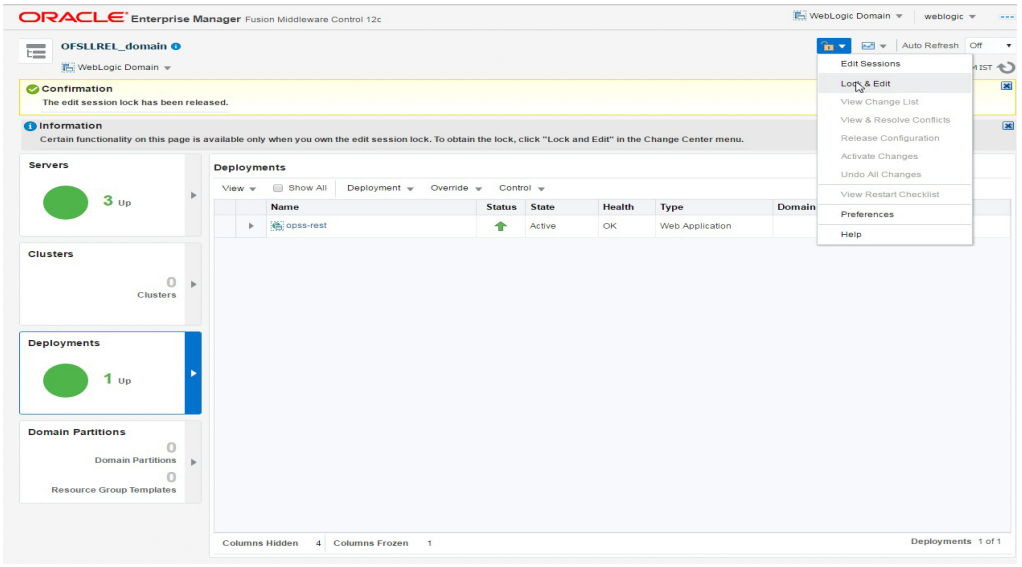
7. Login to Web Logic application server enterprise manager (e.g.: http://hostname:port/em)



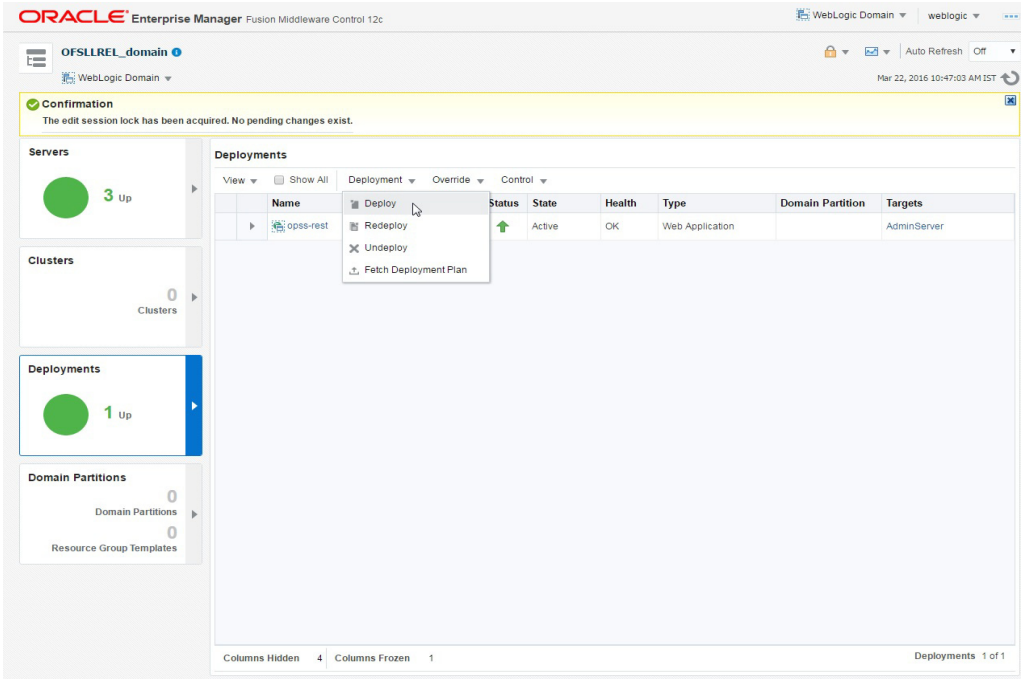
8. Enter valid login credentials. The following window is displayed.



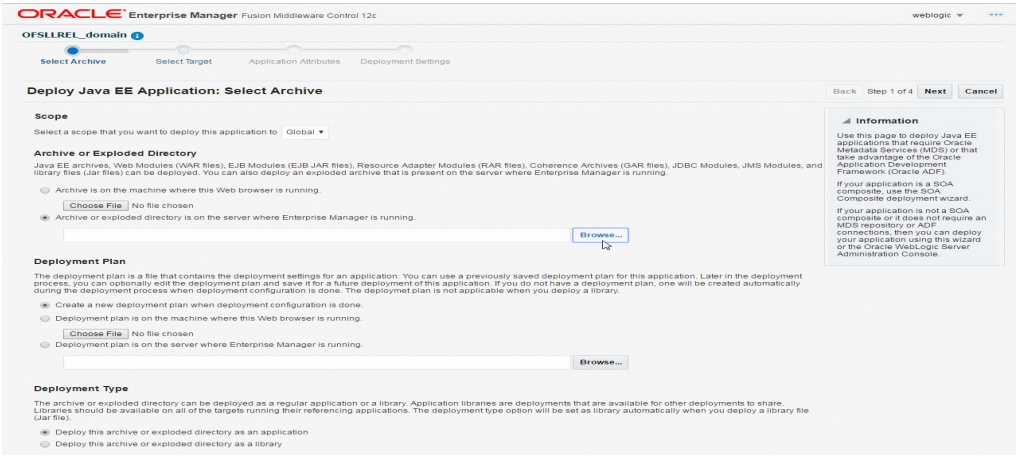
9. Expand the weblogic domain present in the left pane. The following window is displayed.



10. Right click on WebService_ManagedServer in left panel, select Application Deployment > Deploy.



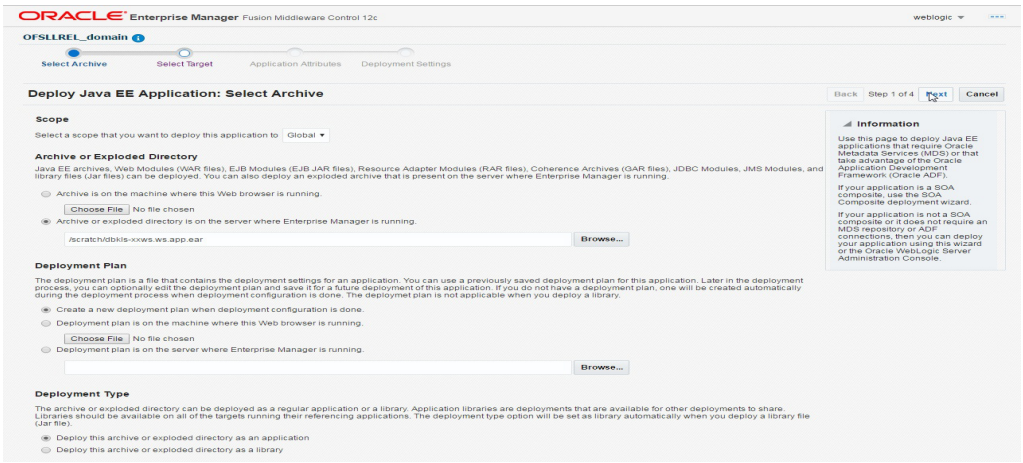
11. The following window is displayed.



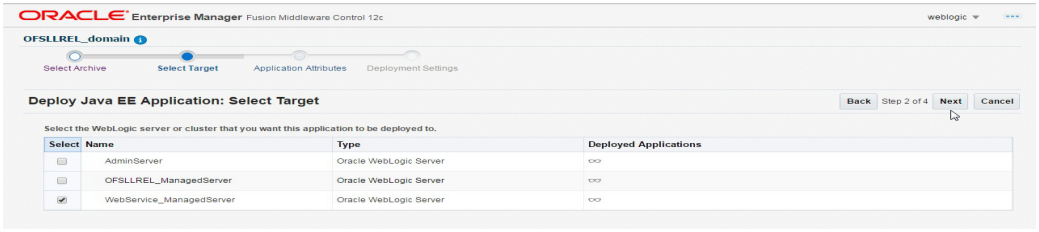
12. Browse to the folder containing the WebService.

    Eg: /scratch/ofsllxws/dbkls-xxws.ws.app

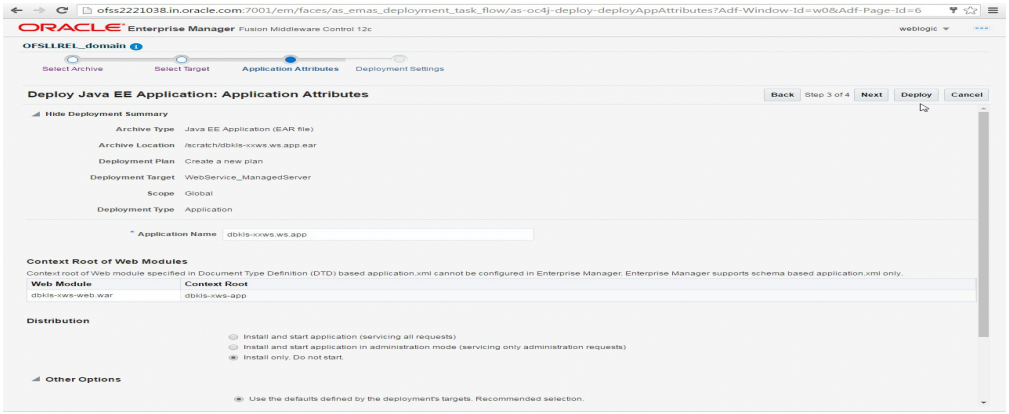
    Click ‘Next’.



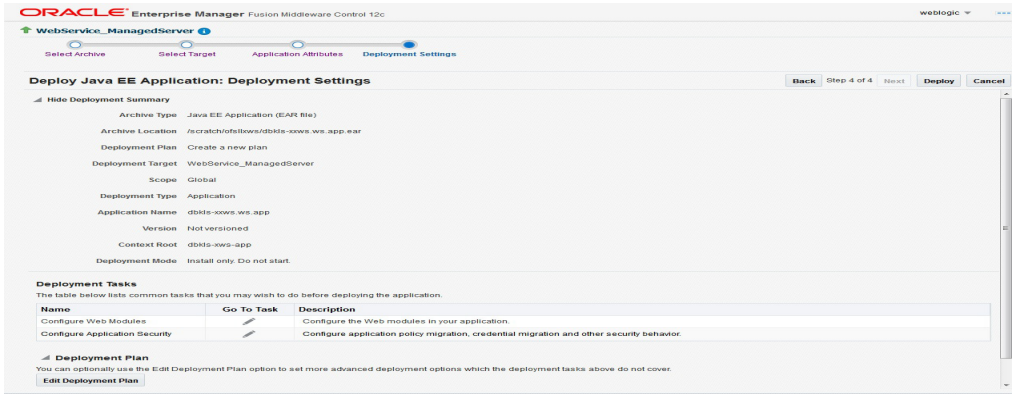
13. The following window is displayed. Select the server on which the WebService needs to be deployed. Click ‘Next’.



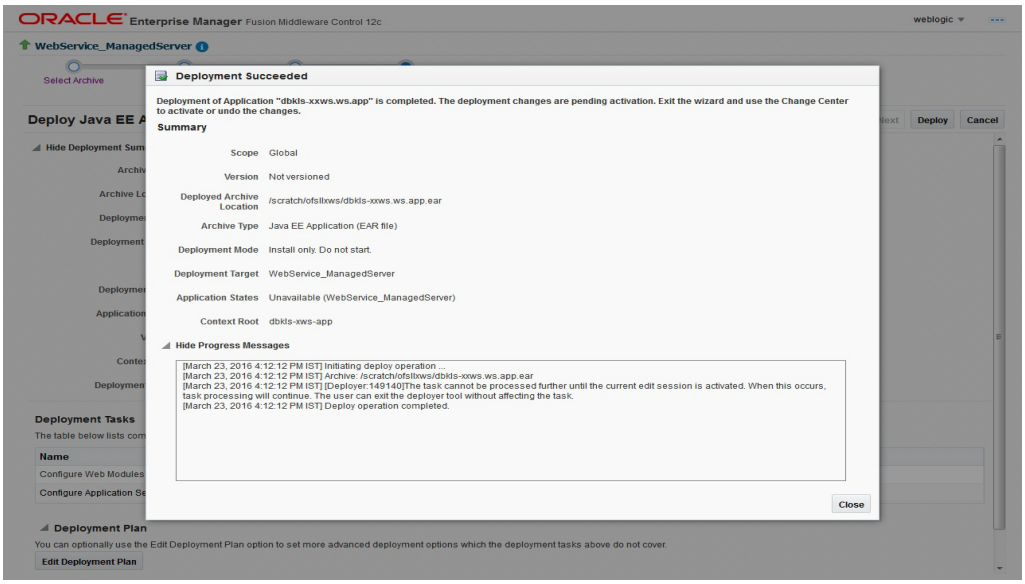
14. The following window is displayed. Check the context root and click ‘Next’.



15. The following window is displayed. Click ‘Deploy’.



16. The following window is displayed. Click ‘Close’.



17. If required, similarly deploy rest of the WebServices.

18. In case the context root has to be changed for the WebServices, it can be changed by editing the application.xml or through the console. To change through console, login to WebLogic Server 12c console (http://hostname:port/console).

19. Go to Deployments > <select the service deployment> > Configuration General and modify the context root to the recommended name as follows:

Service	Service EAR	Recommended Context	Remark
Webservices	dbkls-xxws.ws.app.ear	dbkls-xws-app	If this the recommended context name is not used, then the Service URLs in the configuration files needs to be changed
Edocs Interface	dbkls-xxws.ws.app-ds.ear	dbkls-xws-app-ds	If this the recommended context name is not used, then the Service URLs in the configuration files needs to be changed
Dealer Track Interface	dbkls-xxws.ws.app-ds.ear	dbkls-xws-app-dt	If this the recommended context name is not used, then the Service URLs in the configuration files needs to be changed
Route One Interface	dbkls-xxws.ws.app-ro.ear	dbkls-xws-app-ro	If this the recommended context name is not used, then the Service URLs in the configuration files needs to be changed

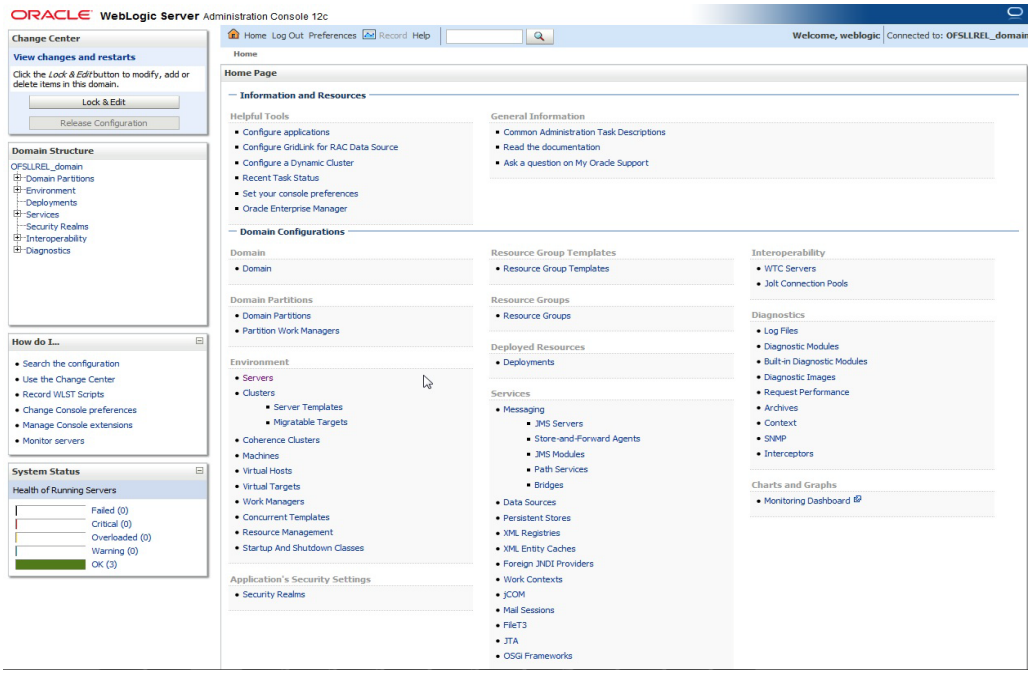
20. Ensure that the application status is ‘Active’.

4. Configuring Weblogic Policy on WebServices

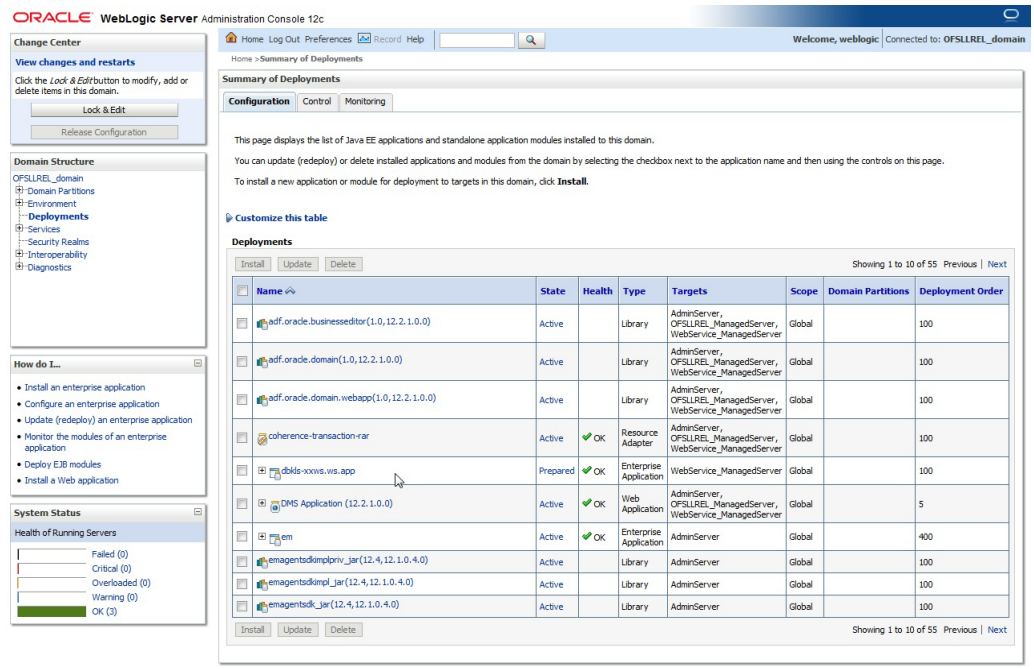
After deploying WebServices, you must configure Weblogic Policy on WebServices.

To configure Weblogic Policy on WebServices

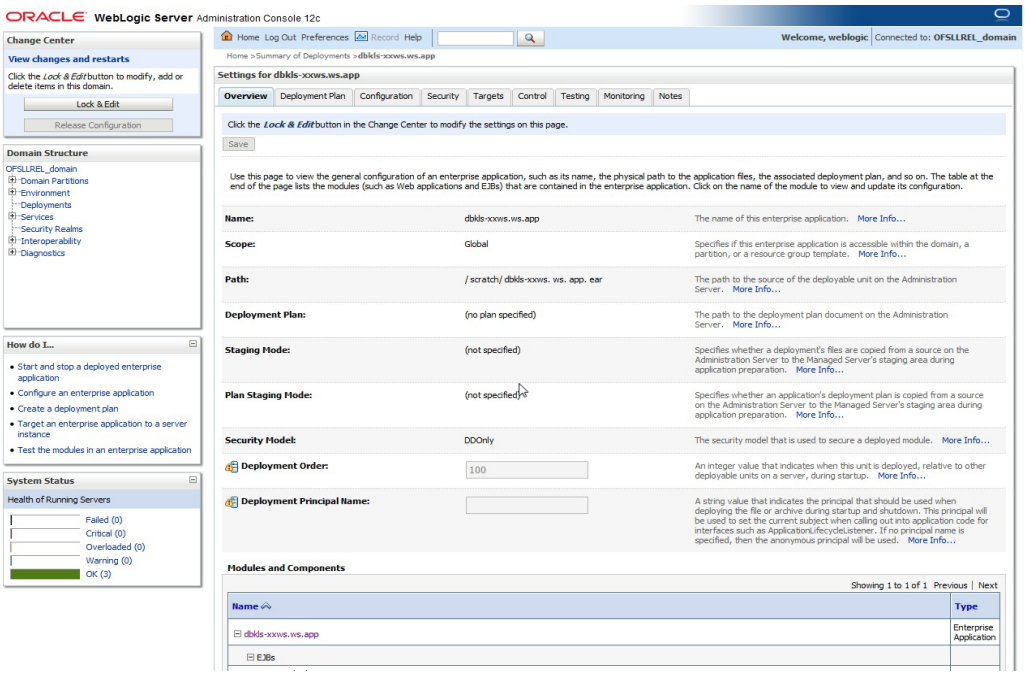
1. Login to WebLogic application server console (http://hostname:port/console). The following window is displayed.



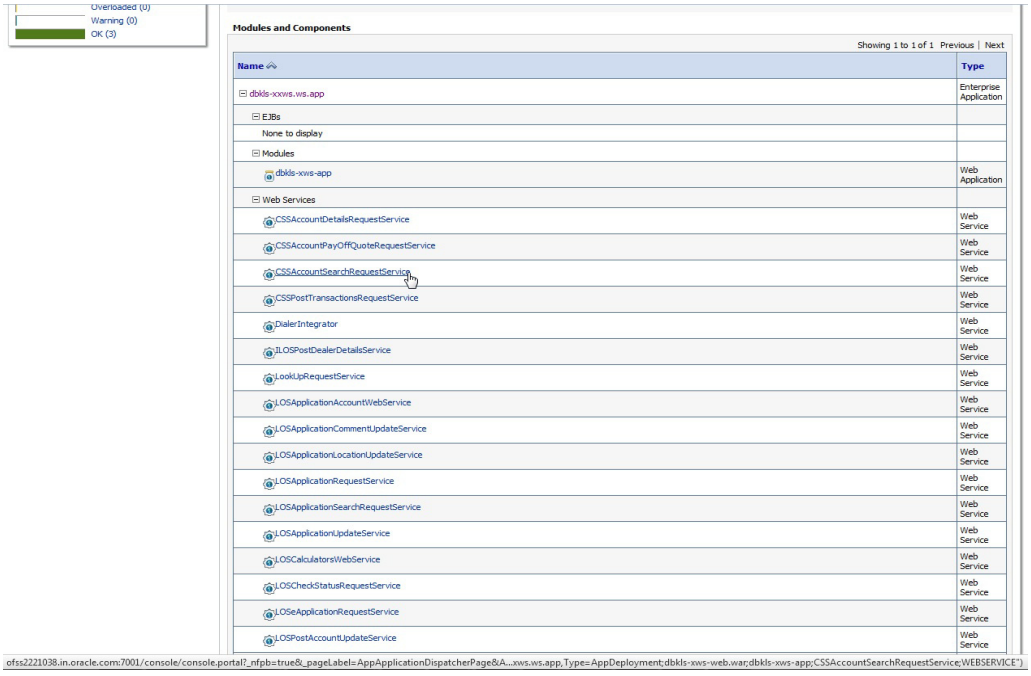
2. Click ‘Deployments’ which is available on both side panels as marked above. The following window is displayed.



