Oracle® Financial Services Lending and Leasing

WebServices Installation Guide

Release 14.12.0.0.0
F82268-01
August 2024

Oracle Financial Services Lending and Leasing WebServices Installation Guide, Release 14.12.0.0.0

F82268-01

Contents

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.

- Install WebServices Database Objects
- Deploy Application Interface WebServices
- Configure Weblogic Policy on WebServices
- Verify Successful Installation
- Enable Logging
- Configure RESTful WebService
- Appendix A :Configuration parameters

This section consists of the following topics:

- Prerequisites
- Audience
- Documentation Accessibility
- Diversity and Inclusion
- Conventions

Prerequisites

1. Download and Install the Oracle Fusion Middleware 12c Version 12.2.1.4.0 (Fusion Middleware Infrastructure installer) from http://www.oracle.com/technetwork/middleware/fusion-middleware/downloads/index.html#close. They are also 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.

Audience

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

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Conventions

The following conventions are used in this document:

Table 1 Conventions used

Term	Refers to
Application	Oracle Financial Services Lending and Leasing

1

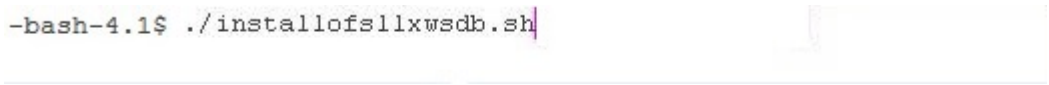
Install WebServices Database Objects

The following section deals with installation of WebServices Database Objects.

Download and unzip the WebServices database (`ofsllxwsdb.zip`) to a staging folder.

Run $ ./installofsllxwsdb.sh

Figure 1-1 Command prompt window 1



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

Figure 1-2 Command prompt window 2

Enter **y** when prompted to continue.

Figure 1-3 Command prompt window 3

Table 1-1 Script Prompts

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.

Figure 1-4 Command prompt window 4

```
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
```

Table 1-2 Script Prompts

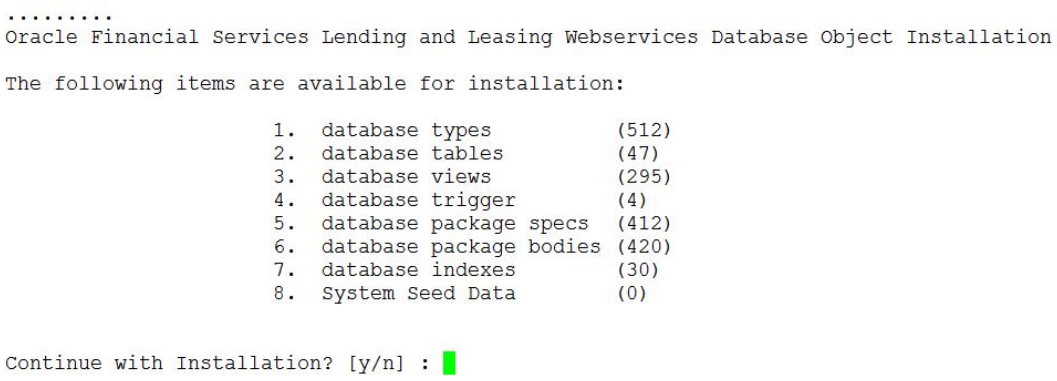
Script Prompts	Description and Action Required
Oracle DB Home Path	Enter the path to the Oracle DB home directory. This is referred to as `$ORACLE_HOME`. Enter **y** when prompted for.
Oracle SID	Enter the Name of Oracle Instance. Enter **y** when prompted for.

Figure 1-5 Command prompt window 5

```
   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
```

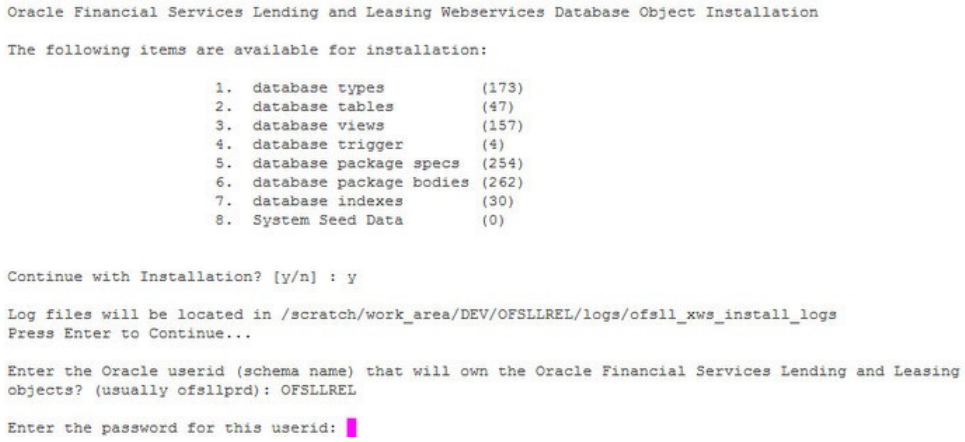
The script lists and sets the CRITICAL environment variables.