3. Click on application name dbkls-xws-app on right side panel. The following window is displayed.

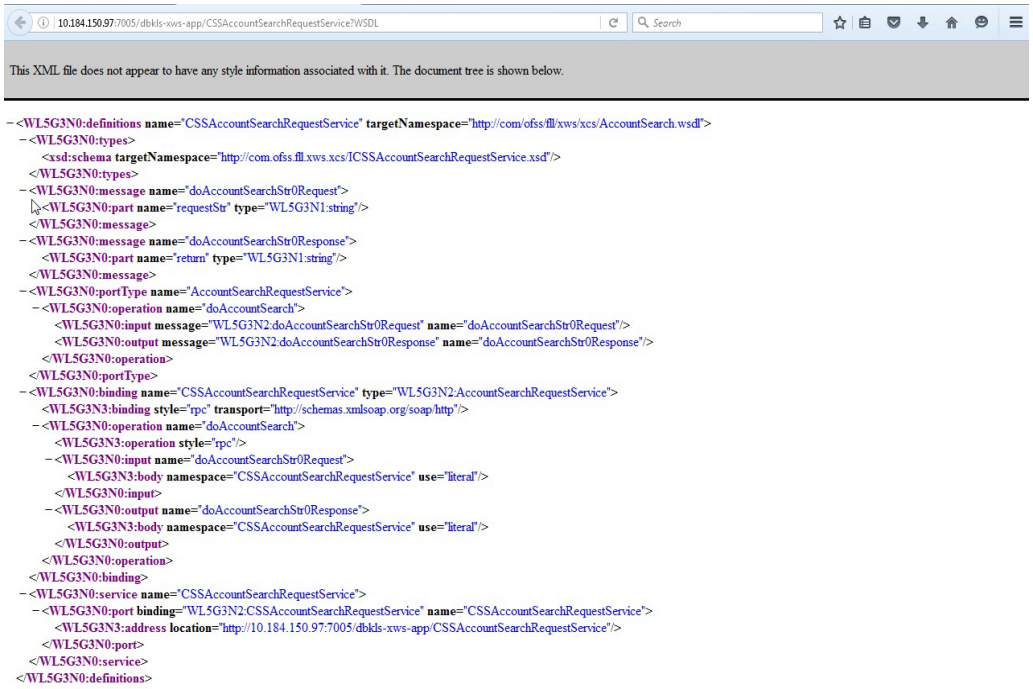


4. Scroll down the page.

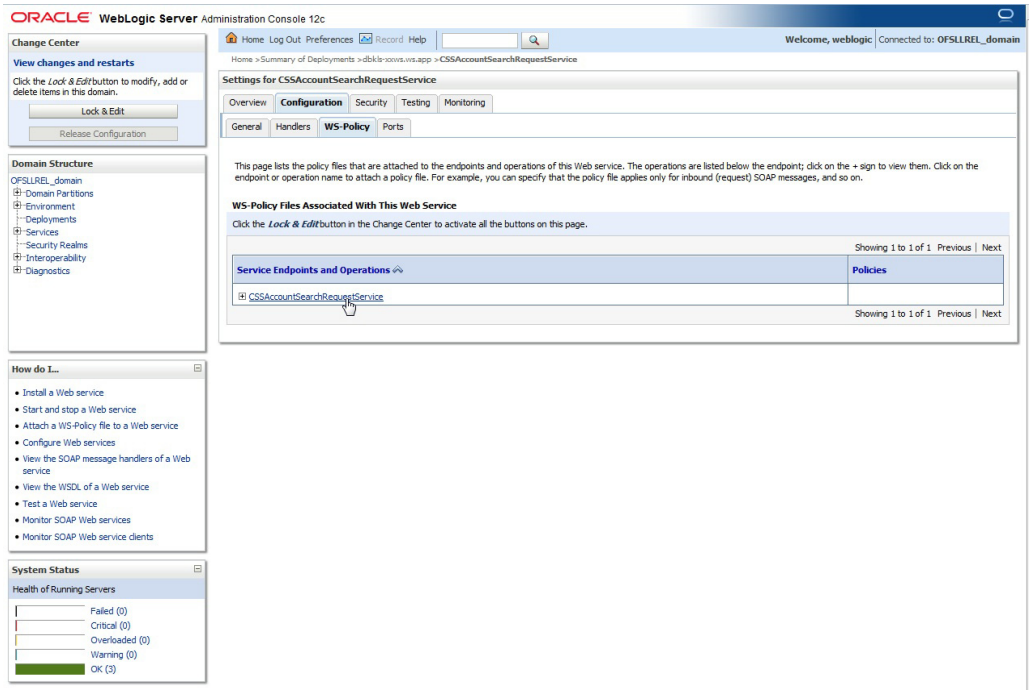


5. Click ‘WebServices CSSAccountSearchRequestService’.

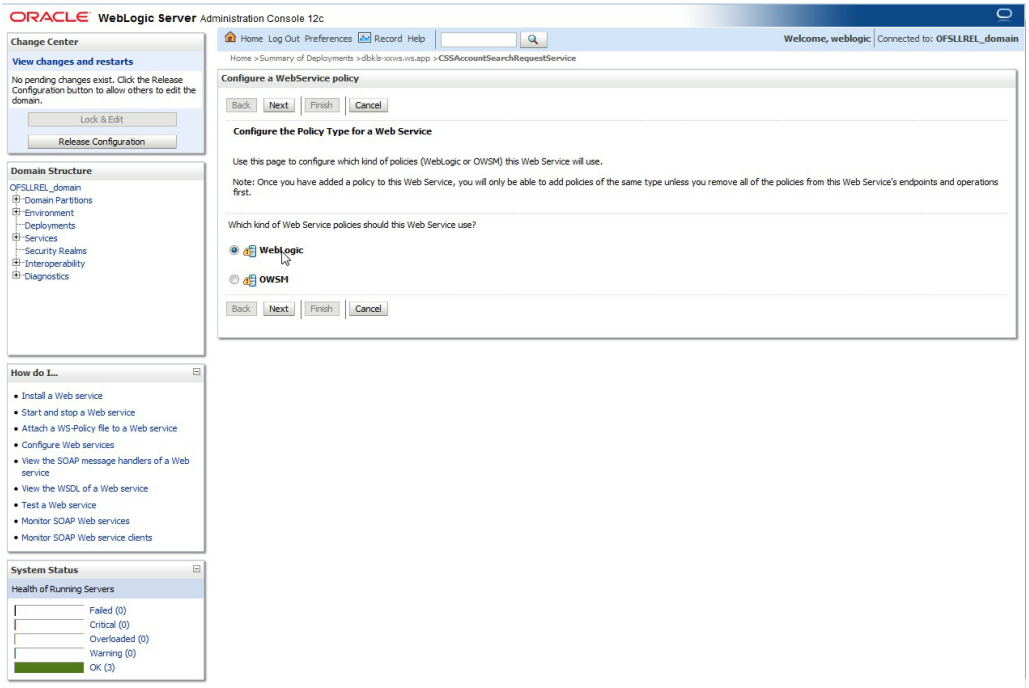
6. The WSDL will be accessible on http before applying WS-Policy.



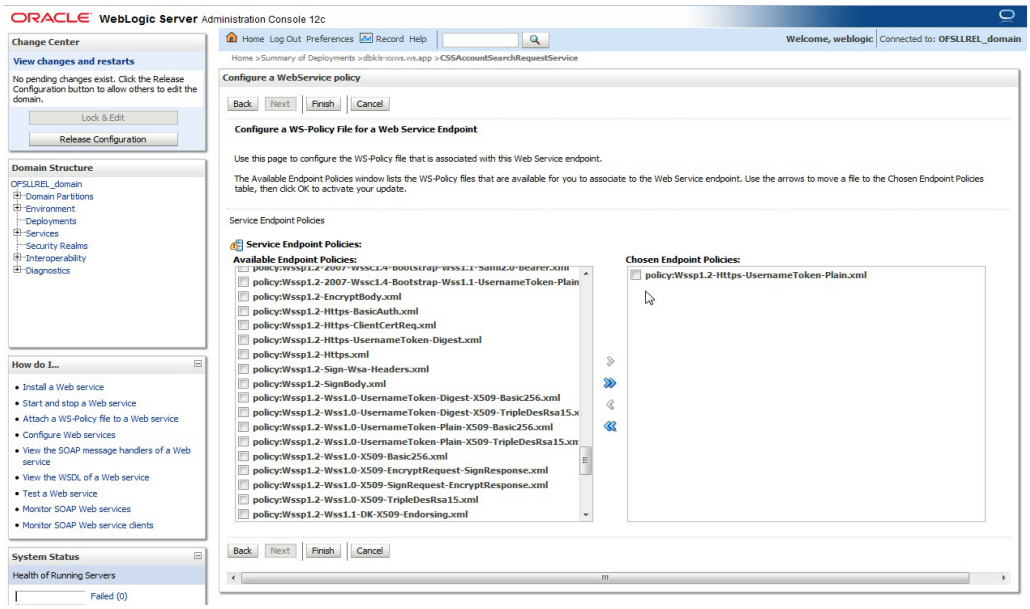
7. Navigate to Configuration > WS-Policy tab.



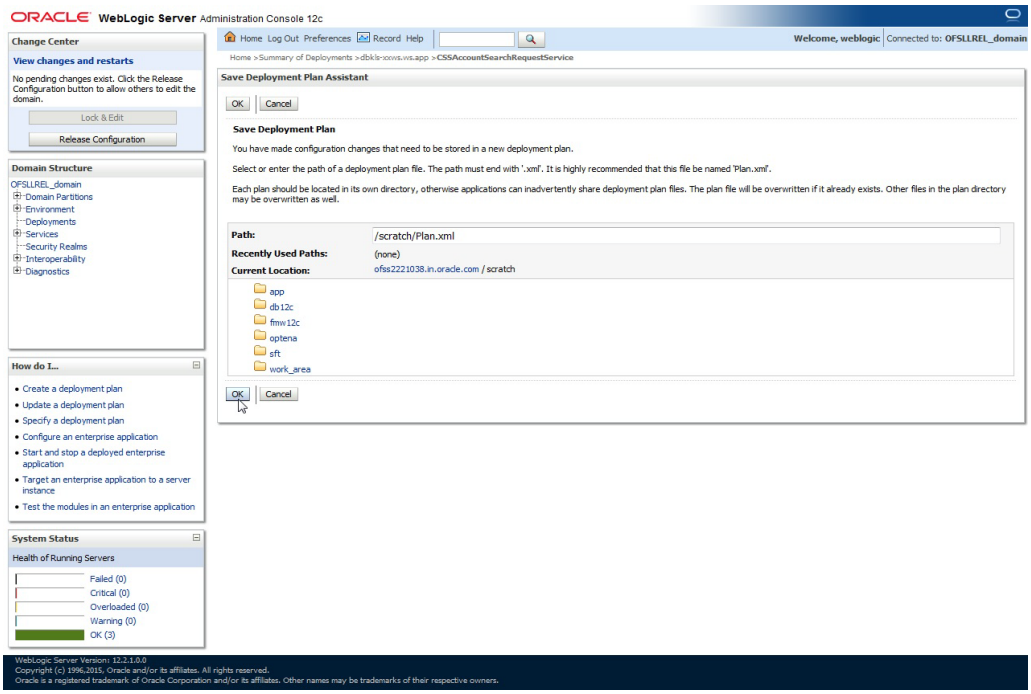
8. Click ‘CSSAccountSearchRequestService’.



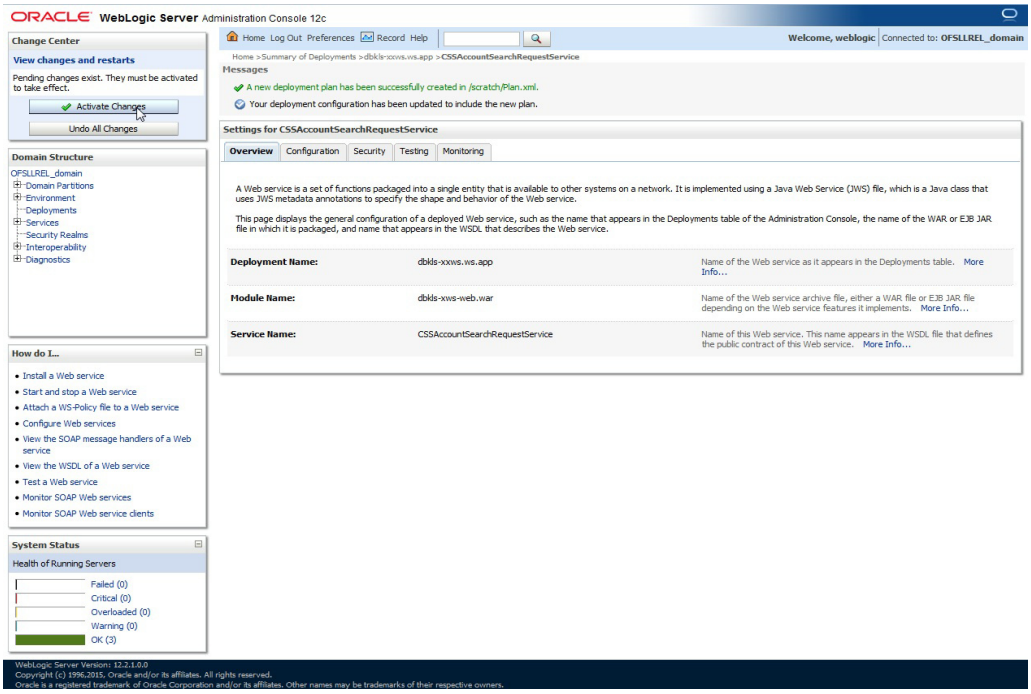
9. Select ‘policy:Wssp1.2-Https-UsernameToken-Plain.xml’ and click right arrow to move it to the selected policies list.



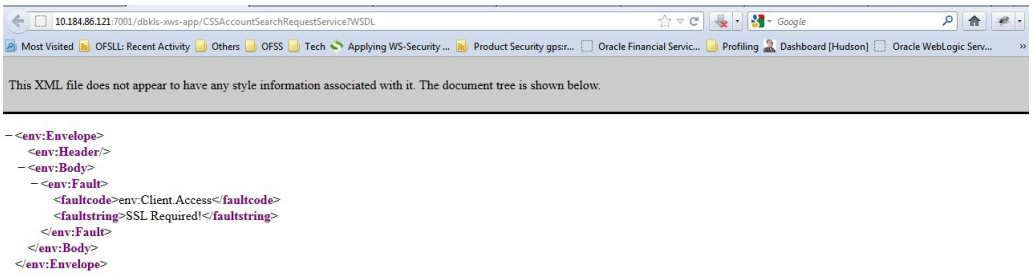
10. Click ‘OK’.



11. Browse to any folder to save Deployment Plan and click ‘OK’.



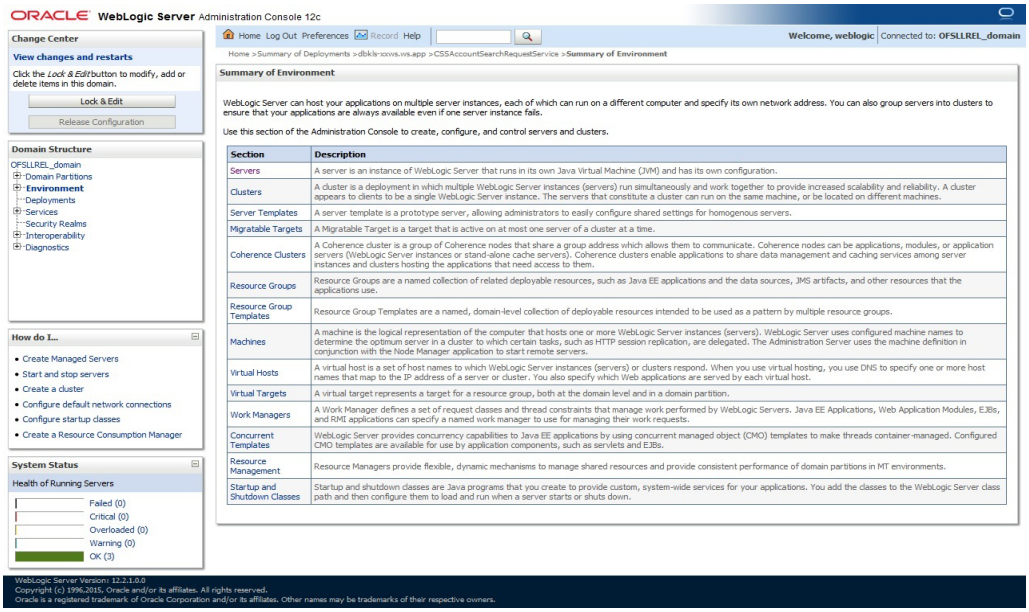
12. Now you will be not able to access the WSDL on http port. Following message will be shown when you access with http URL.



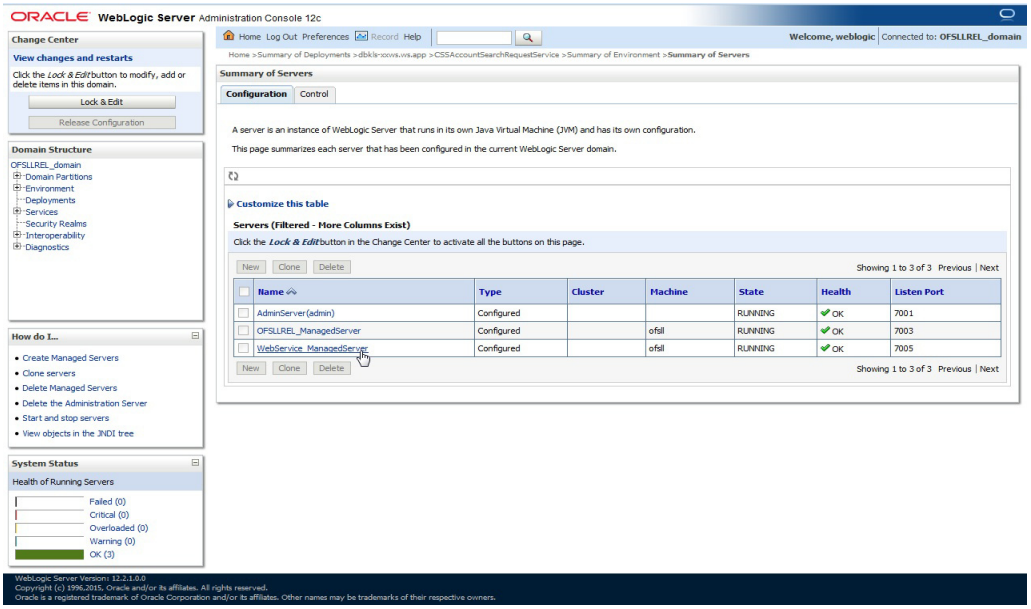
Note

For SSL communication, the vendor servers seek public certificates. Hence, you need to download the certificates from vendor website and import into your java keystore. You then need to configure Weblogic to present the certificates to vendor servers for successful handshake.

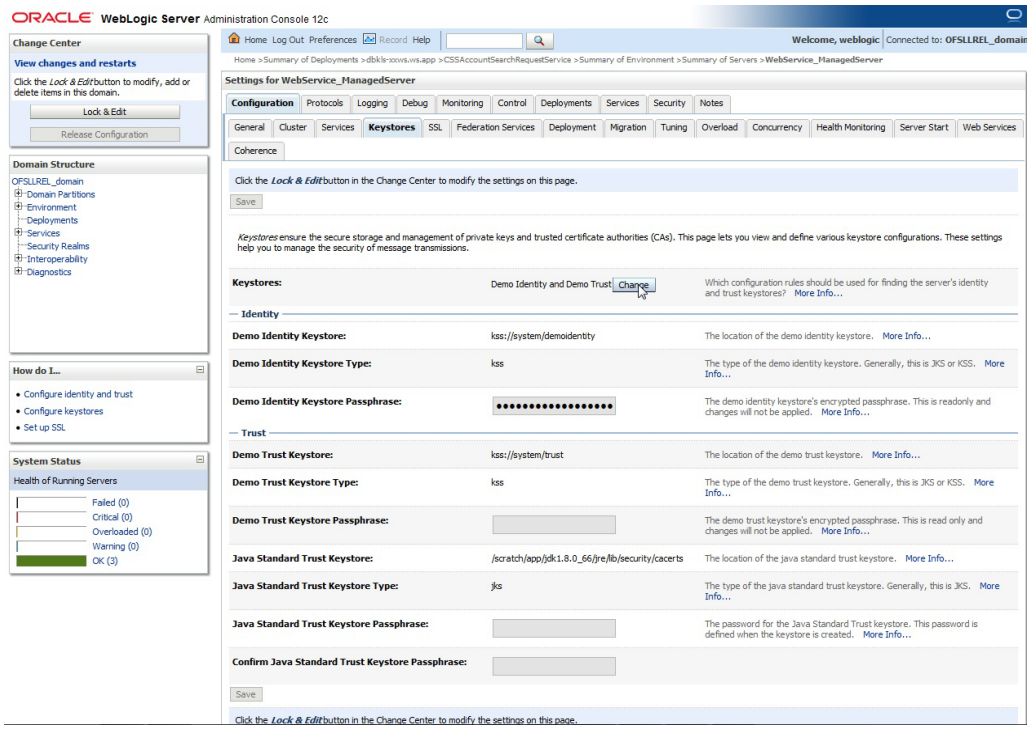
13. Select the environment on the left pane. The following window is displayed.



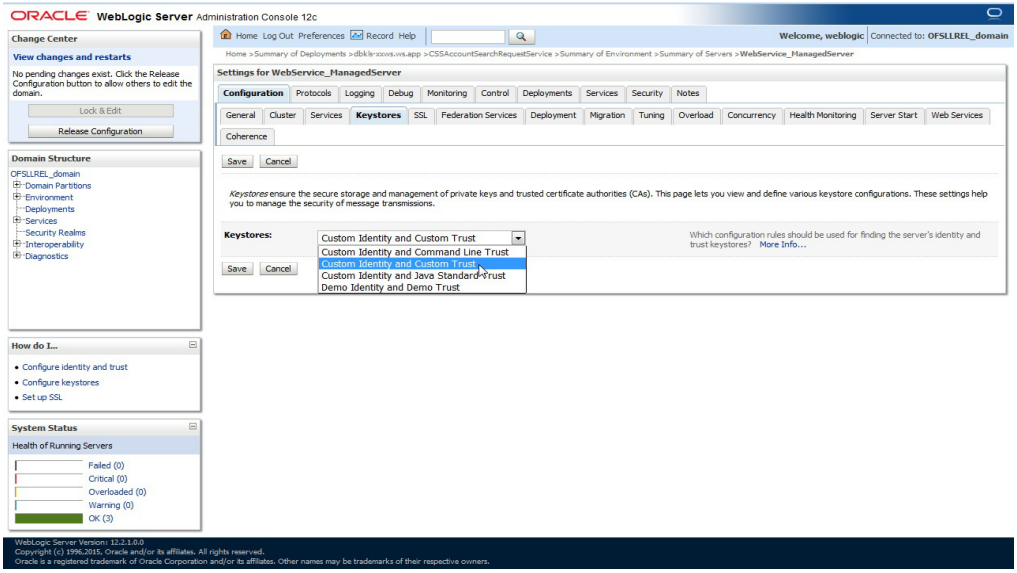
14. Click ‘Servers’.



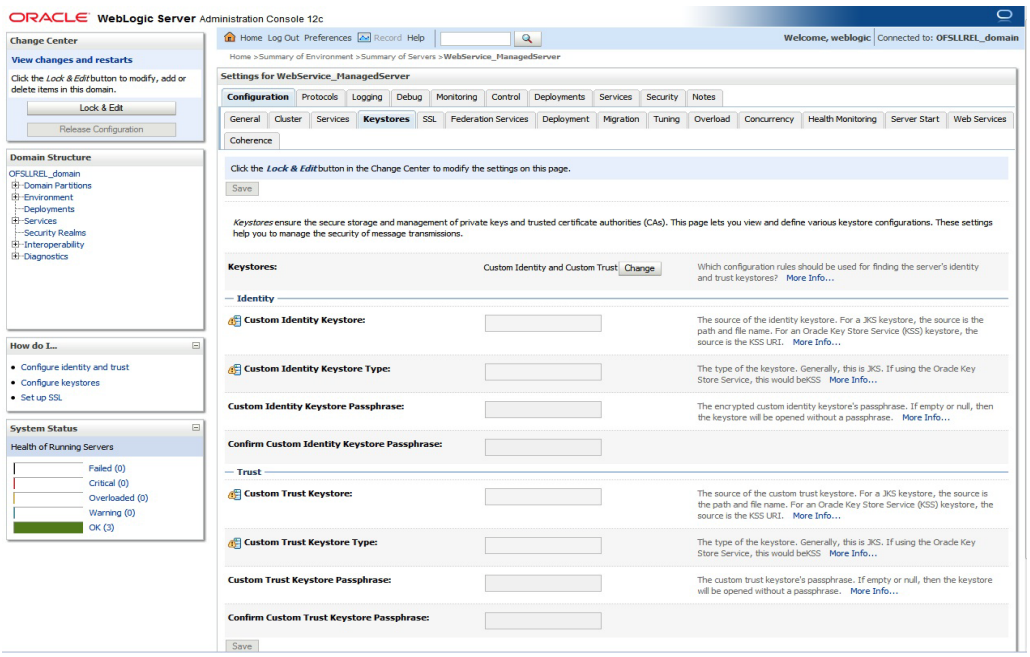
15. Select the Server into which the WebServices are deployed.



16. Select ‘Keystores’.



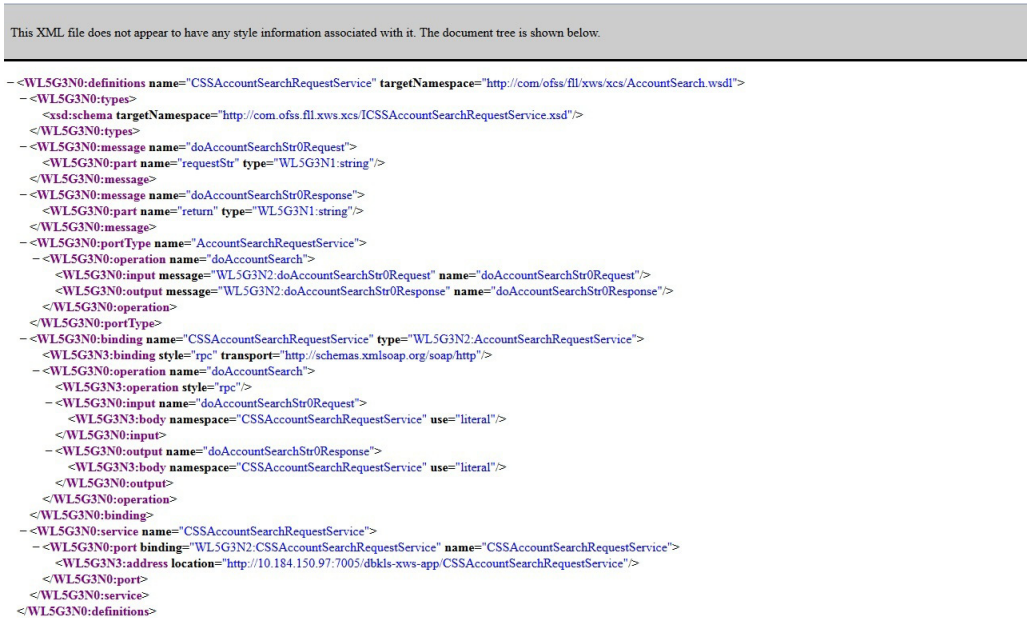
17. Click ‘Change’ and select ‘Custom Identity and Custom trust’ from the drop-down list.



Specify

- Custom Identity Keystore: **Java keystore holding the certificates**
- Custom Trust Keystore: **Java keystore holding the certificates**
- Custom Identity Keystore Type: **jks**
- Custom Trust Keystore Type: **jks**
- %Keystore Passpharses: keystore password

18. Click **Save**. The WSDL can be accessed on https port as below. The WS-Policy will be shown in WSDL.



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
- <WL5G3N0:definitions name="CSSAccountSearchRequestService" targetNamespace="http://com/ofss/fll/xws/xcs/AccountSearch.wsdl">
  - <WL5G3N0:types>
      <xsd:schema targetNamespace="http://com.ofss.fll.xws.xcs/ICSSAccountSearchRequestService.xsd"/>
    </WL5G3N0:types>
  - <WL5G3N0:message name="doAccountSearchStr0Request">
      <WL5G3N0:part name="requestStr" type="WL5G3N1:string"/>
    </WL5G3N0:message>
  - <WL5G3N0:message name="doAccountSearchStr0Response">
      <WL5G3N0:part name="return" type="WL5G3N1:string"/>
    </WL5G3N0:message>
  - <WL5G3N0:portType name="AccountSearchRequestService">
    - <WL5G3N0:operation name="doAccountSearch">
        <WL5G3N0:input message="WL5G3N2:doAccountSearchStr0Request" name="doAccountSearchStr0Request"/>
        <WL5G3N0:output message="WL5G3N2:doAccountSearchStr0Response" name="doAccountSearchStr0Response"/>
      </WL5G3N0:operation>
    </WL5G3N0:portType>
  - <WL5G3N0:binding name="CSSAccountSearchRequestService" type="WL5G3N2:AccountSearchRequestService">
      <WL5G3N3:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
    - <WL5G3N0:operation name="doAccountSearch">
        <WL5G3N3:operation style="rpc"/>
      - <WL5G3N0:input name="doAccountSearchStr0Request">
          <WL5G3N3:body namespace="CSSAccountSearchRequestService" use="literal"/>
        </WL5G3N0:input>
      - <WL5G3N0:output name="doAccountSearchStr0Response">
          <WL5G3N3:body namespace="CSSAccountSearchRequestService" use="literal"/>
        </WL5G3N0:output>
      </WL5G3N0:operation>
    </WL5G3N0:binding>
  - <WL5G3N0:service name="CSSAccountSearchRequestService">
    - <WL5G3N0:port binding="WL5G3N2:CSSAccountSearchRequestService" name="CSSAccountSearchRequestService">
        <WL5G3N3:address location="http://10.184.150.97:7005/dbkls-xws-app/CSSAccountSearchRequestService"/>
      </WL5G3N0:port>
    </WL5G3N0:service>
</WL5G3N0:definitions>
```

Apply WS-Policy policy:Wssp1.2-Https-UsernameToken-Plain.xml to the following WebServices.

- CSSAccountDetailsRequestService
- CSSAccountPayOffQuoteRequestService
- CSSAccountSearchRequestService
- CSSPostTransactionsRequestService

5. Verifying Successful Installation

5.1 Steps to Verify Successful Webservices deployment:

1. Making sure that the state is ACTIVE and health in OK in the Weblogic
2. Accessing the WSDL using: http://<hostname>:<port>/<context root>/ CSSAccountDetailsRequestService?WSDL and receiving error: “No valid XML found”

5.2 Steps to Verify Successful Interface (Route One) deployment:

1. Making sure that the state is ACTIVE and health in OK in the Weblogic
2. Accessing the interface URLs mentioned below from the browser and receiving error: "No valid XML found":
   - "http://<hostname>:<port>/<context root>/postdt2xws?xaeprc
   - "http://<hostname>:<port>/<context root>/postds2xws?xaeprc
   - "http://<hostname>:<port>/<context root>/postro2xws?xaeprc

6. Enable Logging

6.1 Enabling WebService Log

It is not recommended to enable WebServices log while running in production environment. The logging should be enabled in production only for debugging purpose by following the below mentioned steps.

1. The log file path has to be specified in 'logging.properties' file.
2. The file is available inside config folder. Navigate to dbkls-xxws.ws.app/dbkls-xws-web/WEB-INF/classes/config
   - Edit the following with the log file path: java.util.logging.FileHandler.pattern=<Path>/dbkls_xws_%g.log
3. Restart the managed server (in which services are deployed) with the following argument:

Type	Description
Argument	Djava.util.logging.config.file=<Path of the above logging.properties file>/logging.properties
Example	nohup ./startManagedWebLogic.sh -Djava.util.logging.config.file=<Path of the above logging.properties file>/logging.properties &

4. The login levels can be set through the logging.properties file. Un-comment the level required in the logging.properties file
   - #com.ofss.fll.xws.level=FINEST
   - #com.ofss.fll.xws.level=FINER
   - #com.ofss.fll.xws.level=FINE
   - #com.ofss.fll.xws.level=CONFIG
   - #com.ofss.fll.xws.level=INFO

6.2 Enabling SQL Log for WebServices Component

It is not recommended to enable WebServices SQL log while running in production environment. The logging should be enabled in production only for debugging purpose by following the below mentioned steps:

1. The CMN_DEBUG_LEVEL system parameter should be enabled and set to appropriate non-zero value.
2. The CMN_DEBUG_METHOD system parameter should be enabled and set to appropriate non-zero value.

3. The respective package debug level for the webservice should be enabled and set to appropriate non-zero value in system parameter. Please refer the below table to for the service and debug level mapping.

Service	**Functionality**		
DialerIntegrator: xcsadi_em_100_01	To receive loan application from dealer track: XAEPRC_EM_100_01	To receive edocs application update : XAE-EDS_EM_100_01	To receive loan application from route one:XAE-PRC_EM_100_01
CSSAccountDetailsRequestService: XCSPRC_EM_100_01	To receive loan application/deal update from dealer track: XAEUP-D_EM_100_01	To receive edocs comment update : XAE-EDS_EM_100_01	To receive loan application/deal refresh from route one : XAEUP-D_EM_100_01
CSSAccountSearchRequestService: XCSACS_EM_100_01	To receive comments from dealer track : XAEPRC_EM_100_01	To receive edocs location update : XAE-EDS_EM_100_01	To receive comments from route one: XEAUP-D_EM_100_01
CSSPostTransactionsRequestService: XCSUP-D_EM_100_01	To post comments to dealer track: XAECH-K_EM_100_01		To post comments to route one: XAECH-K_EM_100_01
CSSAccountPayOffQuoteRequestService: N/A	To post application status to dealer track: XAECHK_EM_100_01		To post application status to route one: XAECH-K_EM_100_01
LOSeApplicationRequestService: XAE-EDS_EM_100_01	To post dealer details to dealer track: XPR-PRC_EM_100_01		To post dealer details to dealer track: XPR-PRC_EM_100_01
LOSApplicationRequestService: XAE-PRC_EM_100_01			
LOSApplicationSearchRequestService: XAE-QUE_EM_100_01			
LOSApplicationLocationUpdateService: XAE-EDS_EM_100_01			

Service	Functionality		
LOSApplication-CommentUpdate-Service: XAE-EDS_EM_100_01			
LOSPostStatusRe-questService: XAECH-K_EM_100_01			
LOSApplica-tionUpdateSer-vice: XAEUP-D_EM_100_01			
LOSCheckStatus-RequestService: XAECH-K_EM_100_01			
LOSEcontractSer-vice: XACPRC_EM_100_01			

7. Configure AQ-JMS Bridge

The following steps are to be performed to configure the AQ-JMS Bridge through the Weblogic Console:

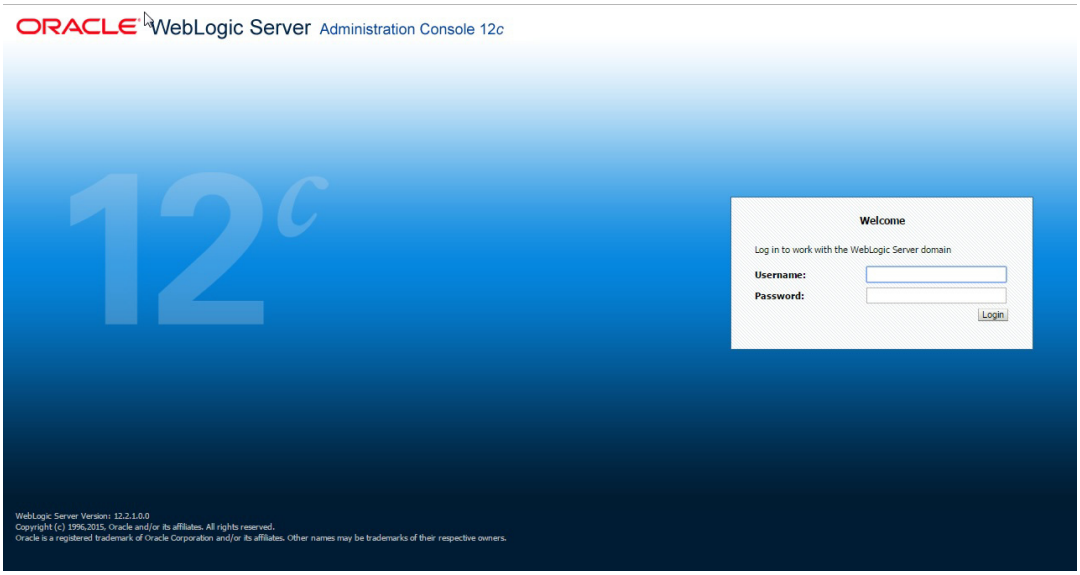
- Create Data Sources for AQ-JMS Bridge
- Configure MDB Flow
- Create Credentials and System Policies
- Deploy MDB EJB

Note

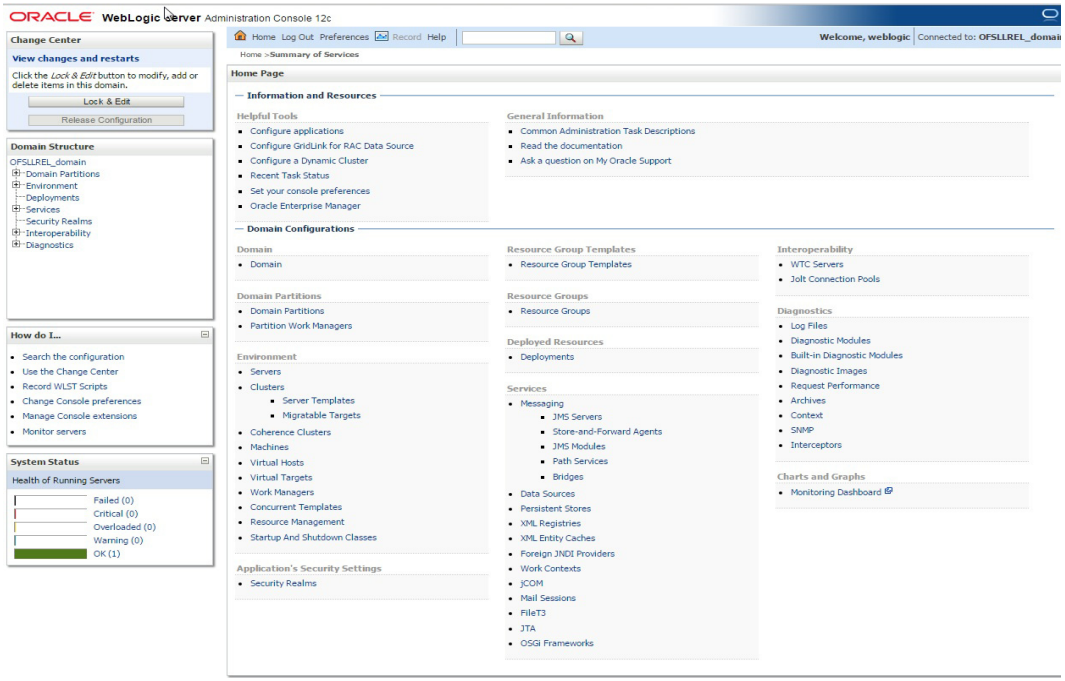
Ensure that MDB EJB is not configured and deployed (i.e. OfsllAppQueue.ear deployment) on the same server on which the other WebServices are deployed.

7.1 Create Data Sources for AQ-JMS Bridge

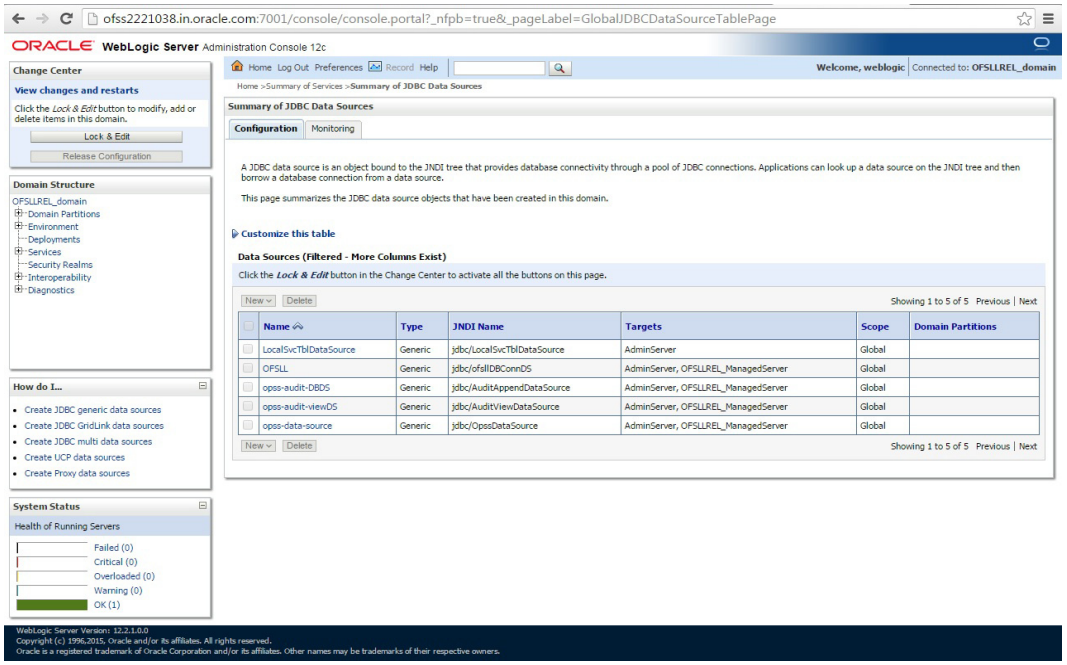
1. Login to Oracle Weblogic 12c console (http://hostname:port/console).



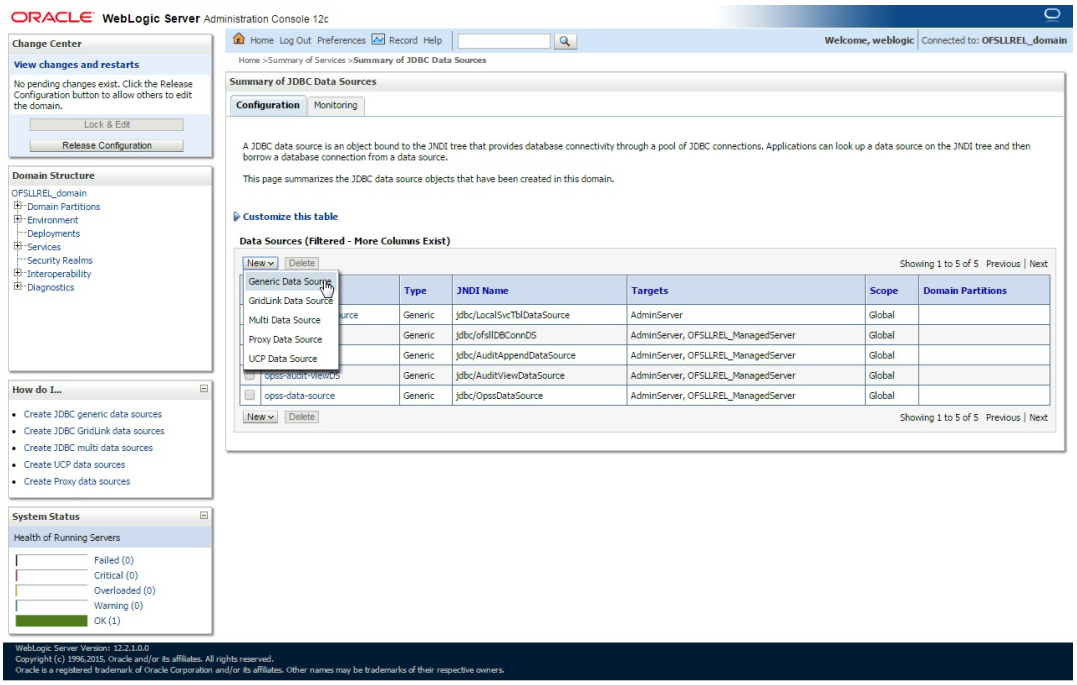
2. On successful login, the following window is displayed.



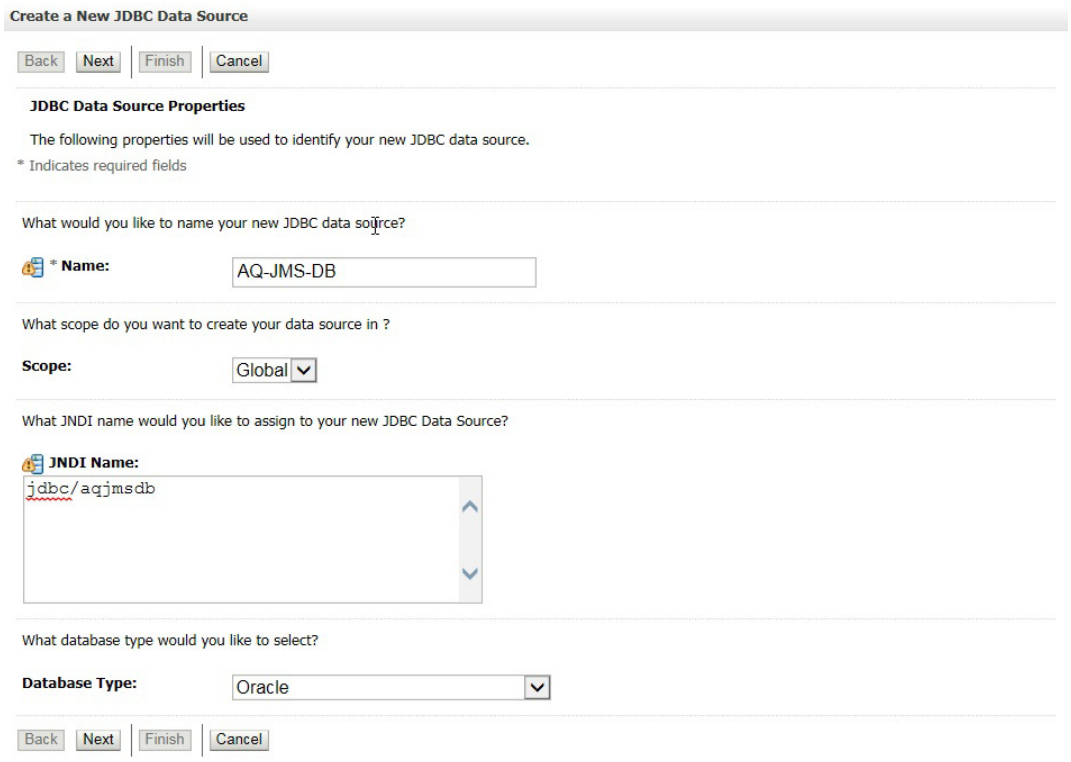
3. Click Domain Name > Services > Data Sources. The following window is displayed.



4. Click ‘Lock & Edit’ button on the left panel. Click ‘New’ on right panel and select ‘Generic Data Source’.



5. The following window is displayed.