Figure 1-6 Command prompt window 6



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

Figure 1-7 Command prompt window 7



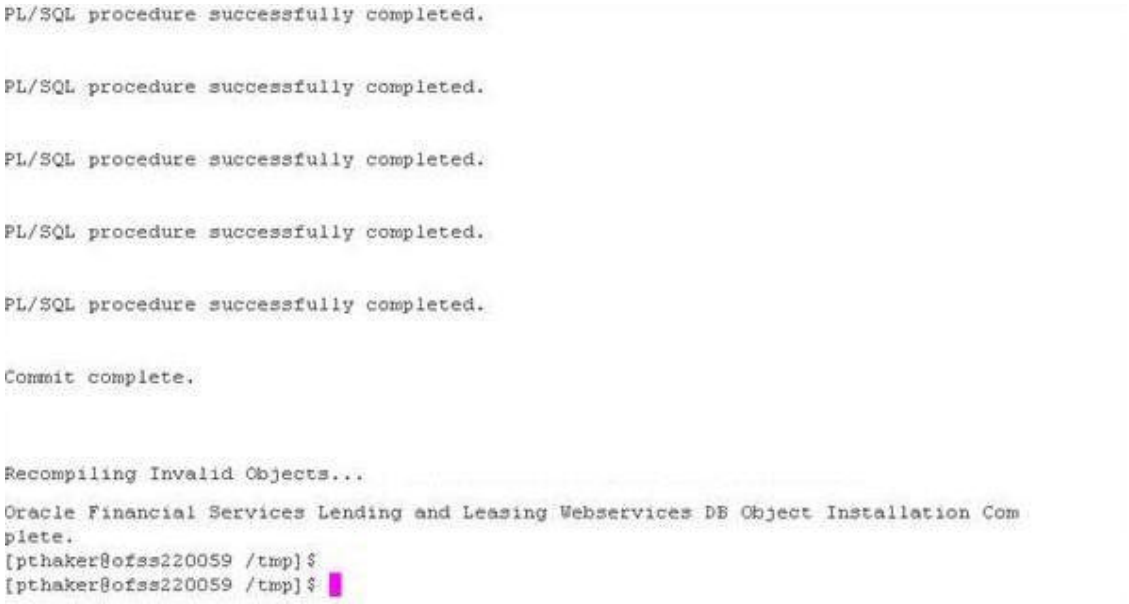
Sets the path for the location of log files. Press **Enter** to continue.

Table 1-3 Script Prompts

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.

Figure 1-8 Command prompt window 8



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

2

Deploy Application Interface WebServices

The following section details the steps to deploy application interface WebServices.

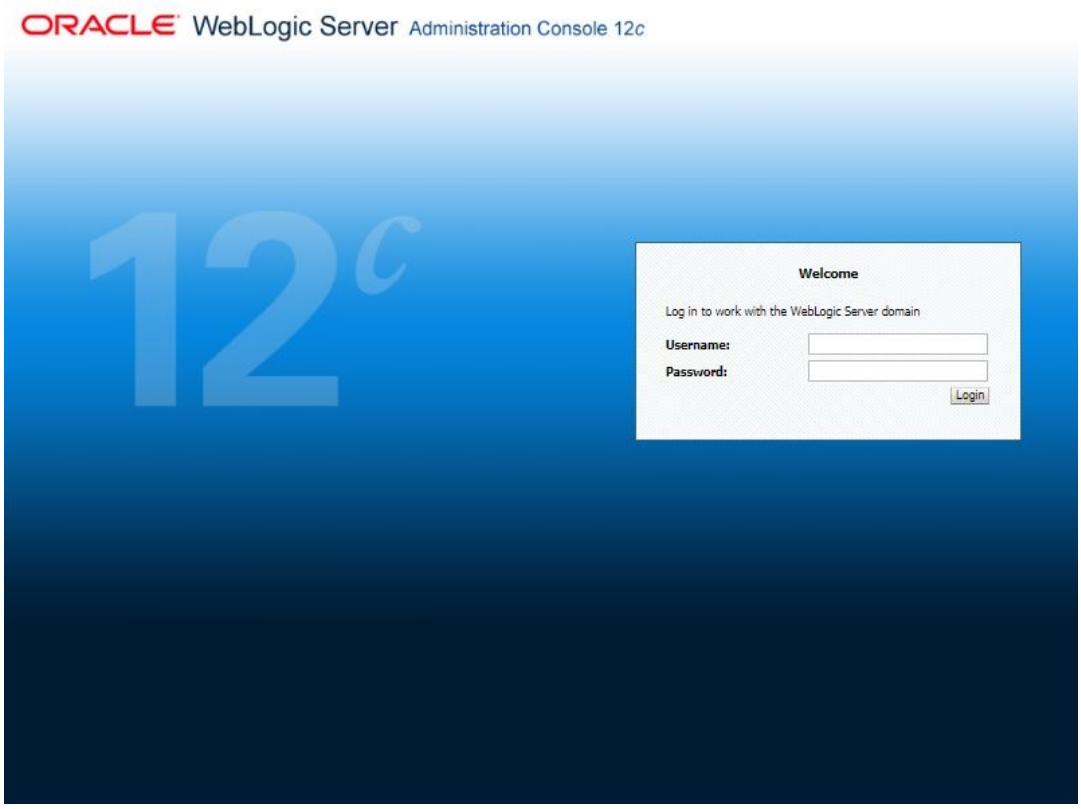
- Creating Data Sources for WebServices
- Work with SSL
- Creating RouteOne Credentials and System Policies
- Deploy Webservices