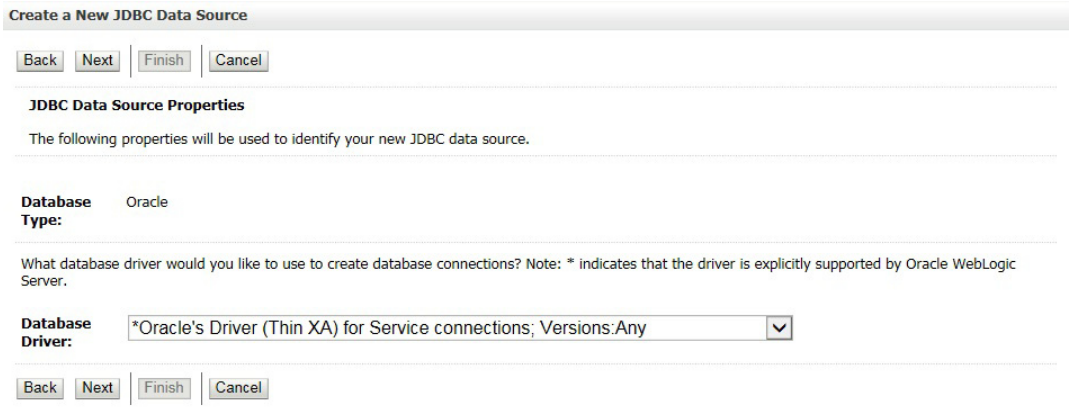
6. Specify the following details:
   - Enter Data source Name
   - Enter the JNDI Name as ‘jdbc/aqjmsdb’.

Note

If required, you can specify any other JNDI name, but however ensure to use the same JNDI name during other configuration steps.

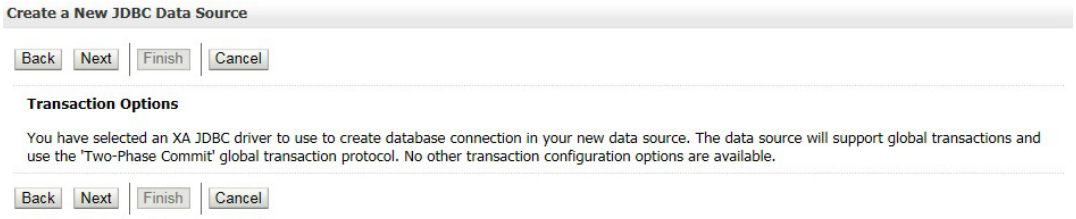
– Select ‘Oracle’ as Database Type.

7. Click ‘Next’. The following window is displayed.

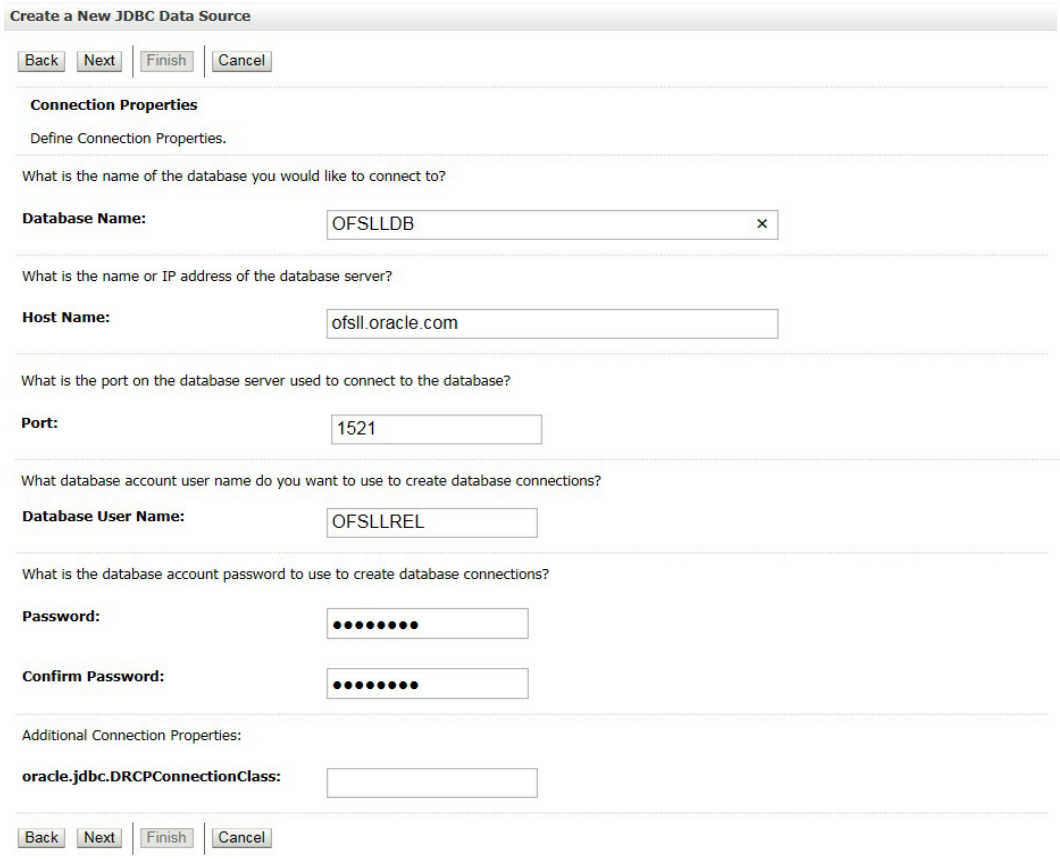


8. Select the Database Driver ‘Oracle's Driver(Thin XA) for Services connections;Versions:Any’.

9. Click ‘Next’. The following window is displayed.

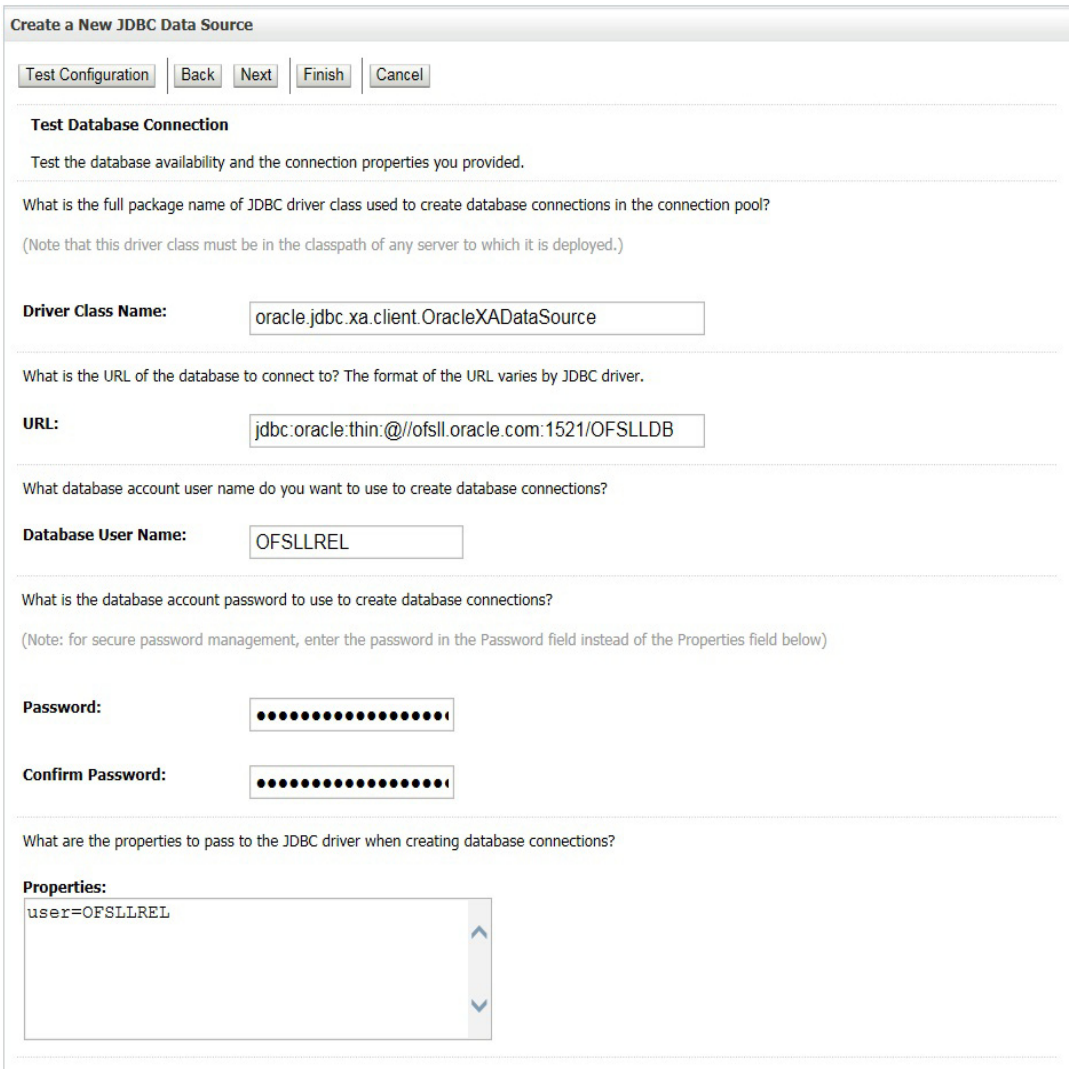


10. Click ‘Next’. The following window is displayed.

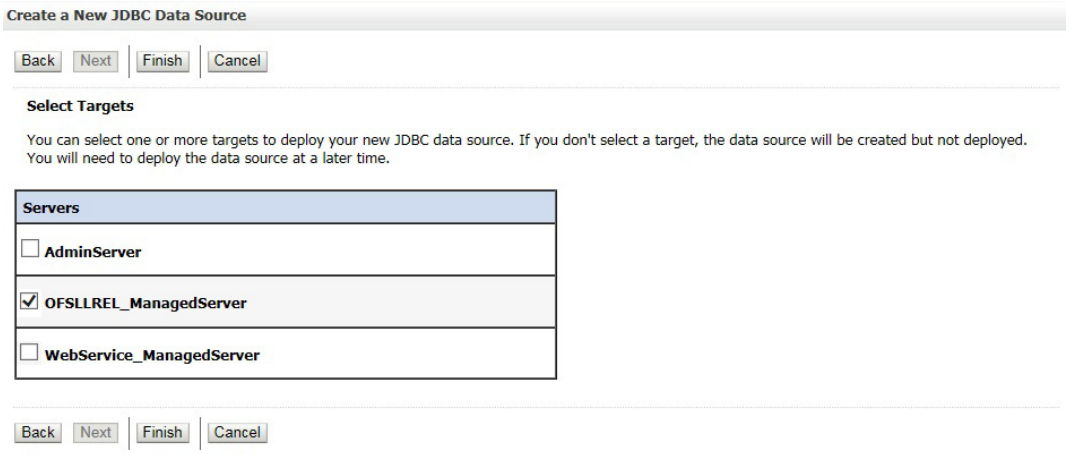


11. Enter the Database details.

12. Click ‘Next’. The following window is displayed.



13. Click ‘Test Configuration’. On completion, displays a confirmation message as ‘Connection test succeeded’.
14. Click ‘Next’. The following window is displayed.

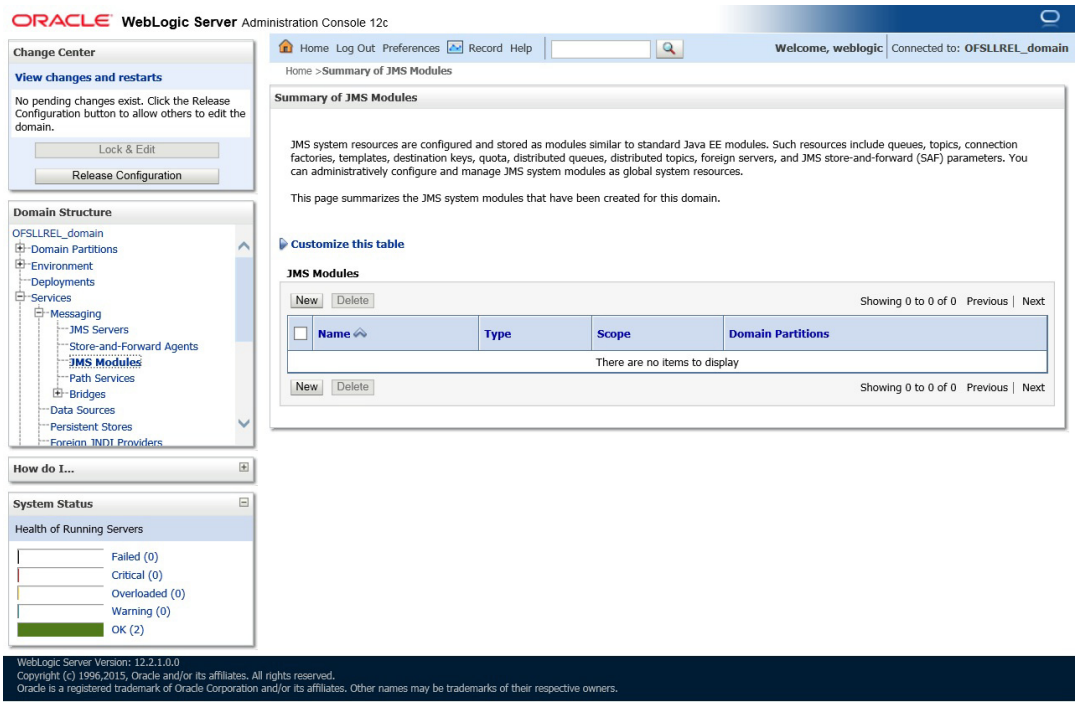


15. Select target Server as ‘OFSLLREL_ManagedServer’.
16. Click ‘Finish’ to activate the changes.

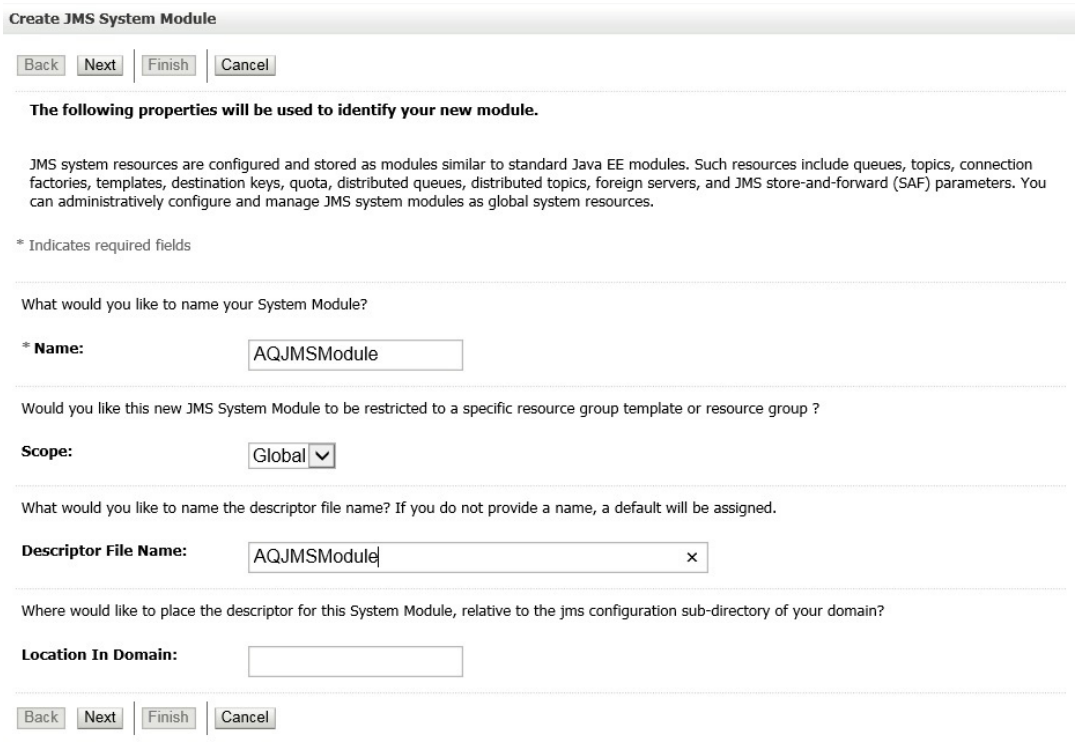
7.2 Configure MDB Flow

To configure the MDB Flow from Weblogic Console, do the following:

1. Login to Oracle Weblogic 12c console (http://hostname:port/console).
2. On the left pane, click 'Services'.
3. In Messaging tree click ‘JMS Modules’. The following window is displayed.



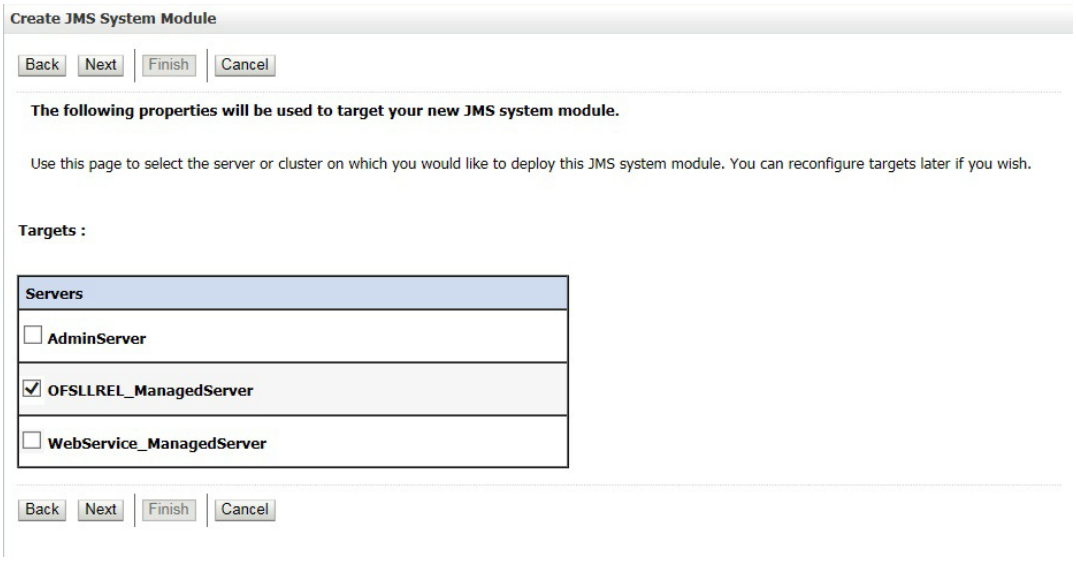
4. Click 'New'. The following screen is displayed.



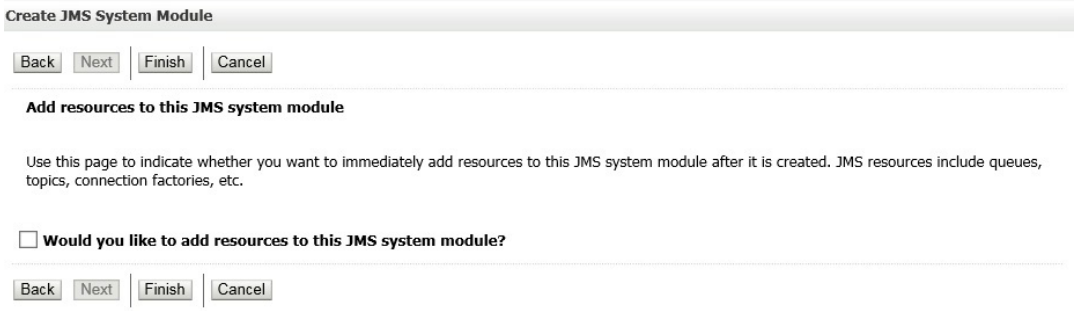
5. Specify the following details:

- Name: AQJMSModule
- Descriptor File Name: AQJMSModule

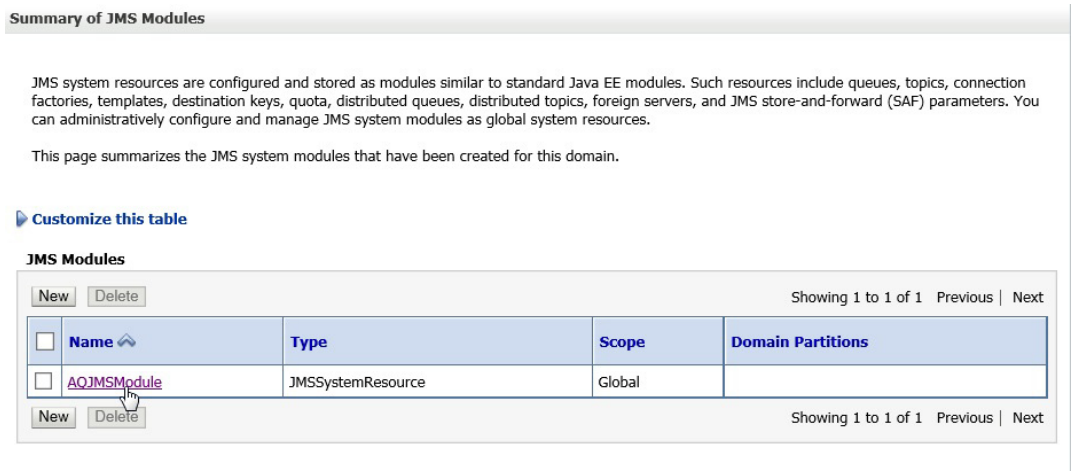
6. Click ‘Next’. The following window is displayed.



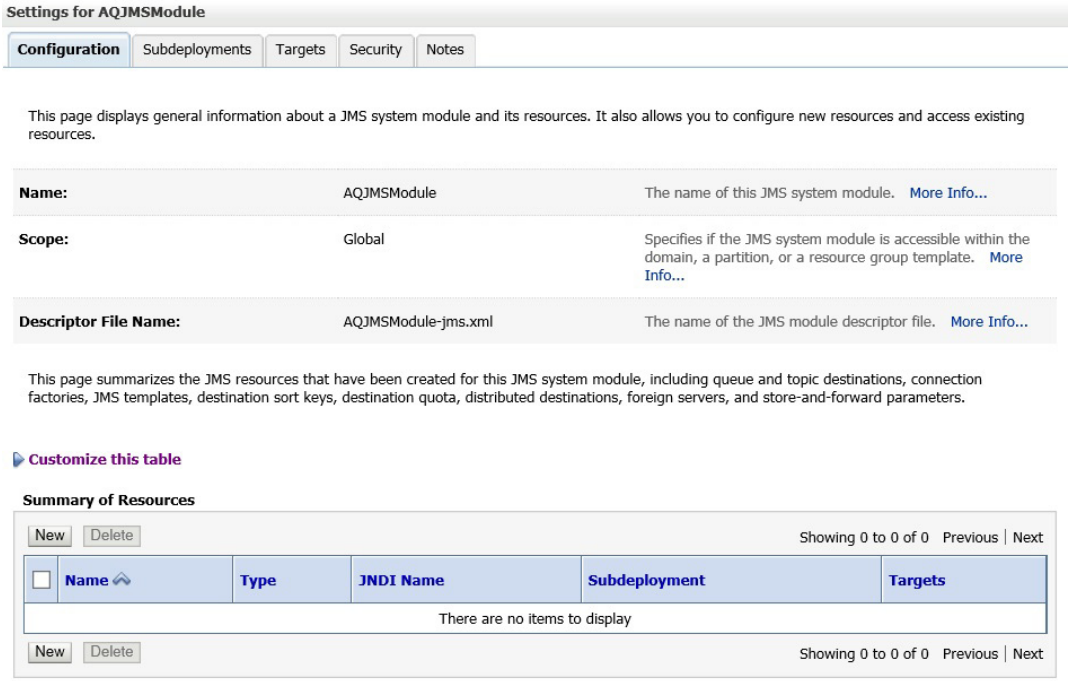
7. Select target servers as ‘OFSLLREL_ManagedServer’.
8. Click ‘Next’. The following window is displayed.



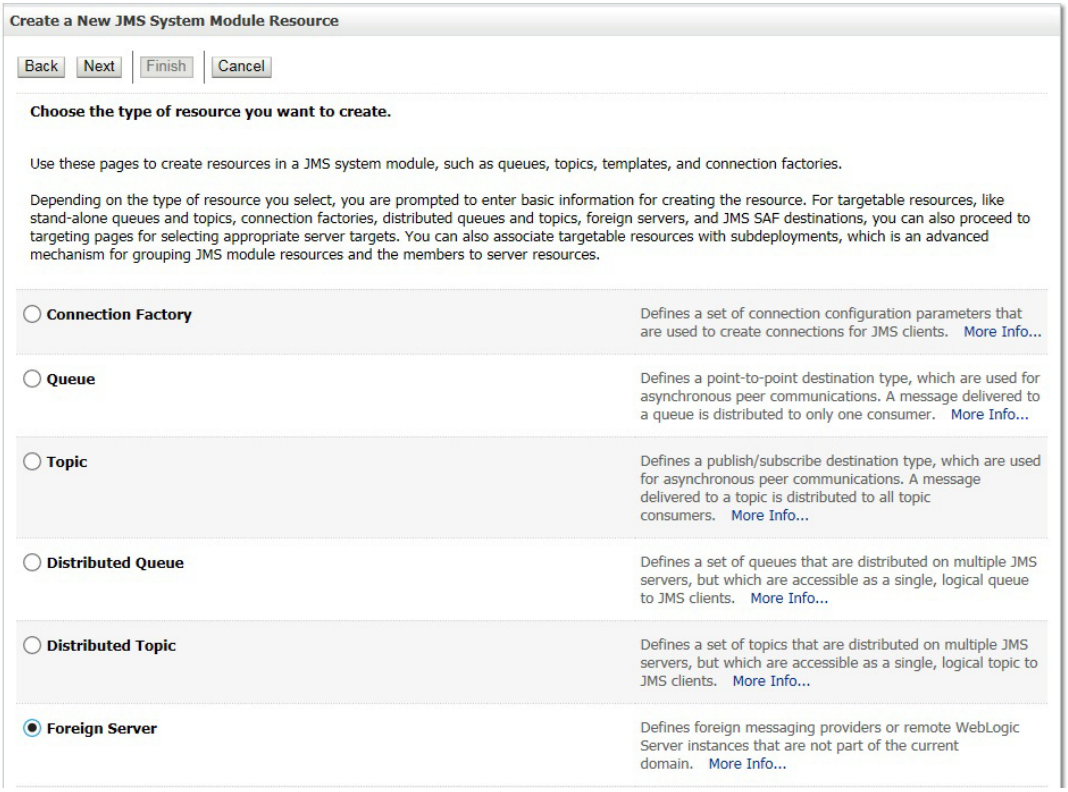
9. Click ‘Finish’ to activate the changes. The following window is displayed.



10. Click on the JMS Module that you created. The following window is displayed.

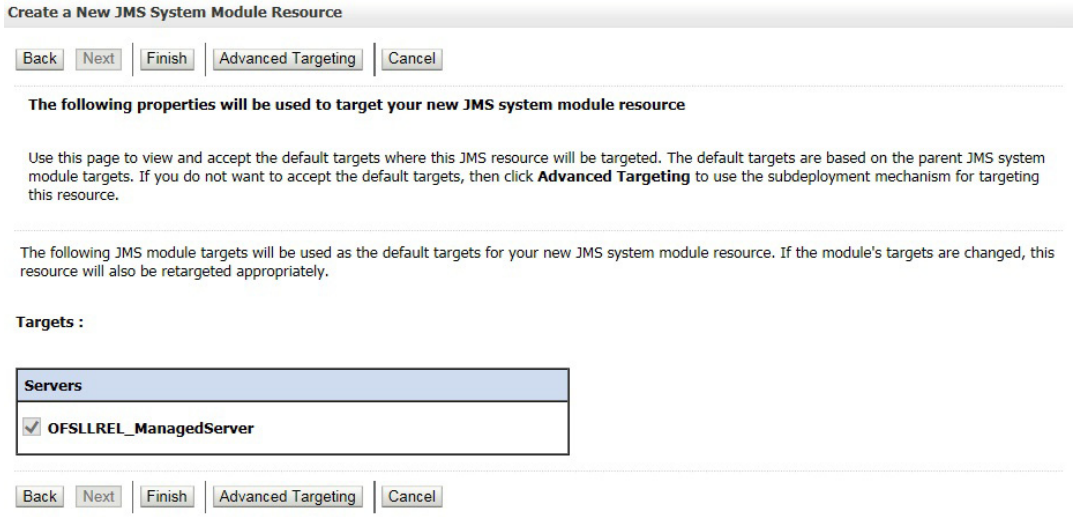


11. Click ‘New’ in ‘Summary of Resources’ section. The following window is displayed.

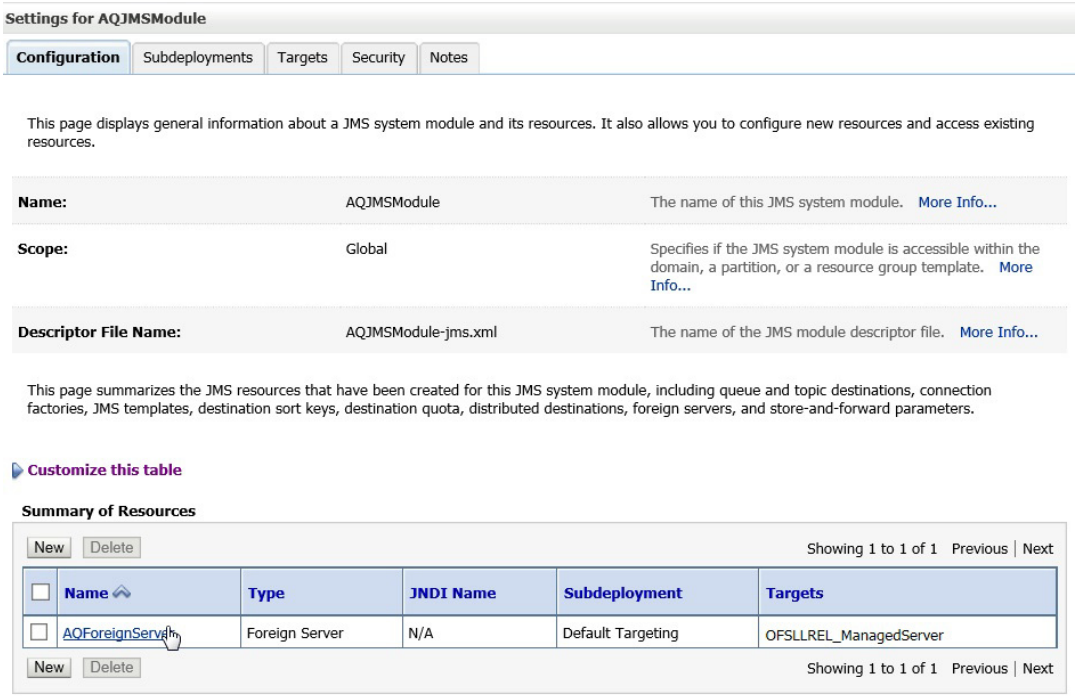


12. Select ‘Foreign Server’ as the option for type of resource to be created.

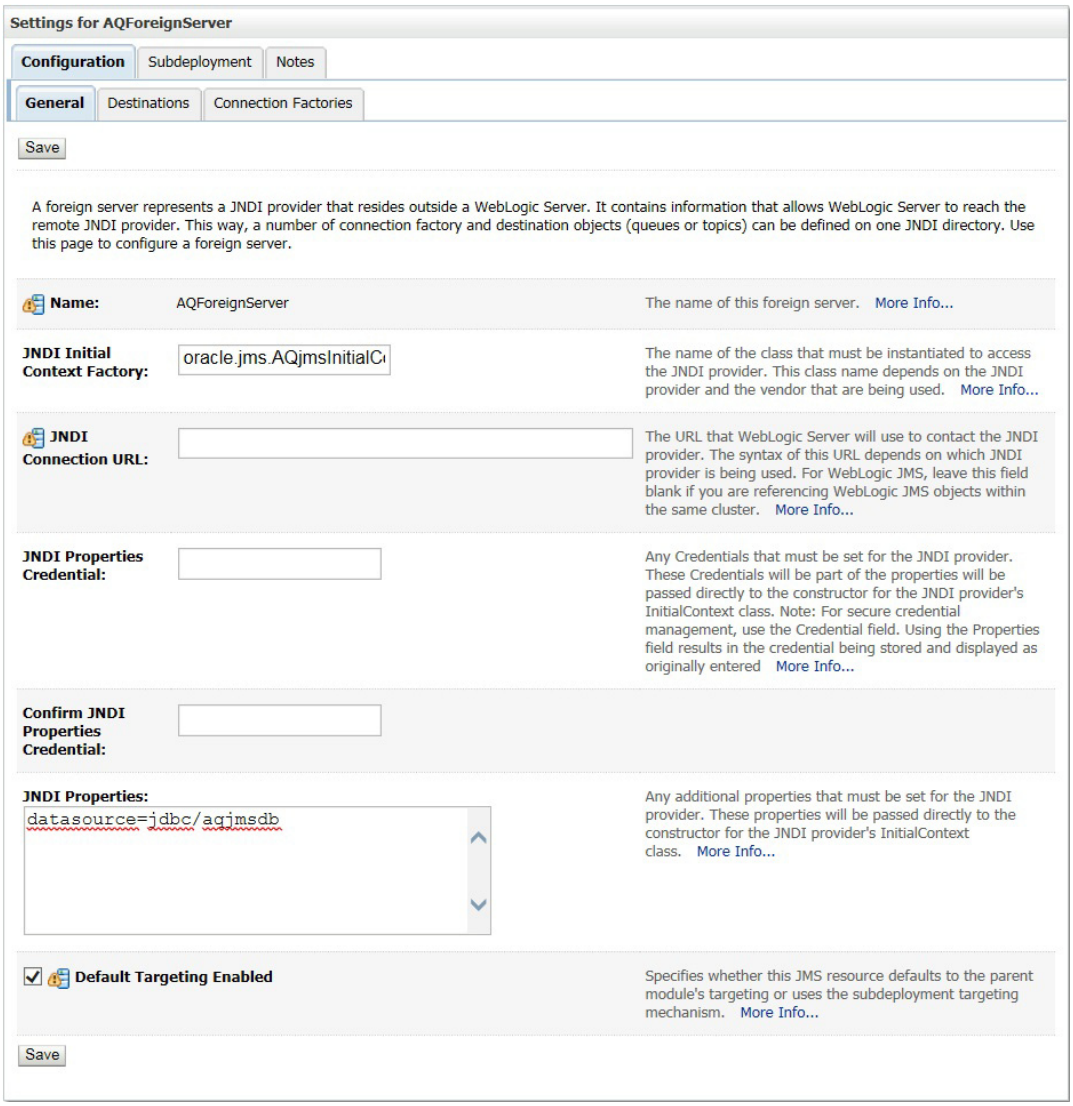
13. Click ‘Next’. The following window is displayed.



14. Click ‘Finish’ and activate the changes. The following window is displayed.

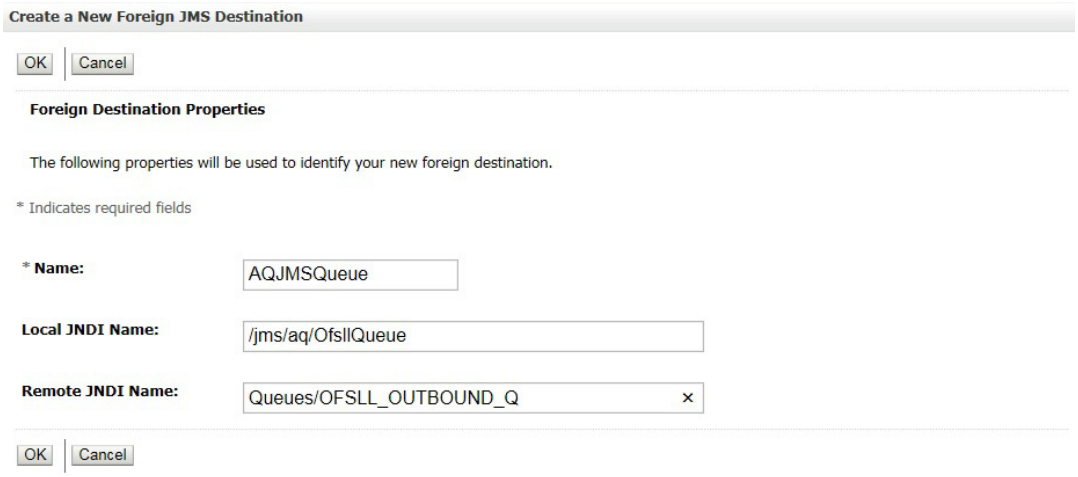


15. Click on the Foreign Server that you created. The following window is displayed.

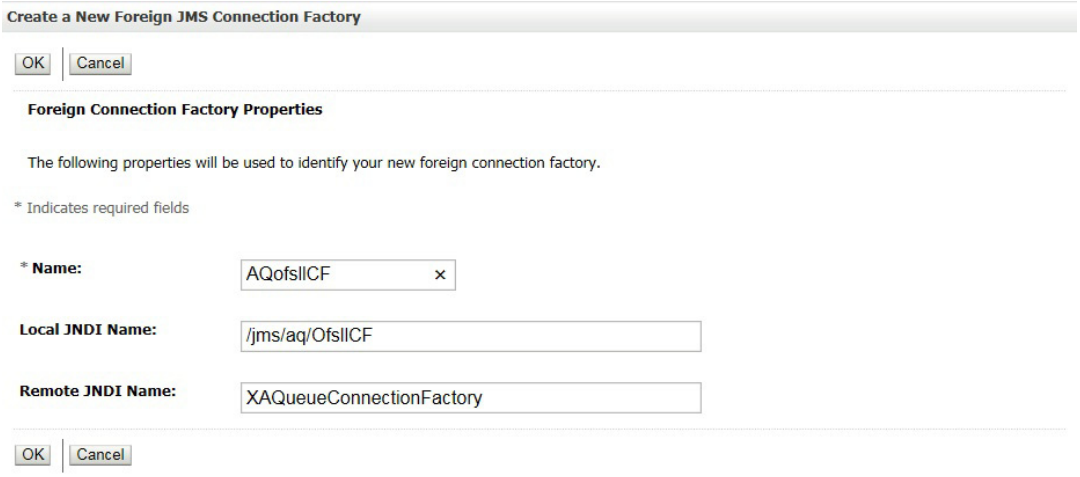


16. Specify the following details:
    - Enter JNDI Initial Context Factory as ‘oracle.jms.AQjmsInitialContextFactory’.
    - JNDI Properties as ‘datasource=jdbc/aqjmsdb’.
    - Ensure ‘Default Targetting Enabled’ option is selected.
17. Click ‘Save’.

18. Select ‘Destinations’ Tab and click ‘New’ to create new destination. The following window is displayed.



19. Specify the following details:
    - Name: AQJMSQueue
    - LocalJNDI Name: /jms/aq/OfsllQueue
    - Remote JNDI Name: Queues/OFSLL_OUTBOUND_Q
20. Click ‘OK’ and save the changes.
21. Select ‘Connection Factories’ Tab and click ‘New’ to add new connection factory. The following window is displayed.



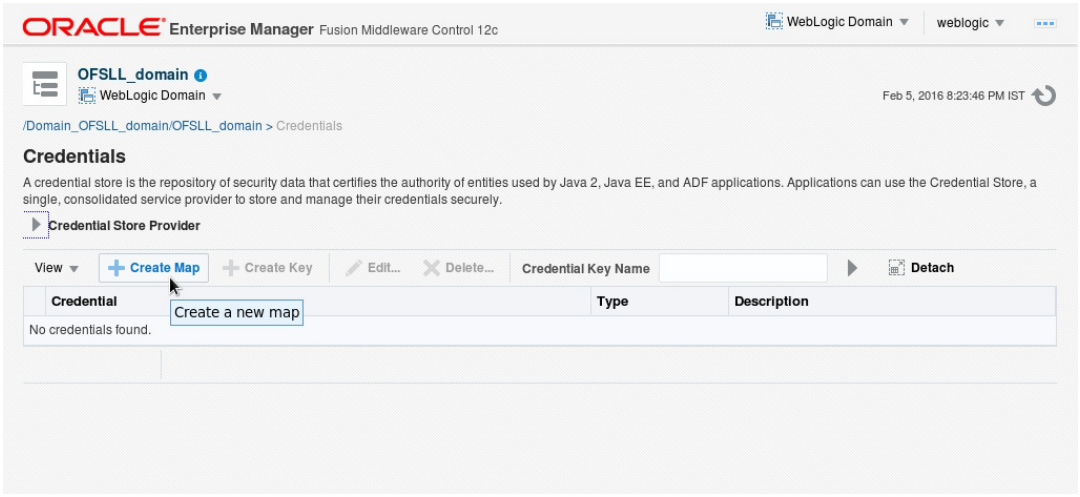
22. Specify the following details:
    - Name: AQofsllCF
    - Local JNDI Name: /jms/aq/OfsllCF
    - Remote JNDI Name: XAQueueConnectionFactory
23. Click ‘OK’ and save the changes.

7.3 Create Credentials and System Policies

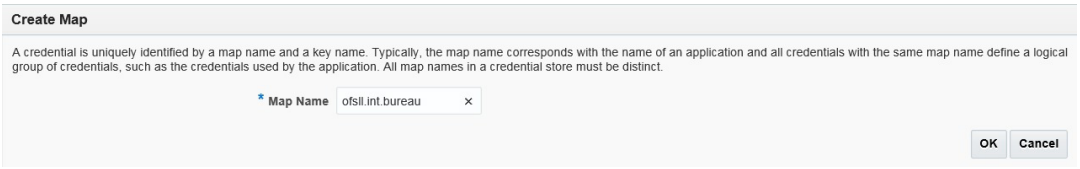
In order Configure MDB flow, you need to create credentials and system policies. The credentials are accessed through CSF framework which is managed by Oracle Weblogic Server. The keys are managed by Maps and Maps need to be given with Permissions.

1. Login to Oracle Enterprise Manager (http://hostname:port/em).

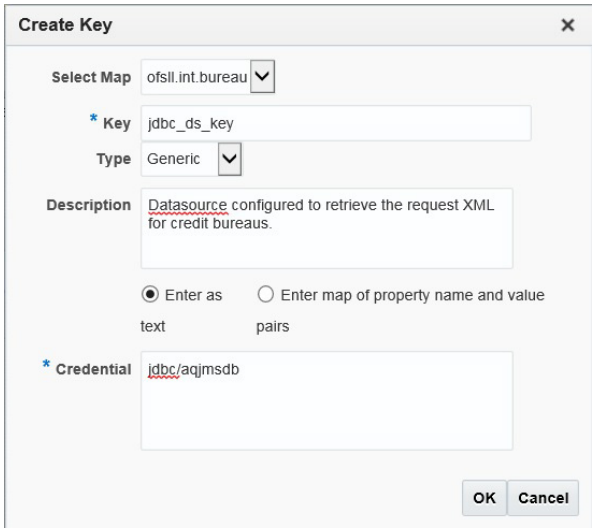
2. On the left panel, right click on OFSLL_domain and select Security > System Policies > Credentials. The following window is displayed.



3. Click ‘Create Map’. The following window is displayed.



4. Enter Map Name as ‘ofsll.int.bureau’ and click ‘OK’.
5. Click ‘Create Key’. The following window is displayed.



6. Specify the following details:
   - Select Map as ‘ofsll.int.bureau’ from the drop down list.
   - Specify Key as ‘jdbc_ds_key’
   - Select Type as ‘Generic’ from the drop down list.
   - Specify the Credential as ‘jdbc/aqjmsdb’.
7. Click ‘OK’.

8. Similarly you need to create the following Maps and corresponding keys as indicated in following table.

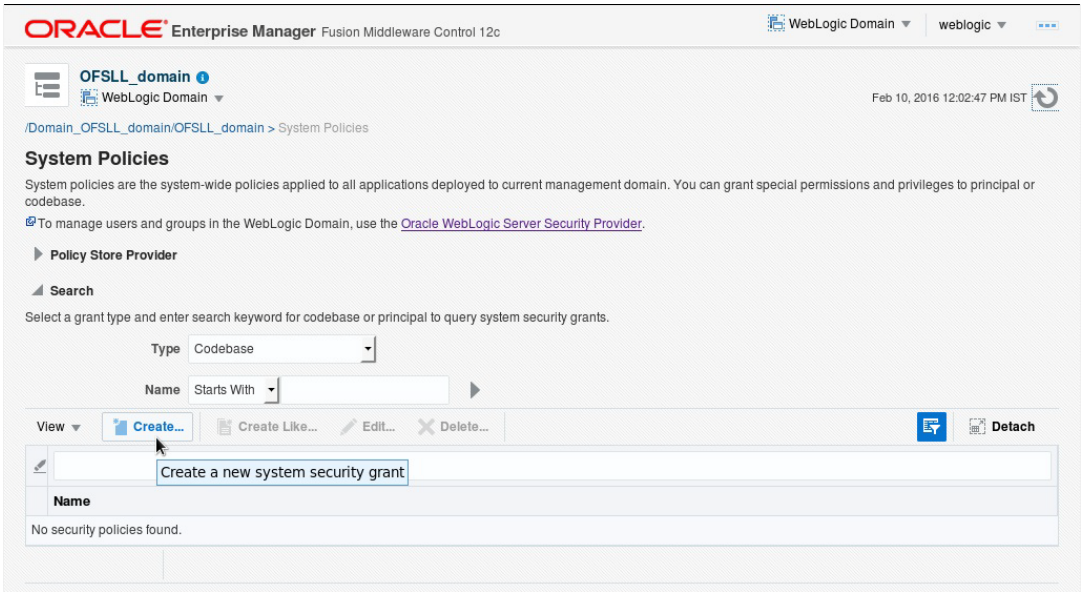
Maps	Keys	Description
ofsll.int.bureau		This map is used to setup keys for all credit bureau interfaces
	ProxyServer	Name of the proxyServer to be configured
	ProxyPort	Port to which ProxyServer is running.
	ExpEcalsURL	The Experian Connection URL to be configured.
	ExpDirectExperianEnabled	If you set value as true, then you would be setting ecals response URL. Else, the Ecals request URL
	ExpCertPath	The location of .jks file which contains the valid certificate for Experian Credit Bureau.
	ExpBusUserNamePassword	Login Credentials to be configured for Experian Business reports.
	ExpConUserNamePassword	Login Credentials to be configured for Experian Consumer reports.
	EfxURL	The Equifax Connection URL to be configured.
	EfxCertPath	The location of .jks file which contains the valid certificate for Equifax Credit Bureau.
	EfxUserNamePassword	Login credentials to be configured for accessing Equifax Reports.
	TucCertPath	The location of .p12 file which contains valid certificate for Transunion Bureau .
	TucCertPassword	The password that requires to read the valid .p12 certificate for the Transunion Bureau.
	TucUserNamePassword	Login credentials to be configured for accessing Transunion reports
	TucConnectionURL	The Transunion URL to be configured.
ofsll.int.outbound		This map is used to setup keys for the RouteOne and Dealer track call back from OFSLL.
	roUserNamePassword	Login Credentials used at the time of call back from OFSLL to RouteOne Interface.
	dtUsernamePassword	Login Credentials used at the time of Call back from OFSLL to Dealer Track Interface.
ofsll.int.bip		This Map is used to setup all the Keys required to setup interface with BIP to generate reports.
	local_top_dir	Define the path of the local BIP server where you would like place the generated BIP reports.

Maps	Keys	Description
	email_from_addr	Define the From Email address to be used while sending email for the generated BIP reports.
	emailBodyContentPath	The path for 'file.properties' file that contains the content of the subject and body required while sending letter, report or correspondence as mail to the applicant or producer. For example; /tmp/file.properties *Refer to note below for details on 'file.properties' file creation for email configuration.
	fax_server	Configure the name of Fax server to be used to fax the generated BIP reports.
ofsll.int.file transfer		This map is used to setup keys for all credit bureau interfaces
	sftp_key	Credentials to login to SFTP server(Username/ Password)
	sftp_top_dir	Top root directory for SFFTP server
	sftp_servers	SFTP server names
ofsll.int.security	bip_key	This is BIP login credentials

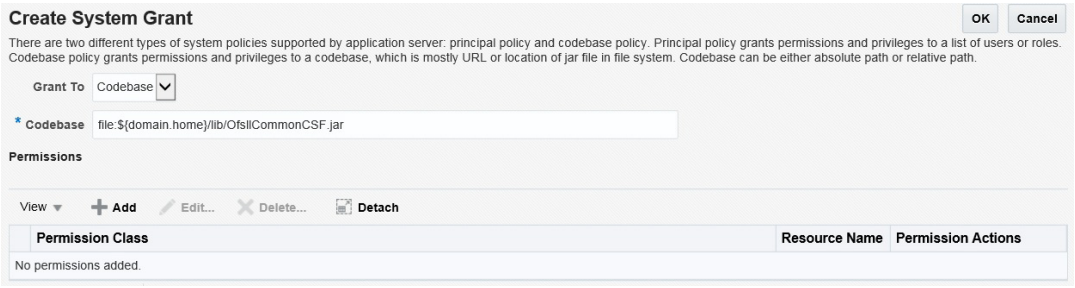
* A new file(file.properties) needs to be created and copied to the application server in the same path as mentioned in the value corresponding to the key 'emailBodyContentPath' under the map 'ofsll.int.bip'. The file should have the following contents:

- letter_subject='Text that is configurable and would be the subject of the mail'
- letter_body='Text that is configurable and would be the body of the mail'
- correspondence_subject='Text that is configurable and would be the subject of the mail'
- correspondence_body='Text that is configurable and would be the body of the mail'
- report_subject='Text that is configurable and would be the subject of the mail'
- report_body='Text that is configurable and would be the body of the mail'

9. On the left panel, right click on OFSLL_domain and select Security > System Policies. The following window is displayed.

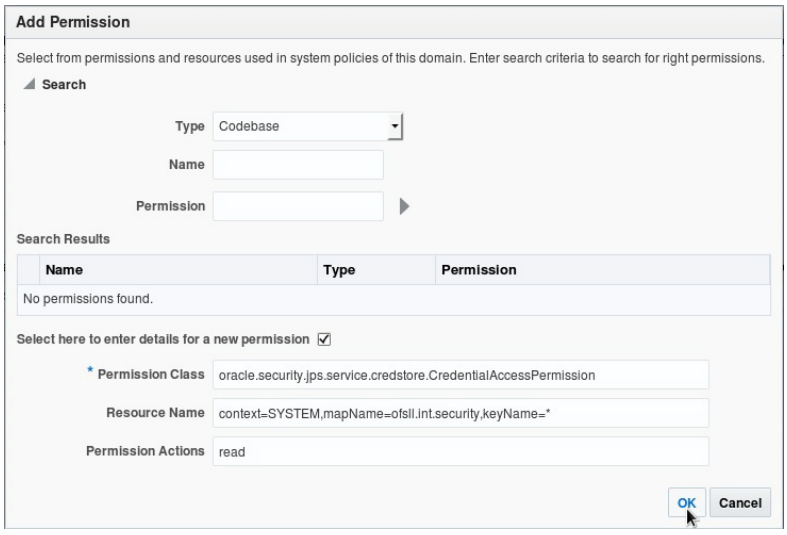


10. Click ‘Create’. The following window is displayed.



11. Enter the codebase as ‘file:${domain.home}/lib/OfsllCommonCSF.jar’

12. Click ‘Add’. The following window is displayed.



13. Select the check box 'Select here to enter details for a new permission'.

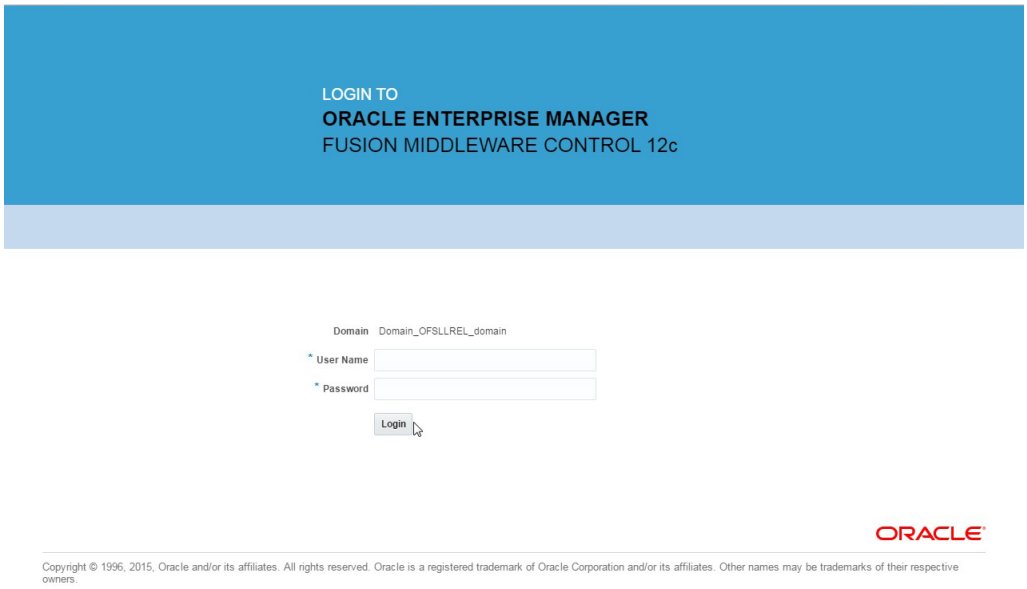
14. Specify the following details as the first permission class.

Permission Class	Resource Name	Permission Actions
oracle.security.jps.service.cred-store.CredentialAccessPermission	context=SYSTEM,mapName=ofsll.int.bureau,keyName=*	read
oracle.security.jps.service.cred-store.CredentialAccessPermission	context=SYSTEM,mapName=ofsll.int.filetransfer,keyName=*	read
oracle.security.jps.service.cred-store.CredentialAccessPermission	context=SYSTEM,mapName=ofsll.int.outbound,keyName=*	read
oracle.security.jps.service.cred-store.CredentialAccessPermission	context=SYSTEM,mapName=ofsll.int.bip,keyName=*	read
oracle.security.jps.service.cred-store.CredentialAccessPermission	context=SYSTEM,mapName=ofsll.http.listener.jndi,keyName=*	read

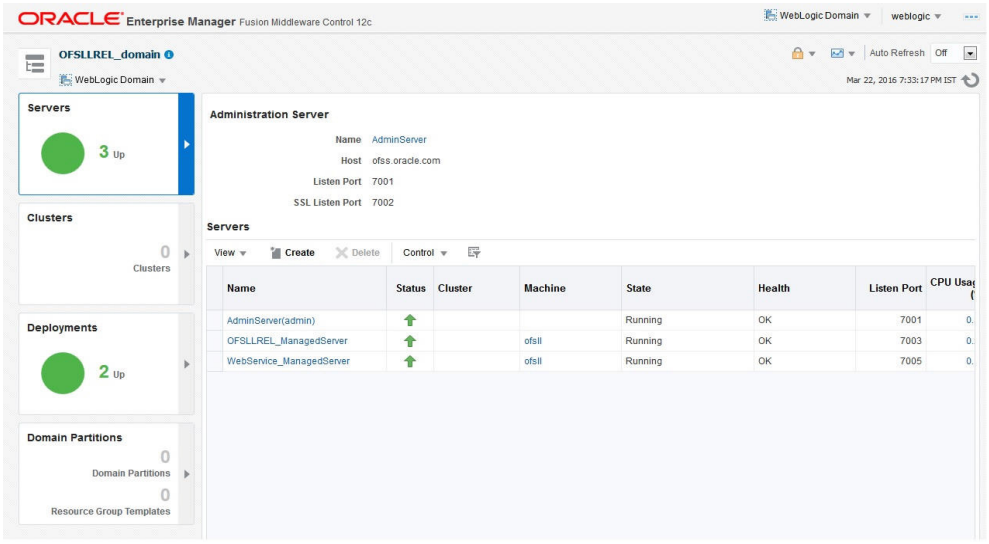
15. Click ‘OK’.

7.4 Deploy MDB EJB

1. Login to Web Logic application server enterprise manager (e.g.:http://hostname:port/em)

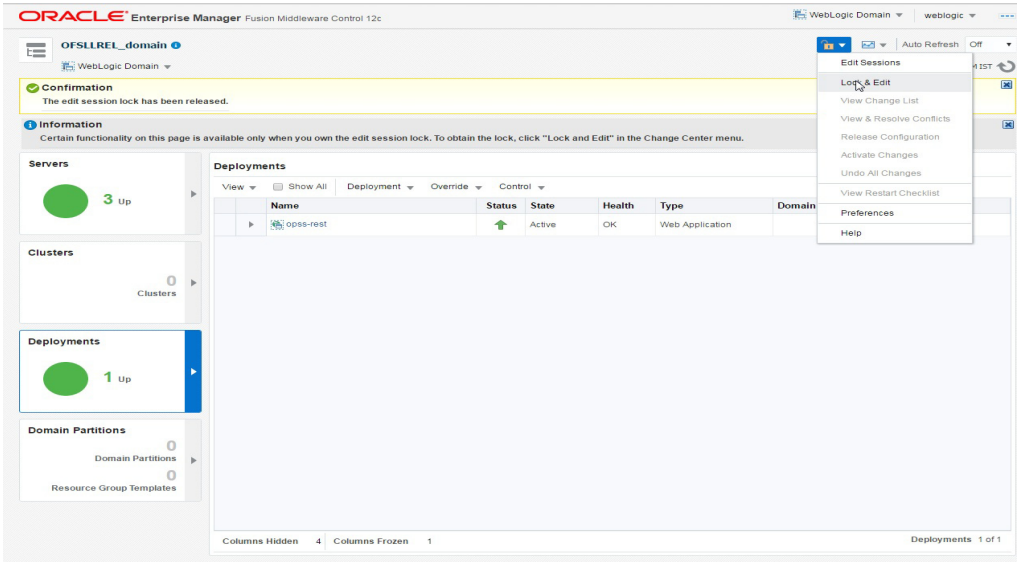


2. Enter valid login credentials. The following window is displayed.

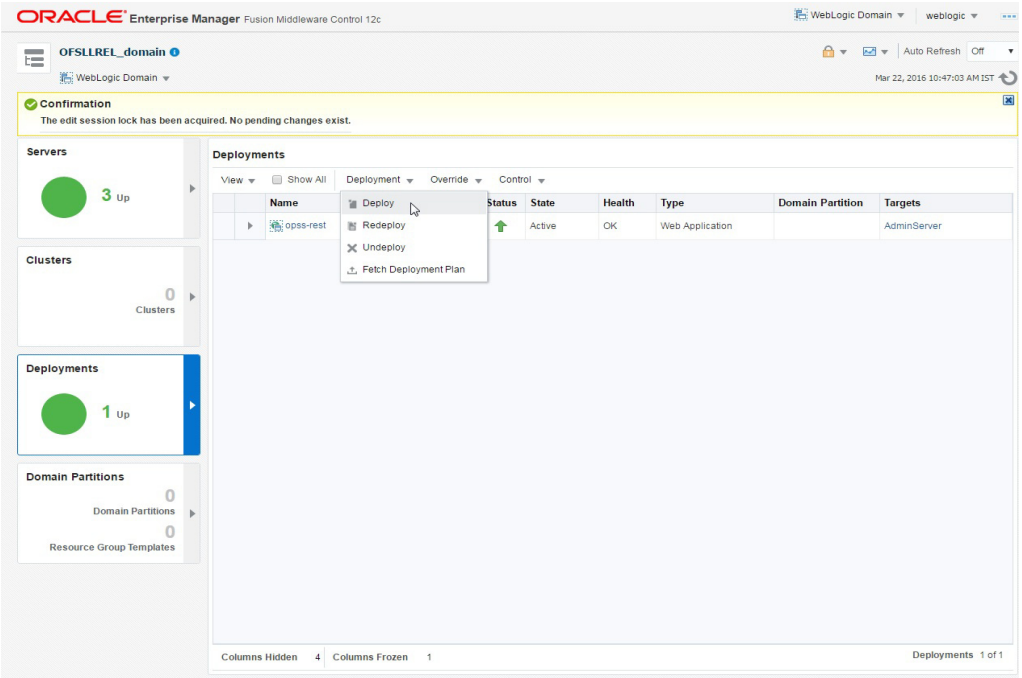


3. Click ‘Deployment’ in the left panel.

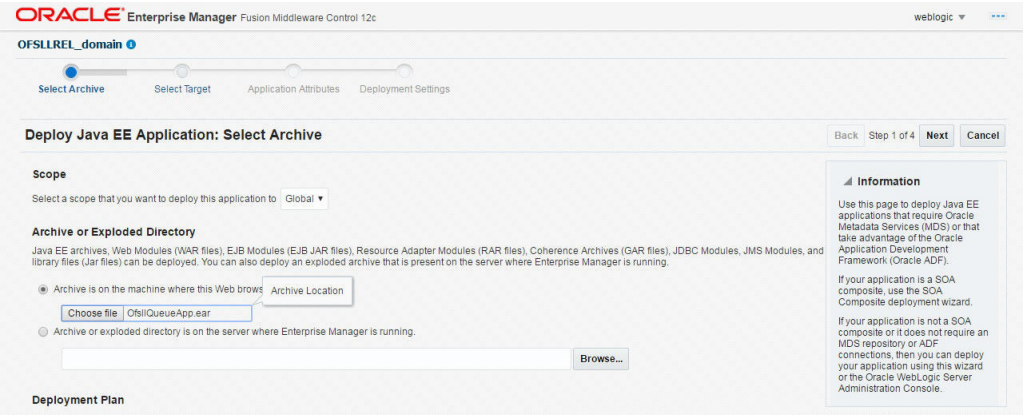
4. Select ‘Lock & Edit’ option in the lock drop-down list available in the header. The following window is displayed.



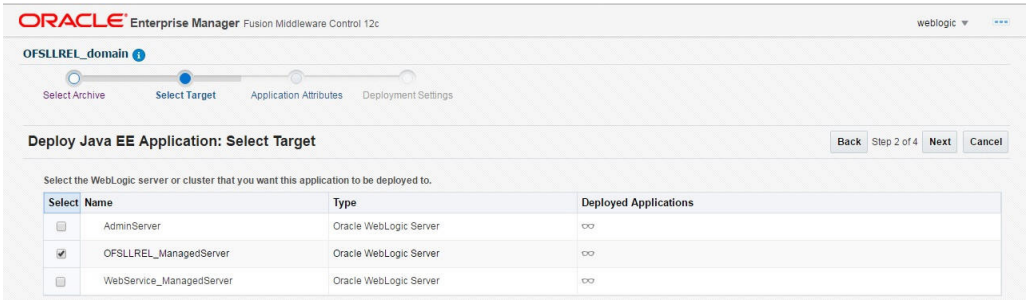
5. Select ‘Deploy’ from the Deployment drop-down list. The following window is displayed.



6. The following window is displayed.

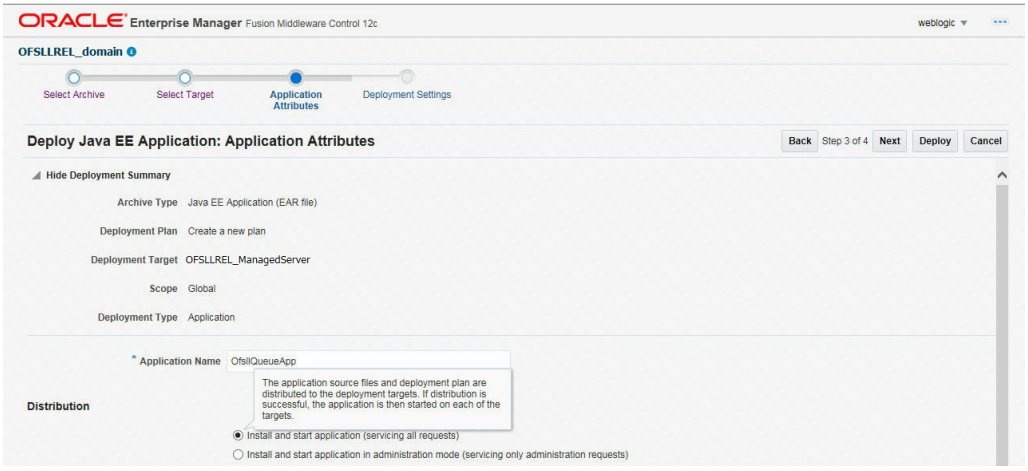


7. Browse to the folder containing the MDB EJB. Eg: C:/OfsllQueueApp.ear
8. Click ‘Next’. The following window is displayed.

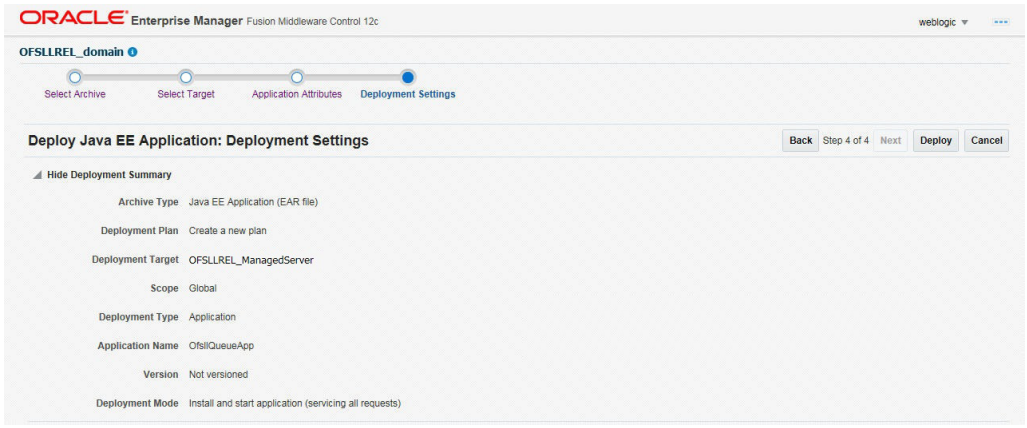


9. Select the server on which the MDB EJB needs to be deployed.

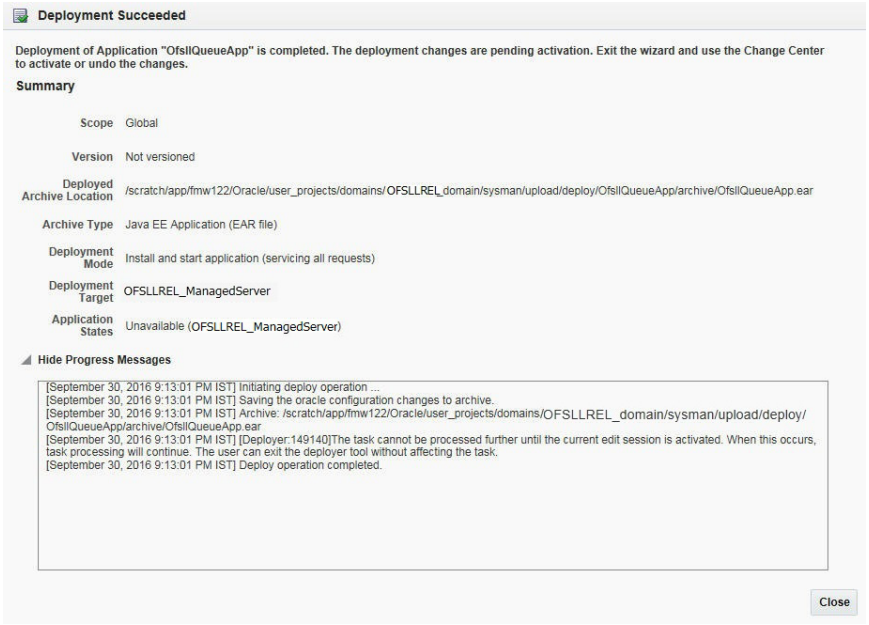
10. Click ‘Next’. The following window is displayed.



11. Select the option ‘Install and start application (servicing all requests)’.
12. Check the context root and click ‘Next’.The following window is displayed.



13. Click ‘Deploy’. On successful deployment, the following window is displayed.



14. Click ‘Close’. Post deployment, you need to activate the changes by selecting ‘Active Changes’ option from ‘Edit Session’ drop-down list as indicated in step 4 above.

Note

While starting the ‘OFSLLREL_ManagedServer’, always start with option ‘-DUseSun-HttpHandler=true’ to enforce the weblogic server to uses SUN SSL implementation.

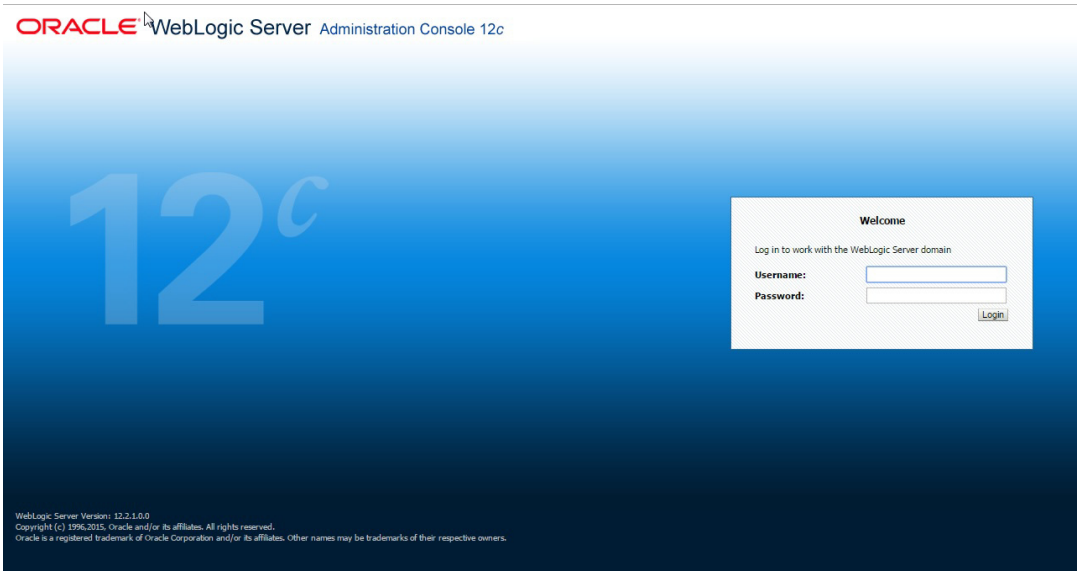
8. Configuring RESTful WebService

Follow the below steps to configure RESTful WebService.

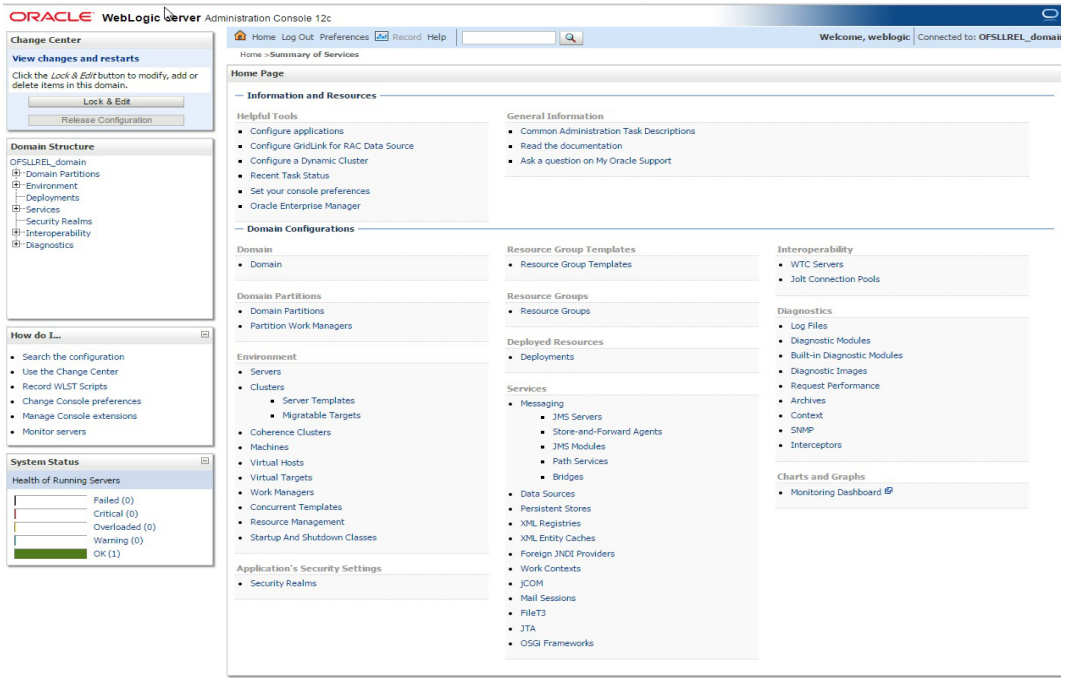
- Create Data Source for RESTful WebService
- Deploy RESTful WebService

8.1 Create Data Sources for RESTful WebService

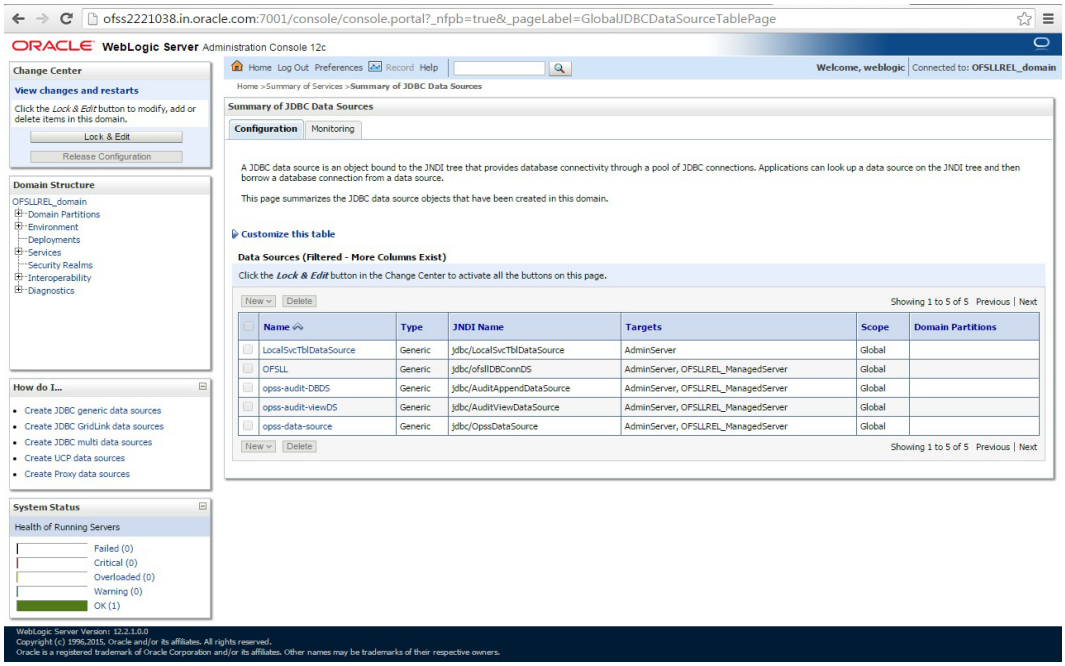
1. Login to Oracle Weblogic 12c console (http://hostname:port/console).



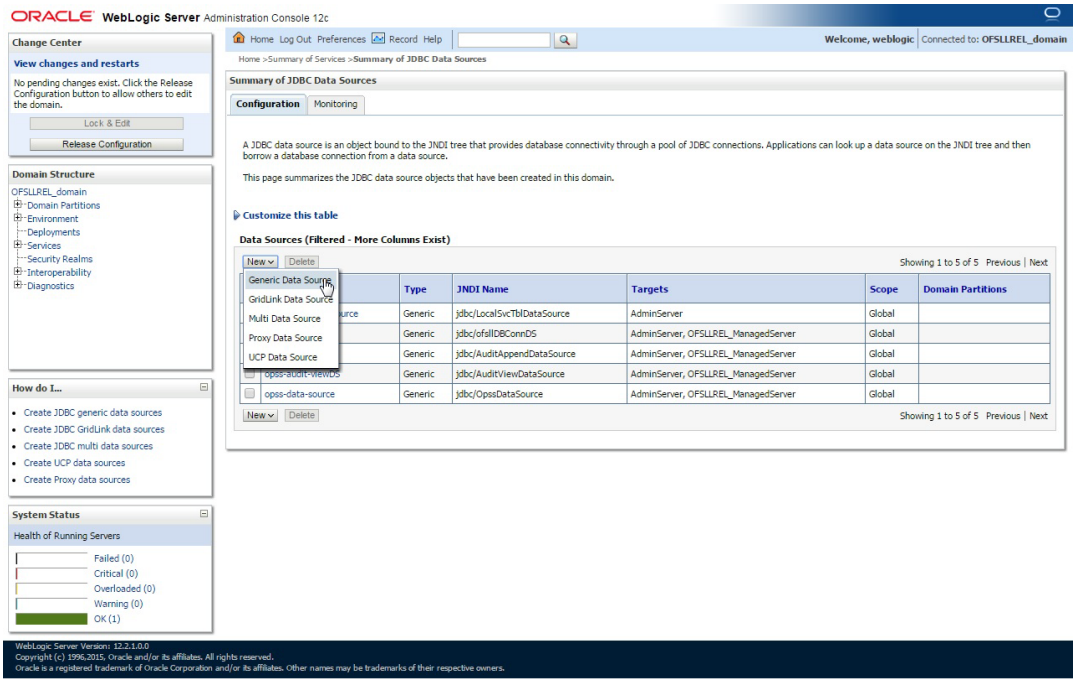
2. On successful login, the following window is displayed.



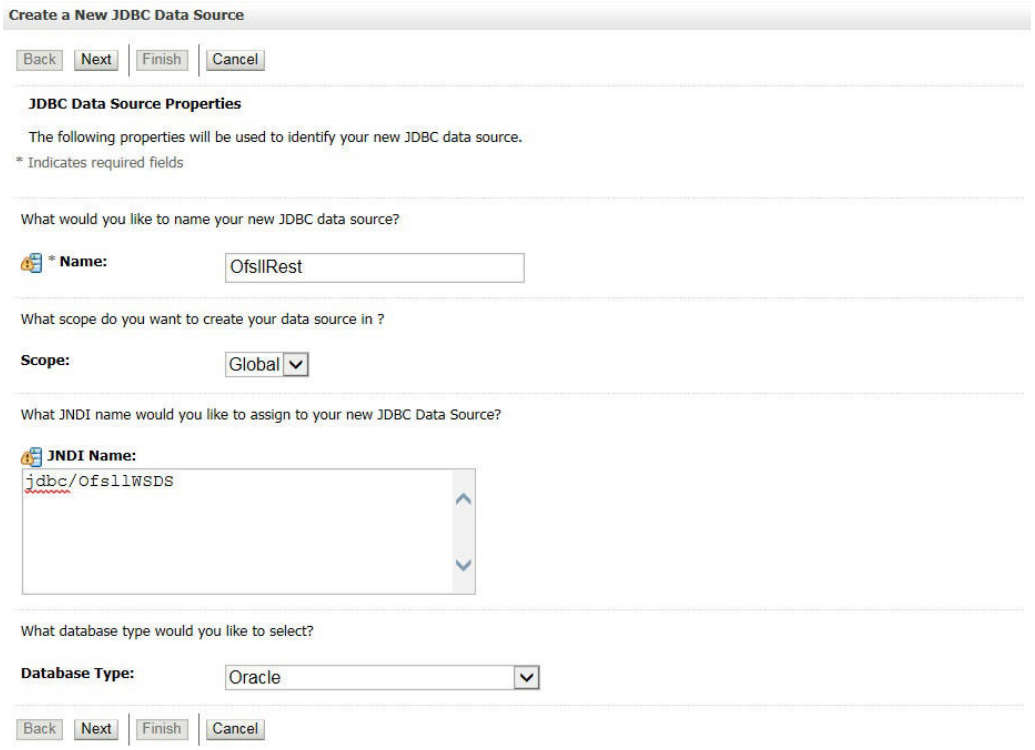
3. Click Domain Name > Services > Data Sources. The following window is displayed.



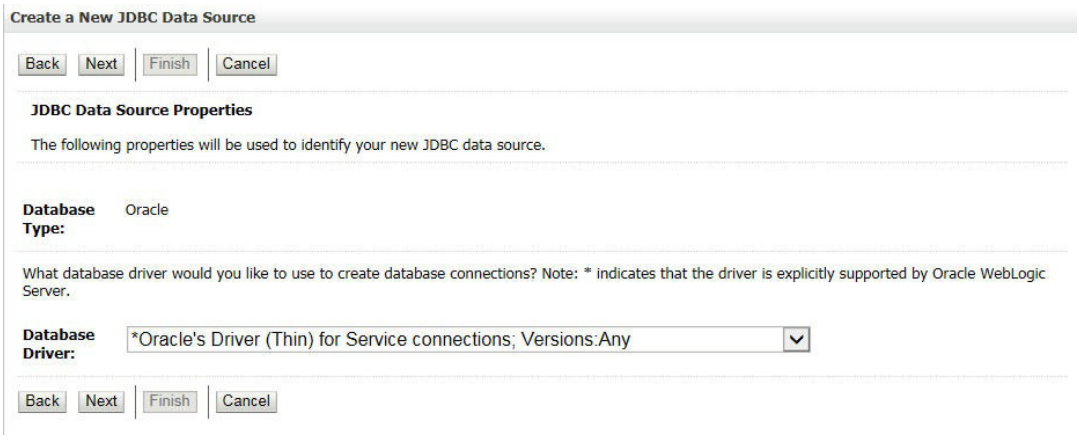
4. Click ‘Lock & Edit’ button on the left panel. Click ‘New’ on right panel and select ‘Generic Data Source’.



5. The following window is displayed.

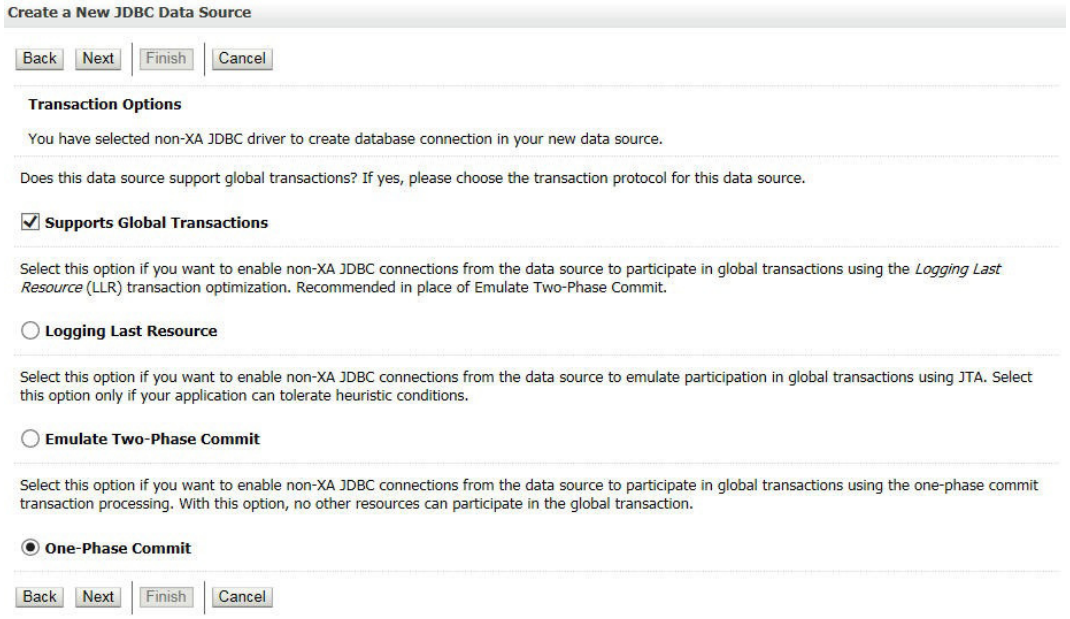


6. Specify the following details:
   - Enter Data source Name
   - Enter the JNDI Name as ‘jdbc/ofsllWSDS’.
   - Select ‘Oracle’ as Database Type.
7. Click ‘Next’. The following window is displayed.

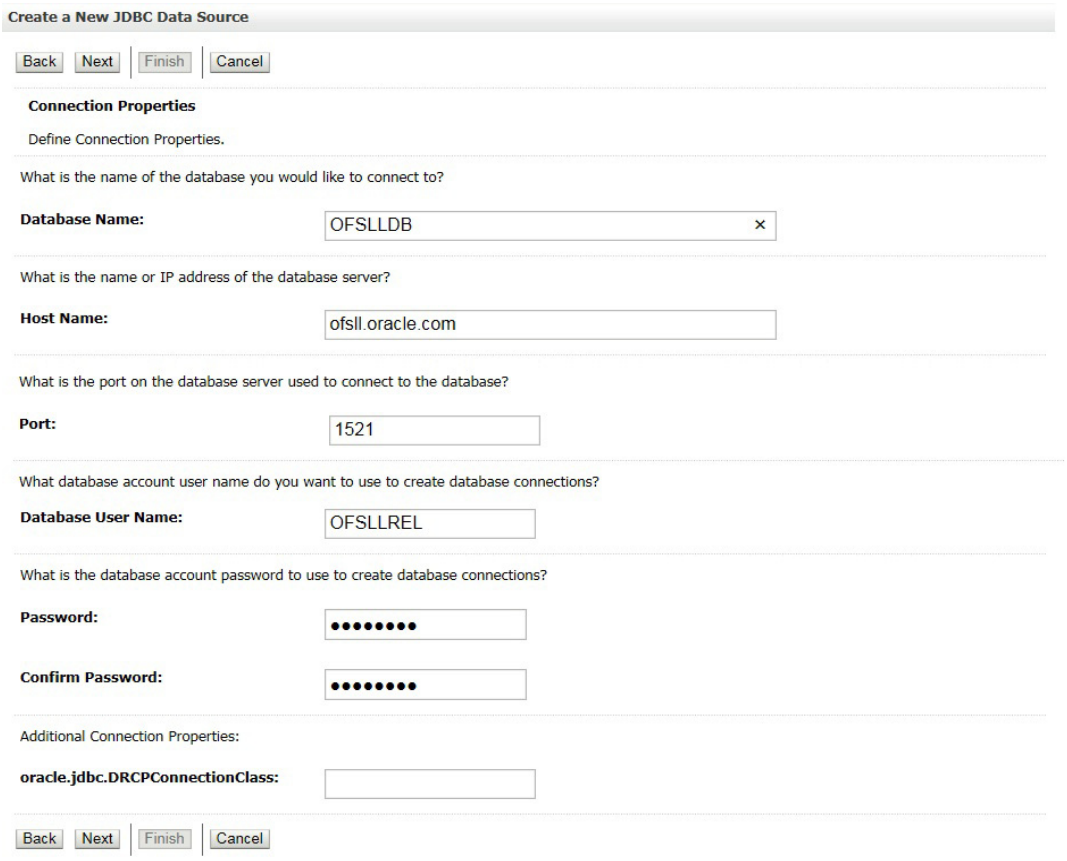


8. Select the Database Driver ‘Oracle's Driver(Thin) for Services connections;Versions:Any’.

9. Click ‘Next’. The following window is displayed.

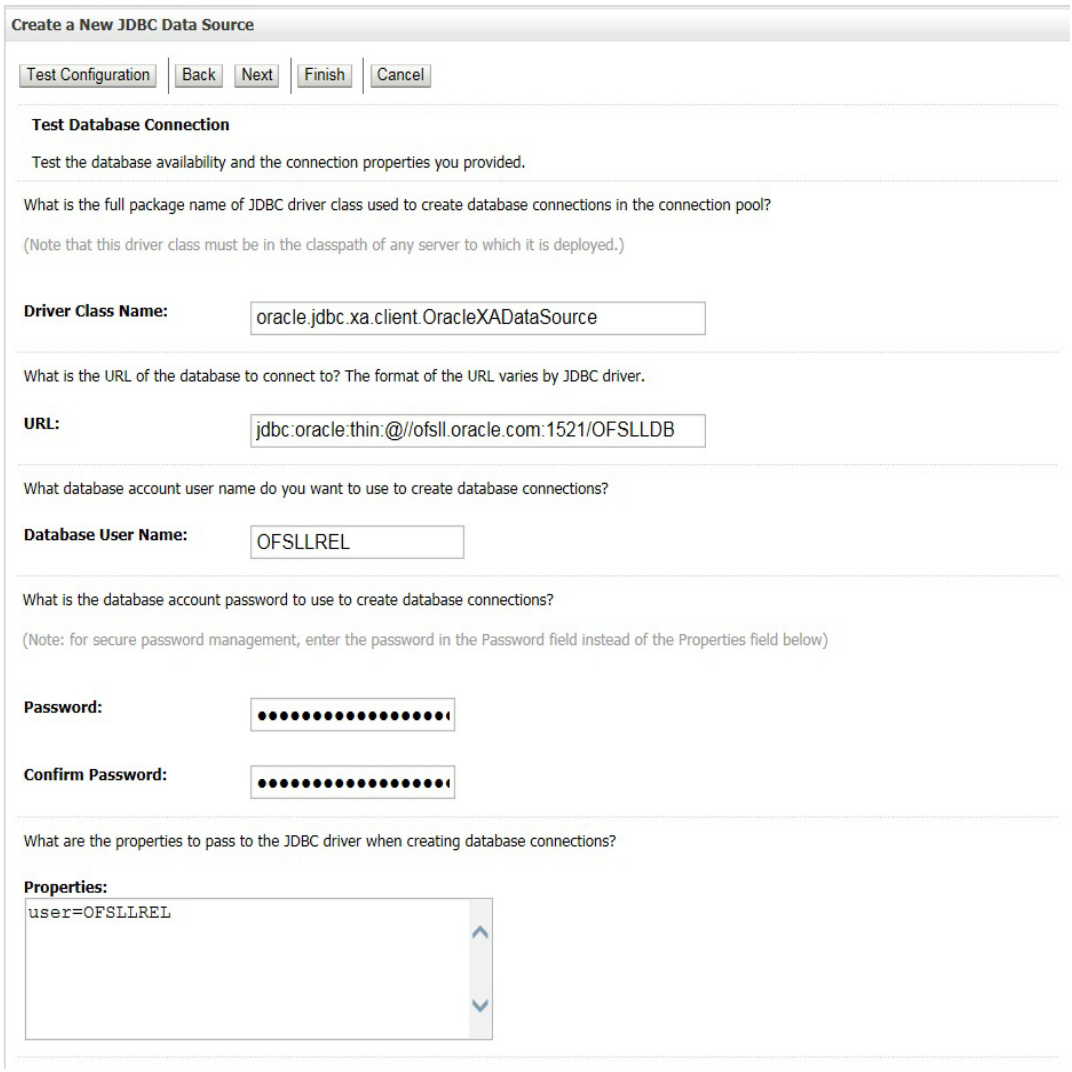


10. Click ‘Next’. The following window is displayed.

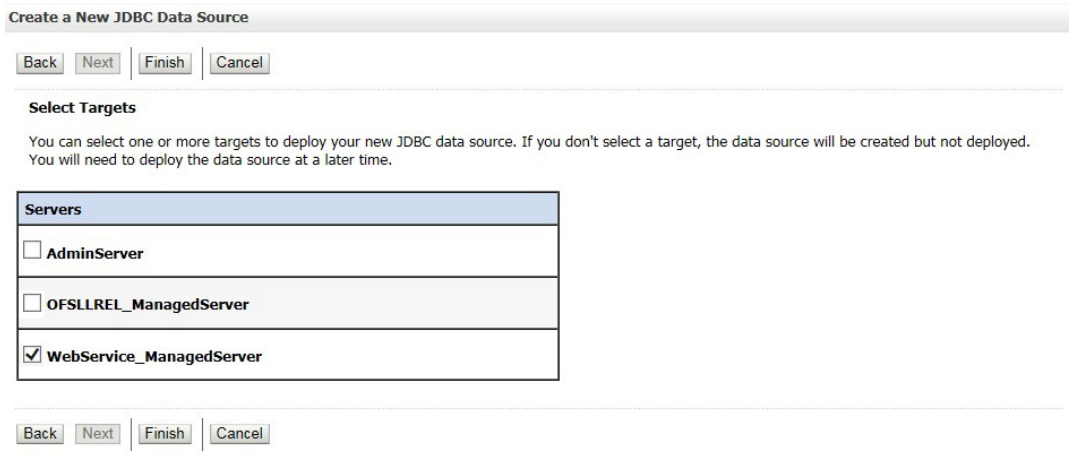


11. Enter the Database details.

12. Click ‘Next’. The following window is displayed.



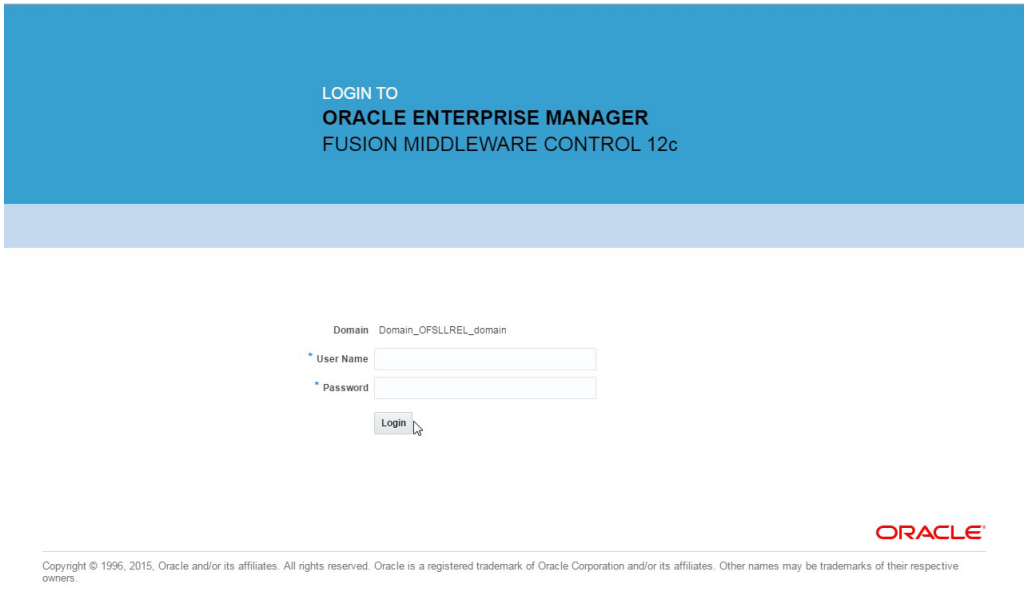
13. Click ‘Test Configuration’. On completion, displays a confirmation message as ‘Connection test succeeded’.
14. Click ‘Next’. The following window is displayed.



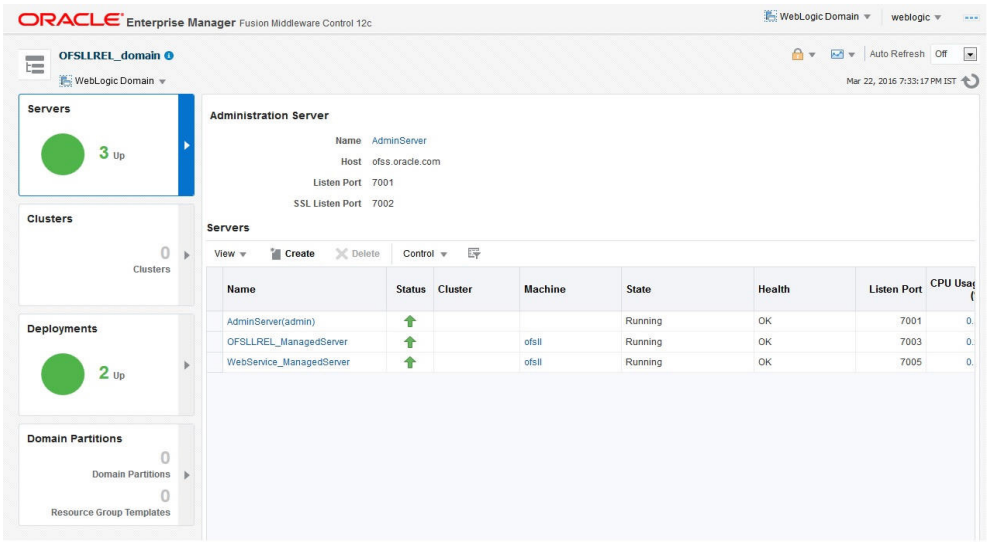
15. Select target Server as ‘WebService_ManagedServer’.
16. Click ‘Finish’ to activate the changes.

8.2 Deploy RESTful WebService

1. Login to Web Logic application server enterprise manager (e.g.:http://hostname:port/em)

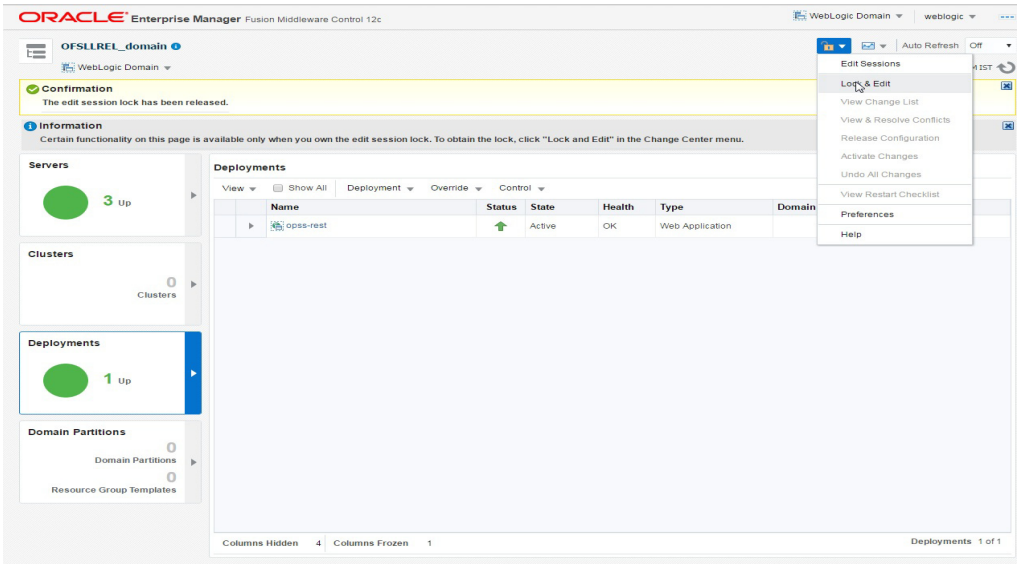


2. Enter valid login credentials. The following window is displayed.

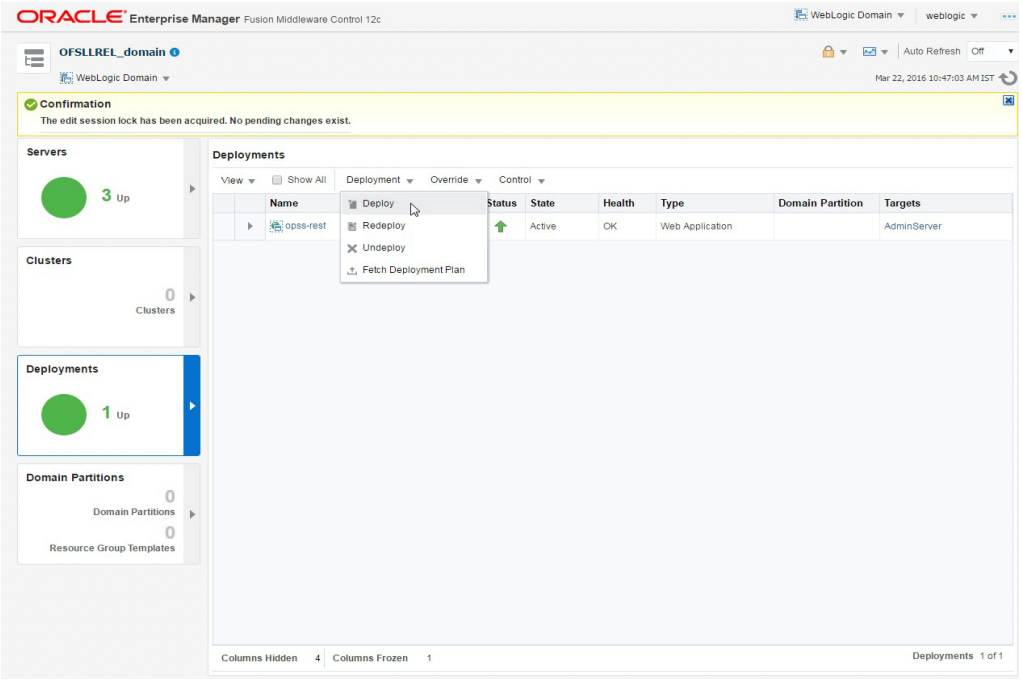


3. Click ‘Deployment’ in the left panel.

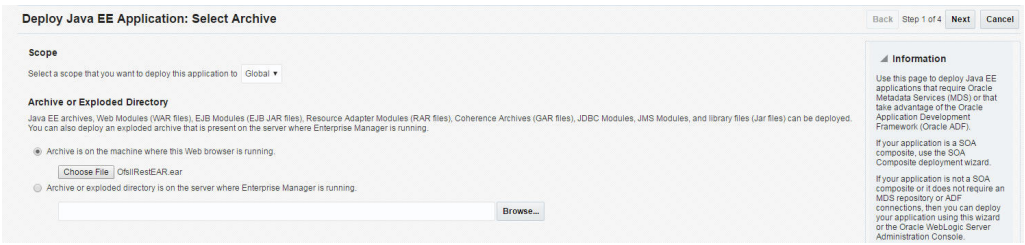
4. Select ‘Lock & Edit’ option in the lock drop-down list available in the header. The following window is displayed.



5. Select ‘Deploy’ from the Deployment drop-down list. The following window is displayed.

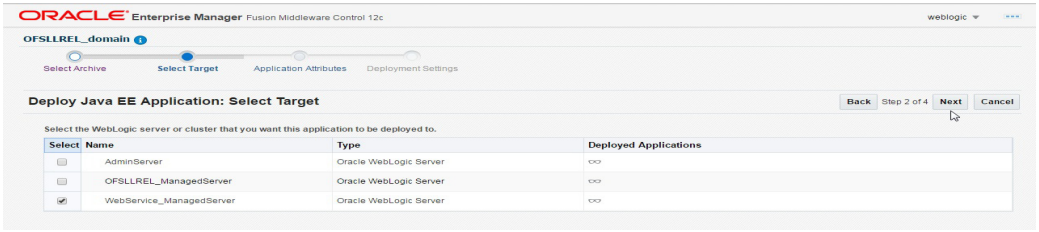


6. The following window is displayed.

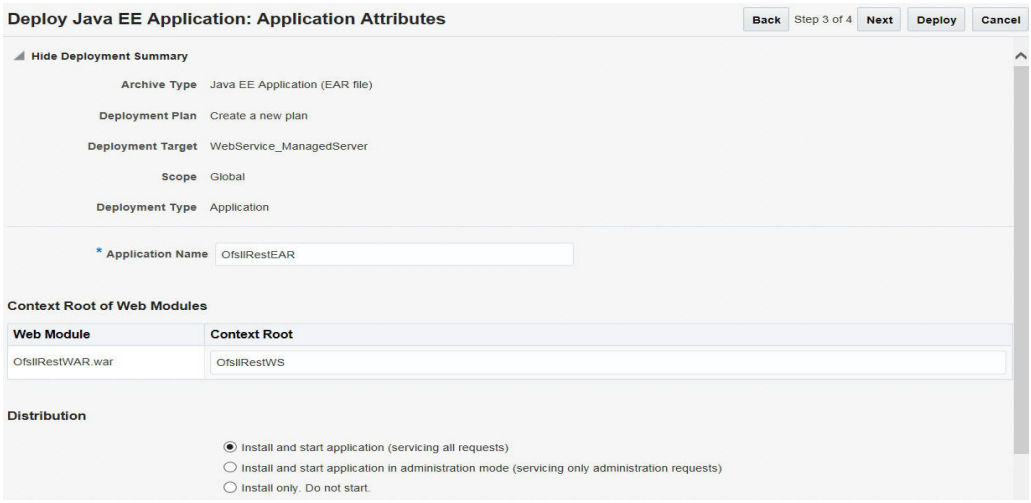


7. Browse to the folder containing the WebService. Eg: C:/OfsllRestEAR.ear

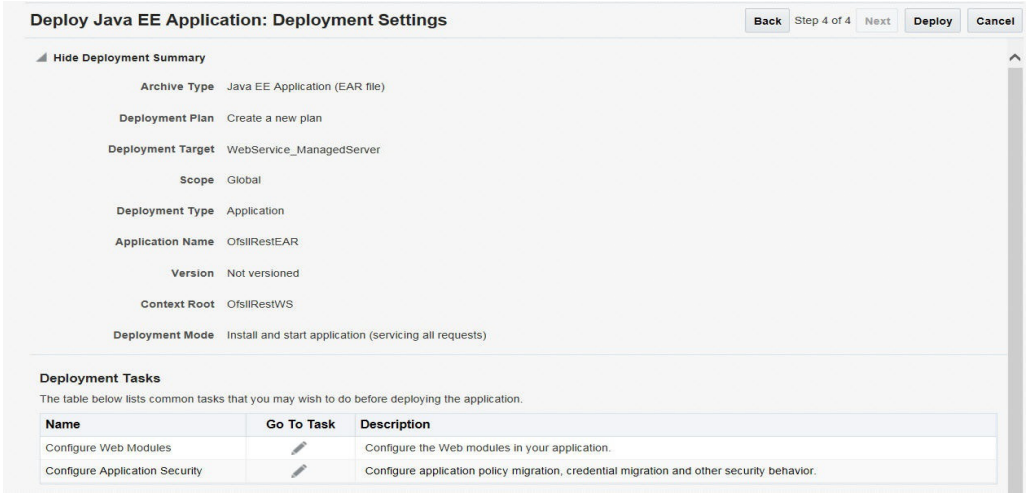
8. Click ‘Next’. The following window is displayed.



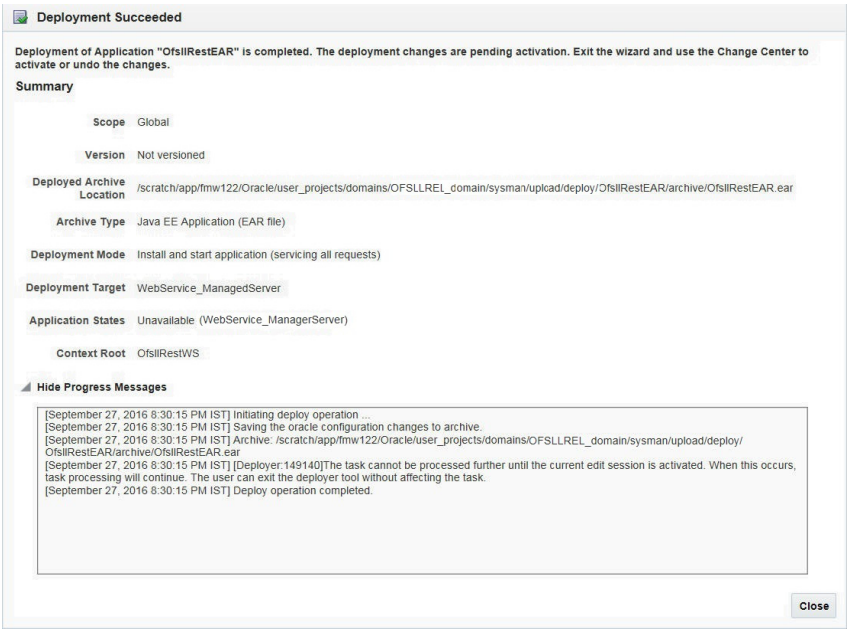
9. Select the server on which the WebService needs to be deployed.
10. Click ‘Next’. The following window is displayed.



11. Select the option ‘Install and start application (servicing all requests)’.
12. Check the context root and click ‘Next’.The following window is displayed.



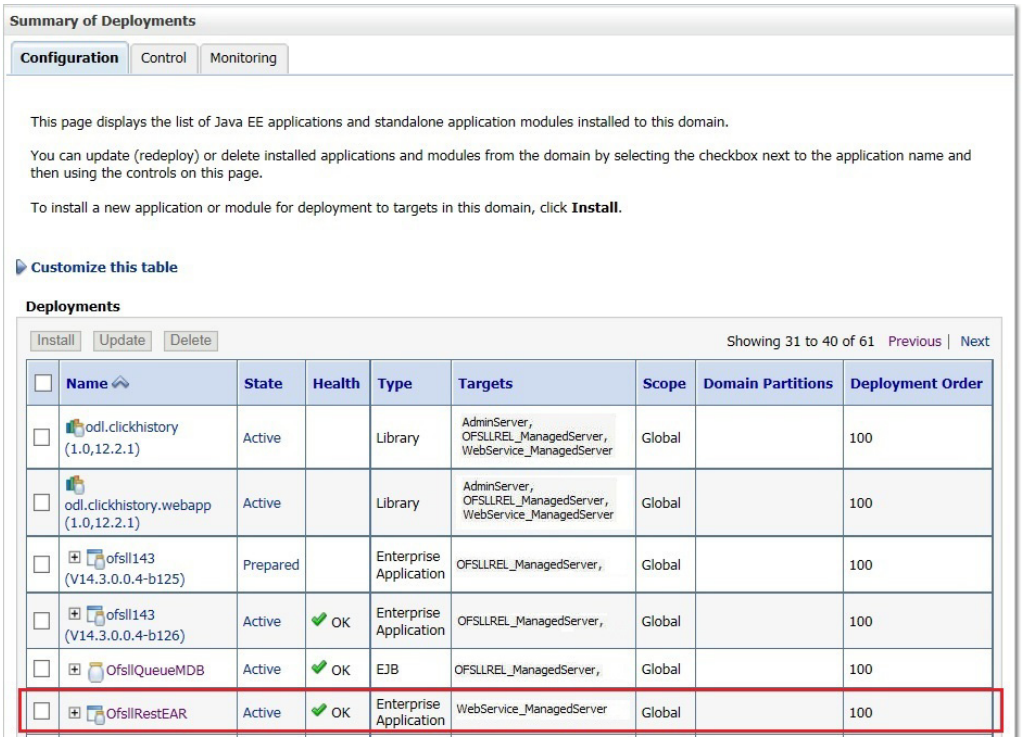
13. Click ‘Deploy’. On successful deployment, the following window is displayed.



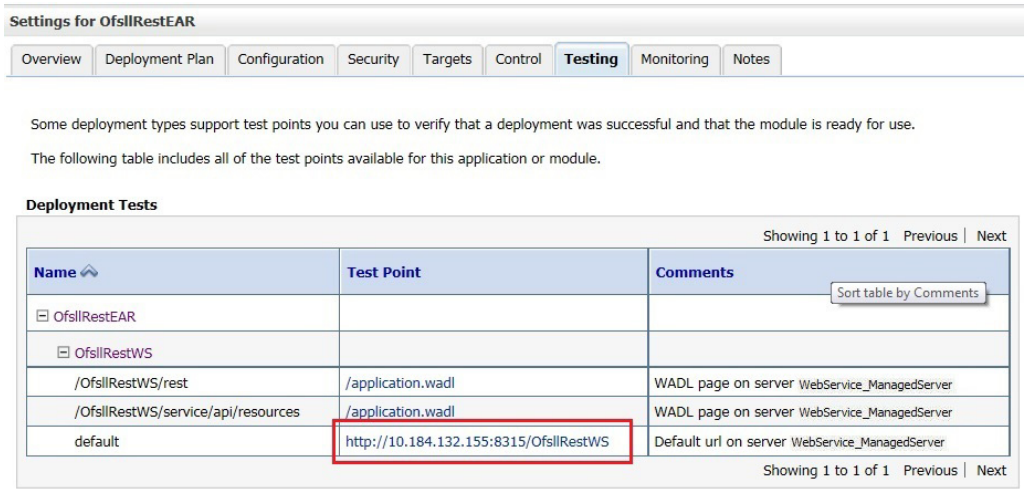
14. Click ‘Close’. Post deployment, you need to activate the changes by selecting ‘Active Changes’ option from ‘Edit Session’ drop-down list as indicated in step 4 above.

To Identify the RESTful Webservice URL

1. Login to WebLogic Server 12c console (http://hostname:port/console).
2. Click ‘Deployments’ Under Configuration tab and select ‘OfsllRestEAR’ services. The following window is displayed.



3. Click ‘Testing’ tab and expand ‘OfsllRestWS’. The following window is displayed.



4. You can view the 'OfsllRestful' Services URL as shown.

A. Appendix - Configuration parameters

Refer to the following section for details on configuration parameters.

Modifying Configuration Files

Route One Configuration

Property Name	Property Value	Description	Remarks
Configuration File: **ro_servlet_init.conf**			
LOSApplicationRequestServiceURL	http://<HOST NAME>:<PORT>/<CONTEXT ROOT>/LOSApplicationRequestService	URL for OFSLL New application WebService	Refer the recommended context root table for CONTEXT ROOT. Sample URL: http://hostname:port/dbkls-xws-app-ro/LOSApplicationRequestService
LOSApplicationUpdateServiceURL	http://<HOST NAME>:<PORT>/<CONTEXT ROOT>/LOSApplicationUpdateService	URL for OFSLL application update WebService	Refer the recommended context root table for CONTEXT ROOT. Sample URL: http://hostname:port/dbkls-xws-app-ro/LOSApplicationUpdateService