2.1 Creating Data Sources for WebServices

It is assumed that a managed server is already created for the following web service deployment(s) as per the process detailed in Application Installation guide section - **3.2 Creating Domain and Servers**.

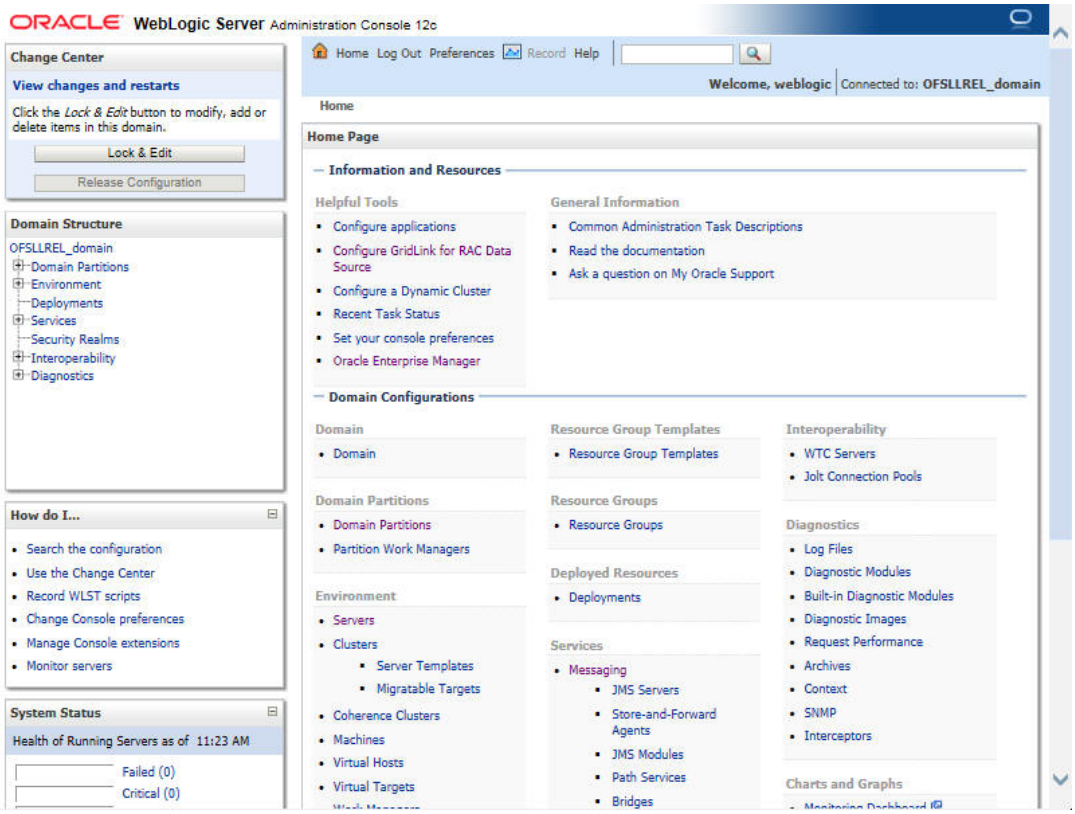
1. Login to WebLogic Server 12c console (`http://hostname:port/console`).