LOSeContractWebServiceServiceURL	http://<localhost>:<port>/dbkls-xws-app-ro/LOSEContractService	URL to validate and receive the contract information	Refer the recommended context root table for CONTEXT ROOT. Sample URL: http://hostname:port/dbkls-xws-app-ro/LOSEContractService
keystoreLocation	config/dls_cacerts	Parameter to set keystore location	The keystore location should not be changed and keystore should be available in this location
postURL	https://messaging.itl.routeone.net/IF1_ITL/CASSBMessagingReceiver	URL for posting acknowledgement message to Route one	Verify the URL with Route One
fileLocation	/tmp	temporary file directory	The directory for temporary files. Make sure that such directory exists on the deployment server
postTimeout	10	Parameter to set timeout value to post XML message to third party.	Timeout value should be specified in number of seconds.

Property Name	Property Value	Description	Remarks
useProxy	0	Parameter to switch on/off proxy use. By default use of proxy is switched off. To use proxy setup value to "1", else setup value as "0"	
proxyHost	abc.com	Parameter to set proxy host. This is required if useProxy is set to 1.	
proxyPort	80	Parameter to set proxy port. This is required if useProxy is set to 1.	
Configuration File: **dbkws_xae_init.conf**			
lenderId	SOME-LENDER_ID	The finance source identifier used in the dealer file	SOMELENDER_ID is a sample value
lenderName	SOME_LENDER_NAME	The finance source name used in the dealer file	SOME_LENDER_NAME is a sample value
application-Source	ROUTEONE	Parameter to set application source	Don't modify this value
dealerEle-mentName	DEALER_DE-TAILS	Name of XML ele-ment used for the dealer details used in the dealer file	DEALER_DETAILS is a sample value
fileLocation	/tmp		The directory for tempo-rary files. Make sure that such directory exists on the deployment server
fileDelimiter	,	Delimiter used in dealer header file	Needed only for Route-One
jndiLookupDa-taSource	jdbc/IN1Huk-WznG0b4esj	Parameter that defines the JNDI look up for the applicaton datasource	Don't modify the value
keystoreLoca-tion	config/dls_cacerts	Parameter to set location for public/pri-vate key store for HTTPS posting and for XML Digital Sig-nature	The keystore location should not be changed and keystore should be available in this location

Property Name	Property Value	Description	Remarks
postTimeout	10	Parameter to set timeout value to post XML message to third party.	Timeout value should be specified in number of seconds.
useProxy	0	Parameter to switch on/off proxy use. By default use of proxy is switched off. To use proxy setup value to "1", else setup value as "0"	
proxyHost	abc.com	Parameter to set proxy host. This is required if useProxy is set to 1.	
proxyPort	80	Parameter to set proxy port. This is required if useProxy is set to 1.	
Configuration File: **dbkws_xec_init.conf**			
jndiLookupDataSource	jdbc/IN1HukWznG0b4esj	Parameter that defines the JNDI look up for the applicaton datasource	Don't modify the value
Configuration File: **dbkws_xcl_init.conf**			
lenderId	SOMELENDER_ID	The finance source identifier used in the dealer file	SOMELENDER_ID is a sample value
lenderName	SOME_LENDER_NAME	The finance source name used in the dealer file	SOME_LENDER_NAME is a sample value
dealerElementName	DEALER_DETAILS	Name of XML element used for the dealer details used in the dealer file	DEALER_DETAILS is a sample value
fileLocation	/tmp		The directory for temporary files. Make sure that such directory exists on the deployment server
fileDelimiter	,	Delimiter used in dealer header file	

Property Name	Property Value	Description	Remarks
jndiLookupDataSource	jdbc/dbk105nlEJBDS	Parameter that defines the JNDI look up for the applicaton datasource	Don't modify the value
keystoreLocation	config/dls_cacerts	Parameter to set location for public/private key store for HTTPS posting and for XML Digital Signature	The keystore location should not be changed and keystore should be available in this location
keystorePassword	changeit	The keystore password	The deafult password is "changeit". Modify the property in case the password is different for the keystore.
sscroKeyAlias	ofss_routeone	OFSLL's private key alias name. The private key is used to sign xml response to RouteOne	You may import the private key with alias "ofss_routeone" else modify the property value to alias used while importing the private key into keystore.
sscroKeyPassword	demotestSSCR1	OFSLL's private key password	demotestSSCR1 is a sample value
Configuration File: **Logging.properties**			
handlers	java.util.logging.FileHandler, java.util.logging.ConsoleHandler		
java.util.logging.FileHandler.level	ALL		
java.util.logging.FileHandler.pattern	/somewhere/logs/dbkls_xws_%g.log		
java.util.logging.FileHandler.limit	1000000		
java.util.logging.FileHandler.count	4		

Property Name	Property Value	Description	Remarks
java.util.logging.FileHandler.append	true		
java.util.logging.FileHandler.formatter	java.util.logging.SimpleFormatter		
java.util.logging.ConsoleHandler.level	WARNING		
java.util.logging.ConsoleHandler.formatter	java.util.logging.SimpleFormatter		
com.ofss.fll.xws.level	FINER	set the logging level for the application	Other Level values -- FINEST,FINE,CONFIG and INFO

Dealer Track

Property Name	Property Value	Description	Remarks
Configuration File: **dt_servlet_init.conf**			
LOSApplicationRequestServiceURL	http://<HOST NAME>:<PORT>/<CONTEXT ROOT>/LOSApplicationRequestService	URL for OFSLL New application WebService	Refer the recommended context root table for CONTEXT ROOT. Sample URL: http://hostname:port/dbkls-xws-app-dt/LOSApplicationRequestService
LOSApplicationUpdateServiceURL	http://<HOST NAME>:<PORT>/<CONTEXT ROOT>/LOSApplicationUpdateService	URL for OFSLL application update WebService	Refer the recommended context root table for CONTEXT ROOT. Sample URL: http://hostname:port/dbkls-xws-app-dt/LOSApplicationUpdateService
postTimeout	10	Parameter to set timeout value to post XML message to third party.	Timeout value should be specified in number of seconds.
useProxy	0	Parameter to switch on/off proxy use. By default use of proxy is switched off. To use proxy setup value to "1", else setup value as "0"	
proxyHost	abc.com	Parameter to set proxy host. This is required if useProxy is set to 1.	
proxyPort	80	Parameter to set proxy port. This is required if useProxy is set to 1.	
Configuration File: **dbkws_xae_init.conf**			
lenderId	SOMELENDER_ID	The finance source identifier used in the dealer file	SOMELENDER_ID is a sample value
lenderName	SOME_LENDER_NAME	The finance source name used in the dealer file	SOME_LENDER_NAME is a sample value
applicationSource	DEALERTRACK	Parameter to set application source	Don't modify this value

Property Name	Property Value	Description	Remarks
dealerElement-Name	DEALER_DETAILS	Name of XML element used for the dealer details used in the dealer file	DEALER_DETAILS is a sample value
fileLocation	/tmp		The directory for temporary files. Make sure that such directory exists on the deployment server
jndi-Lookup-DataSource	jdbc/IN1Huk-WznG0b4esj	Parameter that defines the JNDI look up for the applicaton datasource	Don't modify the value
keystore-Location	config/dls_cacerts	Parameter to set location for public/private key store for HTTPS posting and for XML Digital Signature	The keystore location should not be changed and keystore should be available in this location
keystore-Password	changeit	The keystore password	The deafult password is "changeit". Modify the property in case the password is different for the keystore.
postTimeout	10	Parameter to set timeout value to post XML message to third party.	Timeout value should be specified in number of seconds.
useProxy	0	Parameter to switch on/off proxy use. By default use of proxy is switched off. To use proxy setup value to "1", else setup value as "0"	
proxyHost	abc.com	Parameter to set proxy host. This is required if useProxy is set to 1.	