Figure 2-1 Creating Data sources 1



2. The following window is displayed.

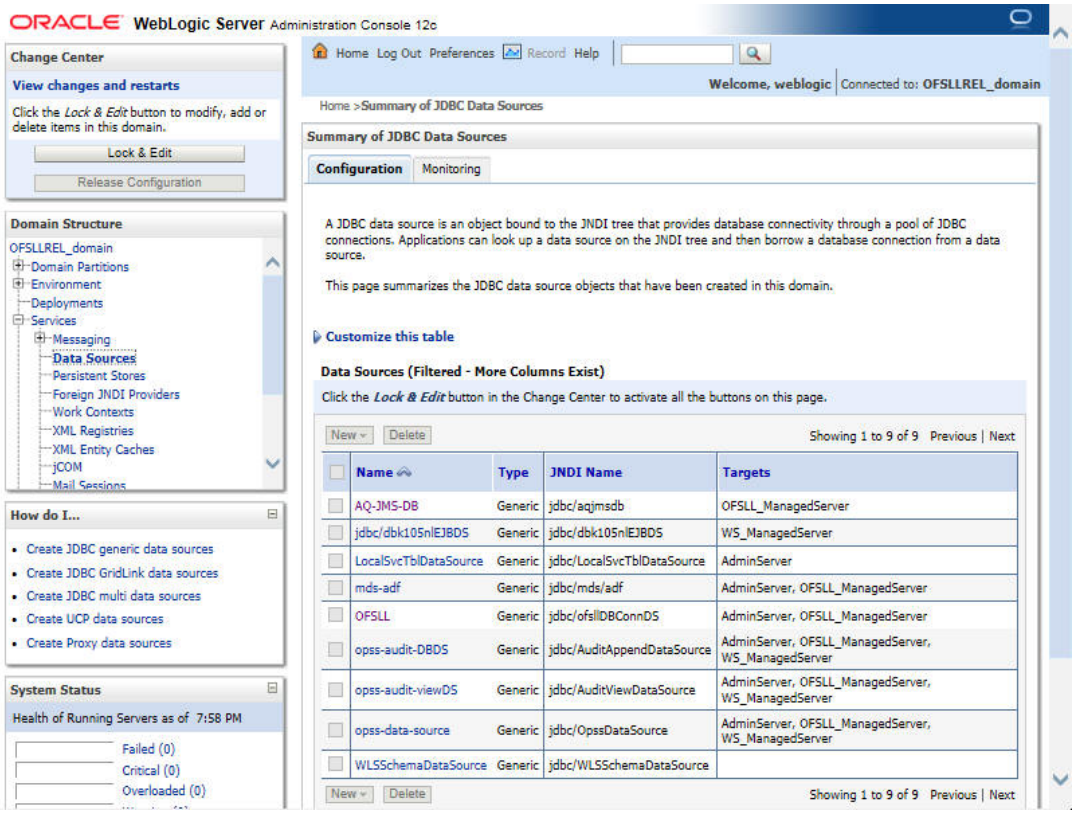
Figure 2-2 Creating Data sources 2



3. Click Domain Name > Services > Data Sources.

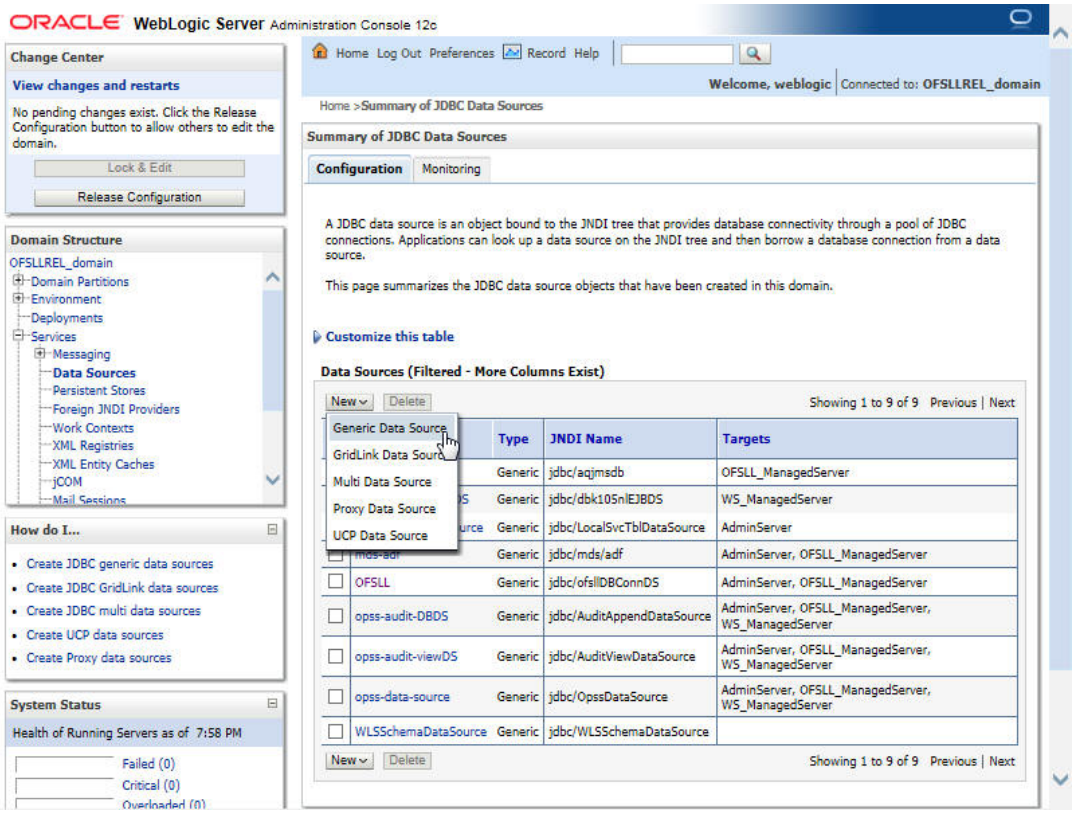
   The following window is displayed.

Figure 2-3 Creating Data sources 3



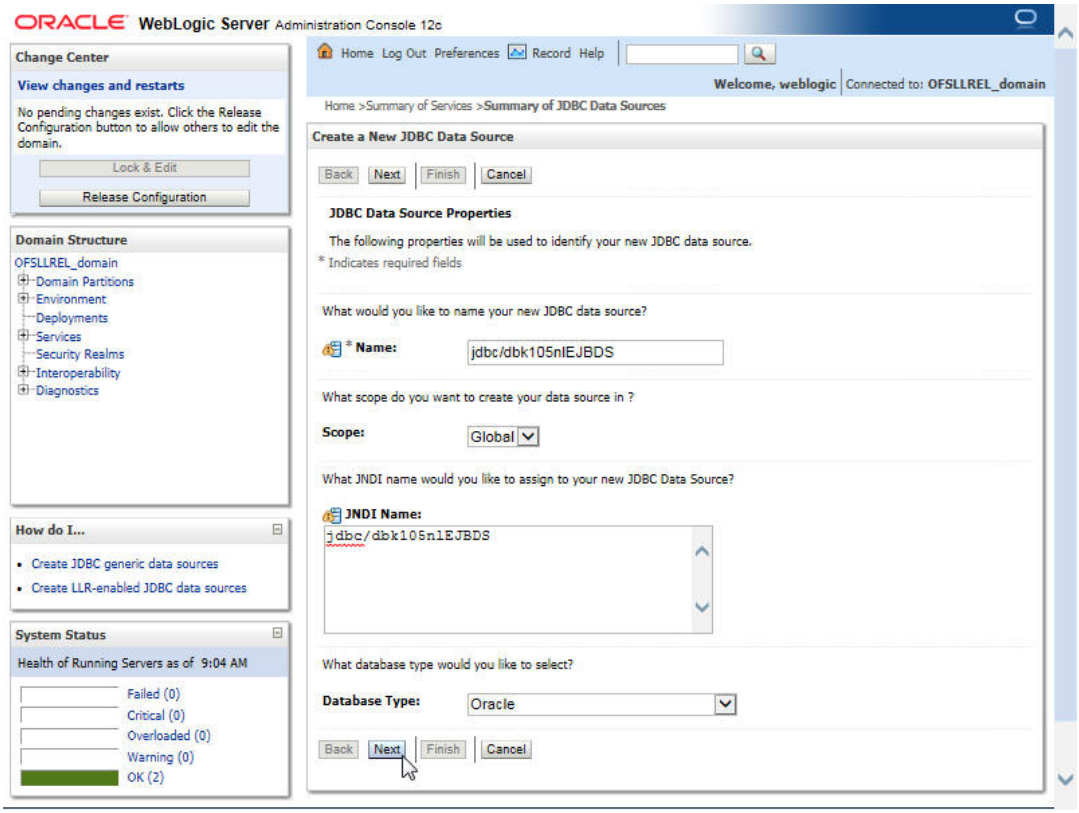
4. Click **Lock & Edit** button on the left panel. Click **New** on right panel and select **Generic Data Source**.

Figure 2-4 Creating Data sources 4



5. Specify the following details:
   - Enter Data source Name.
   - Enter JNDI Name as **jdbc/dbk105nlEJBDS**.
   - Select **Oracle** as Database Type.