proxyPort	80	Parameter to set proxy port. This is required if useProxy is set to 1.	
Configuration File: **dbkws_xcl_init.conf**			
lenderId	SOME-LENDER_ID	The finance source identifier used in the dealer file	SOMELENDER_ID is a sample value
lender-Name	SOME_LENDER_NAME	The finance source name used in the dealer file	SOME_LENDER_NAME is a sample value

Property Name	Property Value	Description	Remarks
dealerElement-Name	DEALER_DETAILS	Name of XML element used for the dealer details used in the dealer file	DEALER_DETAILS is a sample value
fileLocation	/tmp		The directory for temporary files. Make sure that such directory exists on the deployment server
fileDelimiter	,	Delimiter used in dealer header file	Needed only for RouteOne
jndiLookupDataSource	jdbc/dbk105nlEJBDS	Parameter that defines the JNDI look up for the applicaton datasource	Don't modify the value
keystoreLocation	config/dls_cacerts	Parameter to set location for public/private key store for HTTPS posting and for XML Digital Signature	The keystore location should not be changed and keystore should be available in this location
keystorePassword	changeit	The keystore password	The deafult password is "changeit". Modify the property in case the password is different for the keystore.
sscroKey Alias	ofss_routeone	OFSLL's private key alias name. The private key is used to sign xml response to RouteOne	You may import the private key with alias "ofss_routeone" else modify the property value to alias used while importing the private key into keystore.
sscroKeyPassword	demotestSSCR1	OFSLL's private key password	demotestSSCR1 is a sample value
Configuration File: **Logging.properties**			
handlers	java.util.logging.FileHandler, java.util.logging.ConsoleHandler		
java.util.logging.FileHandler.level	ALL		

Property Name	Property Value	Description	Remarks
java.util.logging.FileHandler.pattern	/somewhere/logs/dbkls_xws_%g.log		
java.util.logging.FileHandler.limit	1000000		
java.util.logging.FileHandler.count	4		
java.util.logging.FileHandler.append	true		
java.util.logging.FileHandler.formatter	java.util.logging.SimpleFormatter		
java.util.logging.ConsoleHandler.level	WARNING		
java.util.logging.ConsoleHandler.formatter	java.util.logging.SimpleFormatter		
com.ofss.fll.xws.level	FINER	set the logging level for the application	Other Level values -- FINEST,FINE,CONFIG and INFO

EDOCS

Property Name	Property Value	Description	Remarks
Configuration File: **ds_servlet_init.conf**			
LOSeApplication-RequestServi-ceURL	http://<HOST NAME>:<PORT>/<CONTEXT ROOT>/LOSeAp-plicationRequest-Service	URL for OFSLL eDocs Create/ update applica-tion WebService	Refer the recommended context root table for CON-TEXT ROOT. Sample URL: http://host-name:port/dbkls-xws-app-ds/LOSeApplicationRe-questService
LOSApplication-CommentUpdate-ServiceURL	http://<HOST NAME>:<PORT>/<CONTEXT ROOT>/LOSAp-plicationCom-mentUpdateServi ce	URL for OFSLL eDocs comment update WebSer-vice	Refer the recommended context root table for CON-TEXT ROOT. Sample URL: http://host-name:port/dbkls-xws-app-ds/LOSApplicationCom-mentUpdateService
LOSApplication-LocationUpdate-ServiceURL	http://<HOST NAME>:<PORT>/<CONTEXT ROOT>/LOSAp-plicationLoca-tionUpdateServic e	URL for OFSLL eDocs location update WebSer-vice	Refer the recommended context root table for CON-TEXT ROOT. Sample URL: http://host-name:port/dbkls-xws-app-ds/LOSApplicationLoca-tionUpdateService
postTimeout	10	Parameter to set timeout value to post XML mes-sage to third party.	Timeout value should be specified in number of sec-onds.
useProxy	0	Parameter to switch on/off proxy use. By default use of proxy is switched off. To use proxy setup value to "1", else setup value as "0"	
proxyHost	abc.com	Parameter to set proxy host. This is required if useProxy is set to 1.	
proxyPort	80	Parameter to set proxy port. This is required if useProxy is set to 1.	

Property Name	Property Value	Description	Remarks
Configuration File: **dbkws_xae_init.conf**			
lenderId	SOME-LENDER_ID	The finance source identifier used in the dealer file	SOMELENDER_ID is a sample value
lenderName	SOME_LENDER_NAME	The finance source name used in the dealer file	SOME_LENDER_NAME is a sample value
applicationSource	EDOCS	Parameter to set application source	Don't modify this value
dealerElement-Name	DEALER_DE-TAILS	Name of XML element used for the dealer details used in the dealer file	DEALER_DETAILS is a sample value
fileLocation	/tmp		The directory for temporary files. Make sure that such directory exists on the deployment server
jndiLookupData-Source	jdbc/IN1Huk-WznG0b4esj	Parameter that defines the JNDI look up for the applicaton data-source	Don't modify the value
keystoreLocation	config/dls_cacerts	Parameter to set location for pub-lic/private key store for HTTPS posting and for XML Digital Sig-nature	The keystore location should not be changed and keystore should be availa-ble in this location
keystorePassword	changeit	The keystore password	The deafult password is "changeit". Modify the property in case the pass-word is different for the keystore.
postTimeout	10	Parameter to set timeout value to post XML mes-sage to third party.	Timeout value should be specified in number of sec-onds.

Property Name	Property Value	Description	Remarks
useProxy	0	Parameter to switch on/off proxy use. By default use of proxy is switched off. To use proxy setup value to "1", else setup value as "0"	
proxyHost	abc.com	Parameter to set proxy host. This is required if useProxy is set to 1.	
proxyPort	80	Parameter to set proxy port. This is required if useProxy is set to 1.	
Configuration File: **dbkws_xcl_init.conf**			
lenderId	SOME-LENDER_ID	The finance source identifier used in the dealer file	SOMELENDER_ID is a sample value
lenderName	SOME_LENDER_NAME	The finance source name used in the dealer file	SOME_LENDER_NAME is a sample value
dealerElement-Name	DEALER_DE-TAILS	Name of XML element used for the dealer details used in the dealer file	DEALER_DETAILS is a sample value
fileLocation	/tmp		The directory for temporary files. Make sure that such directory exists on the deployment server
fileDelimiter	,	Delimiter used in dealer header file	
jndiLookupData-Source	jdbc/dbk105nlE-JBDS	Parameter that defines the JNDI look up for the applicaton data-source	Don't modify the value

Property Name	Property Value	Description	Remarks
keystoreLocation	config/dls_cacerts	Parameter to set location for public/private key store for HTTPS posting and for XML Digital Signature	The keystore location should not be changed and keystore should be available in this location
keystorePassword	changeit	The keystore password	The deafult password is "changeit". Modify the property in case the password is different for the keystore.
sscroKeyAlias	ofss_routeone	OFSLL's private key alias name. The private key is used to sign xml response to RouteOne	You may import the private key with alias "ofss_routeone" else modify the property value to alias used while importing the private key into keystore.
sscroKeyPassword	demotestSSCR1	OFSLL's private key password	demotestSSCR1 is a sample value
Configuration File: **Logging.properties**			
handlers	java.util.logging.FileHandler, java.util.logging.ConsoleHandler		
java.util.logging.FileHandler.level	ALL		
java.util.logging.FileHandler.pattern	/somewhere/logs/dbkls_xws_%g.log		
java.util.logging.FileHandler.limit	1000000		
java.util.logging.FileHandler.count	4		
java.util.logging.FileHandler.append	true		
java.util.logging.FileHandler.formatter	java.util.logging.SimpleFormatter		

Property Name	Property Value	Description	Remarks
java.util.logging.ConsoleHandler.level	WARNING		
java.util.logging.ConsoleHandler.formatter	java.util.logging.SimpleFormatter		
com.ofss.fll.xws.level	FINER	set the logging level for the application	Other Level values -- FINEST,FINE,CONFIG and INFO

Webservices