Figure 2-5 Creating Data sources 5

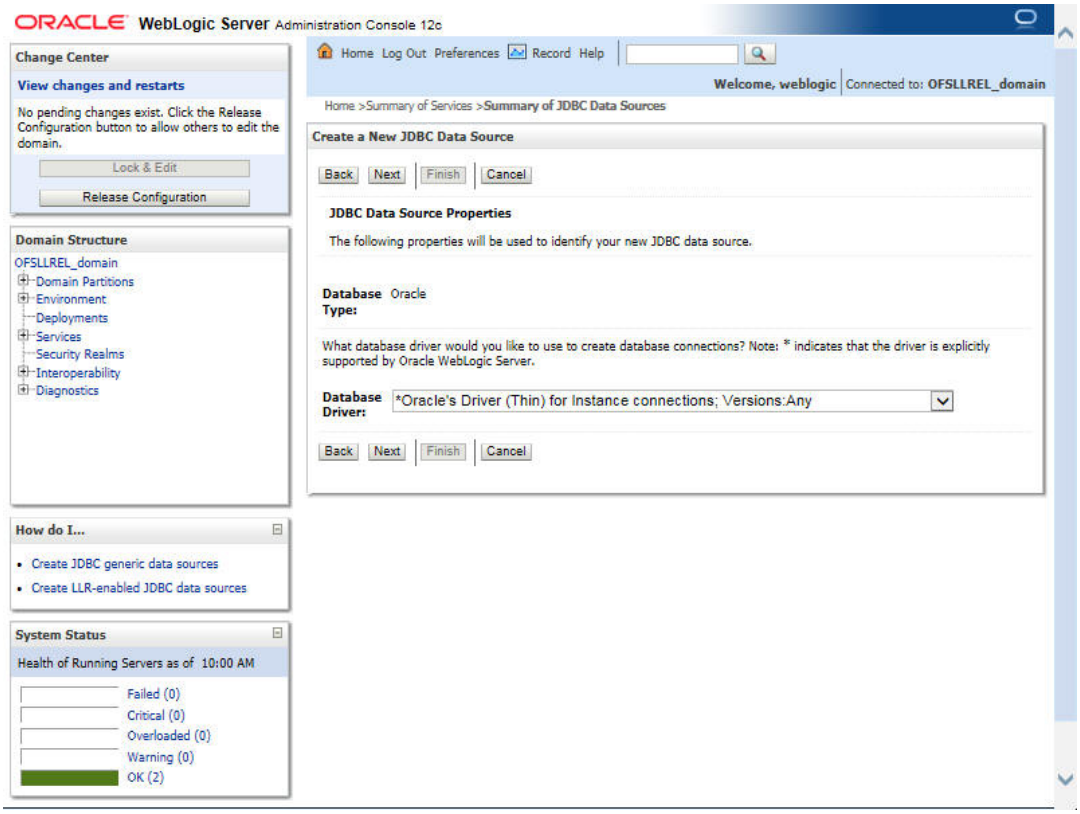


6. Click **Next**.

   The following window is displayed.

7. Select the Database Driver **Oracle's Driver(Thin) for Instance connections;Versions:Any** as shown.

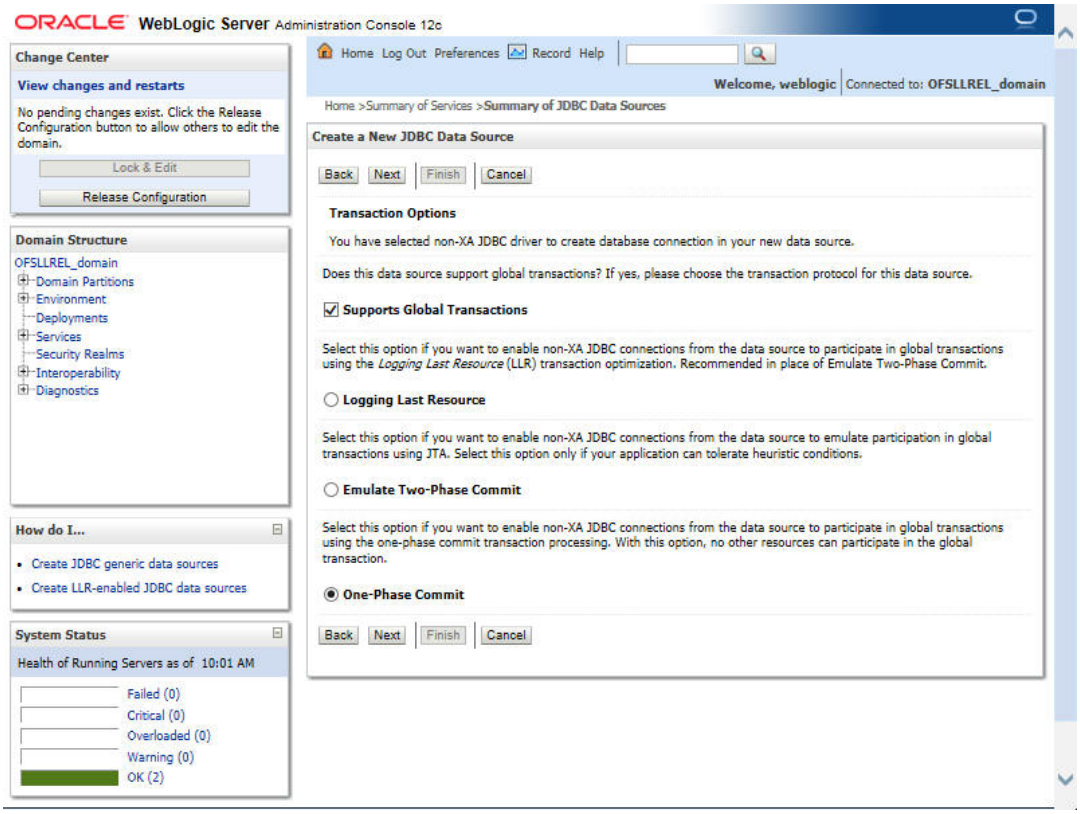
Figure 2-6 Creating Data sources 6



8. Click ‘Next’.

   The following window is displayed.

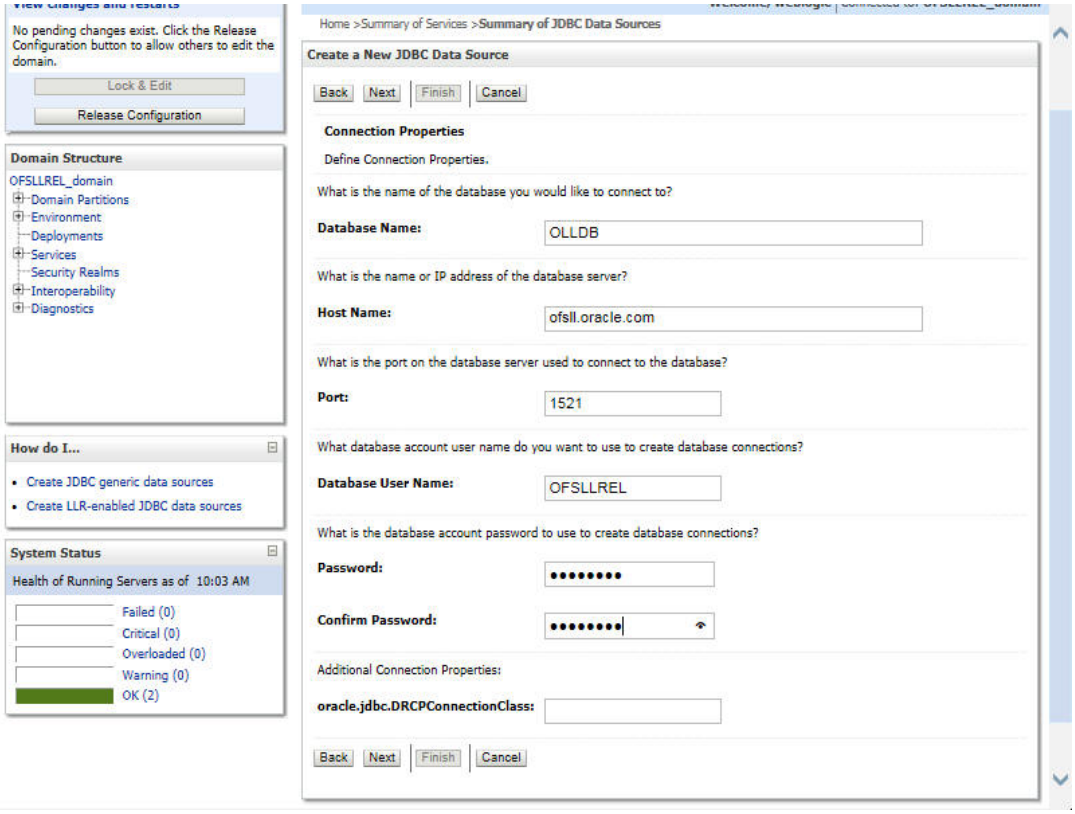
Figure 2-7 Creating Data sources 7



9. Click **Next**.

   The following window is displayed.