Property Name	Property Value	Description	Remarks
Configuration File: **dbkws_lookups_init.conf**			
jndiLookupDataSource	jdbc/dbkwsDS	Parameter that defines the JNDI look up for the applicaton datasource	Don't modify the value
keystoreLocation	config/dls_cacerts	Parameter to set keystore location	The keystore location should not be changed and keystore should be available in this location
keystorePassword	changeit	The keystore password	The deafult password is "changeit". Modify the property in case the password is different for the keystore.
Configuration File: **dbkws_xae_init.conf**			
lenderId	SOMELENDER_ID	The finance source identifier used in the dealer file	SOMELENDER_ID is a sample value
lenderName	SOME_LENDER_NAME	The finance source name used in the dealer file	SOME_LENDER_NAME is a sample value
dealerElementName	DEALER_DETAILS	Name of XML element used for the dealer details used in the dealer file	DEALER_DETAILS is a sample value
fileLocation	/tmp		The directory for temporary files. Make sure that such directory exists on the deployment server
jndiLookupDataSource	jdbc/IN1HukWznG0b4esj	Parameter that defines the JNDI look up for the applicaton datasource	Don't modify the value

Property Name	Property Value	Description	Remarks
keystoreLocation	config/dls_cacerts	Parameter to set location for public/private key store for HTTPS posting and for XML Digital Signature	The keystore location should not be changed and keystore should be available in this location
keystorePassword	changeit	The keystore password	The deafult password is "changeit". Modify the property in case the password is different for the keystore.
sscroKeyAlias	ofss_routeone	OFSLL's private key alias name. The private key is used to sign xml response to RouteOne	You may import the private key with alias "ofss_routeone" else modify the property value to alias used while importing the private key into keystore.
sscroKeyPassword	demotestSSCR1	OFSLL's private key password	demotestSSCR1 is a sample value
postTimeout	10	Parameter to set timeout value to post XML message to third party.	Timeout value should be specified in number of seconds.
useProxy	0	Parameter to switch on/off proxy use. By default use of proxy is switched off. To use proxy setup value to "1", else setup value as "0"	
proxyHost	abc.com	Parameter to set proxy host. This is required if useProxy is set to 1.	
proxyPort	80	Parameter to set proxy port. This is required if useProxy is set to 1.	
Configuration File: **dbkws_xcl_init.conf**			
lenderId	SOMELENDER_ID	The finance source identifier used in the dealer file	SOMELENDER_ID is a sample value
lenderName	SOME_LENDER_NAME	The finance source name used in the dealer file	SOME_LENDER_NAME is a sample value
dealerElementName	DEALER_DETAILS	Name of XML element used for the dealer details used in the dealer file	DEALER_DETAILS is a sample value

Property Name	Property Value	Description	Remarks
fileLocation	/tmp		The directory for temporary files. Make sure that such directory exists on the deployment server
fileDelimiter	,	Delimiter used in dealer header file	
jndiLookup-DataSource	jdbc/dbk105nlE-JBDS	Parameter that defines the JNDI look up for the applicaton datasource	
keystoreLo-cation	config/dls_-cacerts	Parameter to set location for public/private key store for HTTPS posting and for XML Digital Sig-nature	The keystore location should not be changed and keystore should be availa-ble in this location
keystore-Password	changeit	The keystore password	The deafult password is "changeit". Modify the prop-erty in case the password is different for the keystore.
sscroKeyAlia s	ofss_route-one	OFSLL's private key alias name. The private key is used to sign xml response to RouteOne	You may import the private key with alias "ofss_route-one" else modify the prop-erty value to alias used while importing the private key into keystore.
sscroKey-Password	demotestSS-CR1	OFSLL's private key password	demotestSSCR1 is a sam-ple value
Configuration File: **dbkws_xcs_init.conf**			
jndiLookup-DataSource	jdbc/dbk-wsDS	Parameter that defines the JNDI look up for the applicaton datasource	Don't modify the value
keystoreLo-cation	config/dls_-cacerts	Parameter to set location for public/private key store for HTTPS posting and for XML Digital Sig-nature	The keystore location should not be changed and keystore should be availa-ble in this location
keystore-Password	changeit	The keystore password	The deafult password is "changeit". Modify the prop-erty in case the password is different for the keystore.
fileLocation	/tmp		The directory for temporary files. Make sure that such directory exists on the deployment server

Property Name	Property Value	Description	Remarks
application-Source	DBKTST	Application source	DBKTST is a sample value. Set it to appropriate value
Configuration File: **Logging properties**			
handlers	java.util.logging.FileHandler, java.util.logging.ConsoleHandler		
java.util.logging.FileHandler.level	ALL		
java.util.logging.FileHandler.pattern	/somewhere/logs/dbkls_xws_%g.log		
java.util.logging.FileHandler.limit	1000000		
java.util.logging.FileHandler.count	4		
java.util.logging.FileHandler.append	true		
java.util.logging.FileHandler.formatter	java.util.logging.SimpleFormatter		
java.util.logging.ConsoleHandler.level	WARNING		
java.util.logging.ConsoleHandler.formatter	java.util.logging.SimpleFormatter		
com.ofss.fll.xws.level	FINER	set the logging level for the application	Other Level values -- FINEST,FINE,CONFIG and INFO

Others

- Verify that latest XWS SQL Types, Views and Packages are installed.
- Required Java permissions have been granted

- Verify that System parameters for WebServices URLs as well as Post Response URLs are set. Please see the table below for details
- System parameter CMN_WALLET_PATH and CMN_WALLET_PASSWORD is setup
- Verify lenderId, lendername in config(dbkws_xae_init.conf) file and also set lenderId in all response xsl in element "A:TargetId".
- For DealerTrack access, user name/pwd should be setup in file ".htpasswd" under config dir

System Parameters to be configured:

System Parameter Name	System Parameter Desc
XWS_XAE_DLR_TRACK_RESP_PWD	DEALERTRACK APPLICATION RESPONSE BASIC AUTH PASSWORD
XWS_XAE_DLR_TRACK_RESP_URL	DEALERTRACK APPLICATION RESPONSE URL
XWS_XAE_DLR_TRACK_RESP_USER	DEALERTRACK APPLICATION RESPONSE BASIC AUTH USER
XWS_XAE_DLR_TRACK_WS_URL	OFSLL WEBSERVICE URL TO POST APPLICATION RESPONSE TO DEALER-TRACK
XWS_XAE_ROUTEONE_RESP_PWD	ROUTE ONE APPLICATION RESPONSE BASIC AUTH PASSWORD
XWS_XAE_ROUTEONE_RESP_URL	ROUTE ONE APPLICATION RESPONSE URL
XWS_XAE_ROUTEONE_RESP_USER	ROUTE ONE APPLICATION RESPONSE BASIC AUTH USER
XWS_XAE_ROUTEONE_WS_URL	OFSLL WEBSERVICE URL TO POST APPLICATION RESPONSE TO ROUTE ONE
XWS_XAE_EDOC_RESP_URL	EDOCS APPLICATION RESPONSE BASIC AUTH PASSWORD
XWS_XAE_EDOC_WS_URL	EDOCS APPLICATION RESPONSE URL
XWS_XAE_EDOC_RESP_USER	EDOCS APPLICATION RESPONSE BASIC AUTH USER
XWS_XAE_EDOC_RESP_PWD	OFSLL WEBSERVICE URL TO POST APPLICATION RESPONSE TO EDOCS
XWS_XPR_DLR_TRACK_RESP_PWD	DEALER TRACK DEALER LOAD PASS-WORD
XWS_XPR_DLR_TRACK_RESP_URL	DEALER TRACK DEALER LOAD RESPONSE URL
XWS_XPR_DLR_TRACK_RESP_USER	DEALER TRACK DEALER LOAD USER ID
XWS_XPR_DLR_TRACK_WS_URL	OFSLL WEBSERVICE URL TO POST DEALER DETAILS TO DEALER TRACK

System Parameter Name	System Parameter Desc
XWS_XPR_INCLUDE_TEMP	INCLUDE TEMP PRODUCERS
XWS_XPR_ROUTEONE_RESP_PWD	ROUTE ONE DEALER LOAD PASS-WORD
XWS_XPR_ROUTEONE_RESP_URL	ROUTE ONE DEALER LOAD RESPONSE URL
XWS_XPR_ROUTEONE_RESP_USER	ROUTE ONE DEALER LOAD USER ID
XWS_XPR_ROUTEONE_WS_URL	OFSLL WEBSERVICE URL TO POST DEALER DETAILS TO ROUTE ONE
XWS_XAE_DLR_TRACK_LENDERID	DEALERTRACK LENDER ID
XWS_XAE_ROUTEONE_LENDERID	ROUTEONE LENDER ID
XWS_XAE_DL-R_TRACK_LENDER_NAME	DEALERTRACK LENDER NAME
XWS_XAE_ROUTE-ONE_LENDER_NAME	ROUTEONE LENDER NAME
XWS_XAE_ECON_ROU-TEEONE_RESP_URL	Route One E-contract response URL