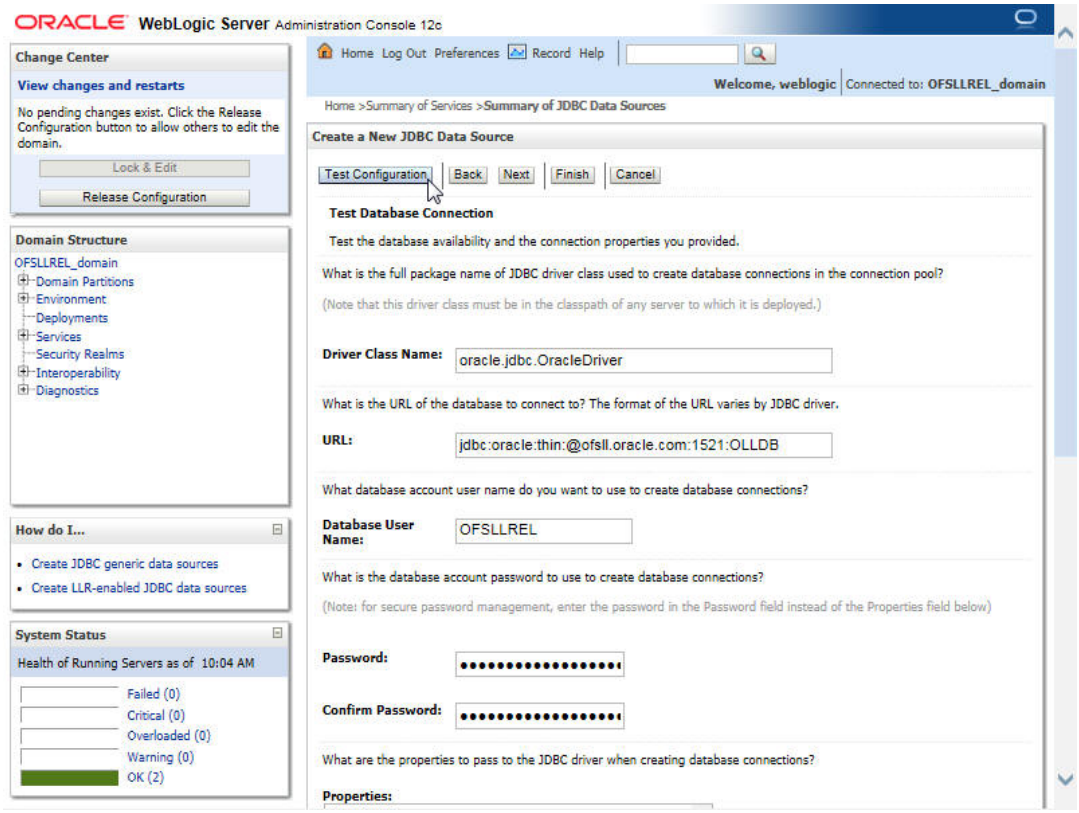
Figure 2-8 Creating Data sources 8



10. Enter the Database details.
11. Click **Next**. T

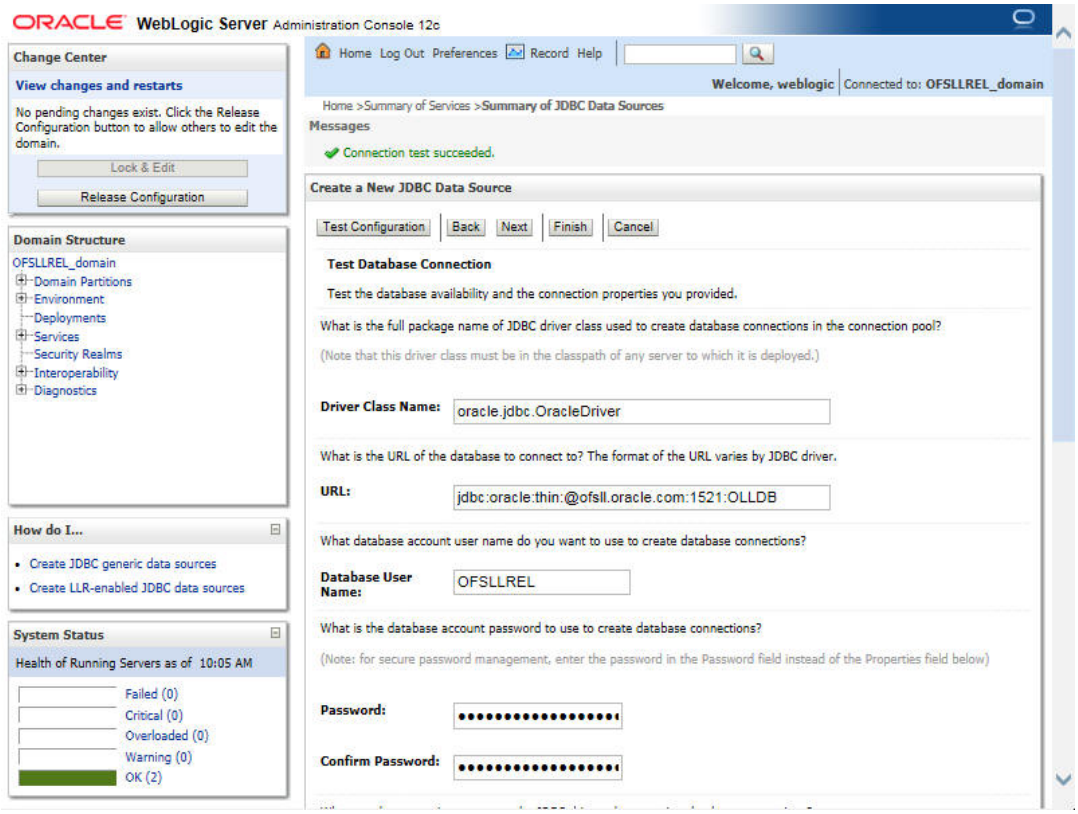
    he following window is displayed.

Figure 2-9 Creating Data sources 9



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

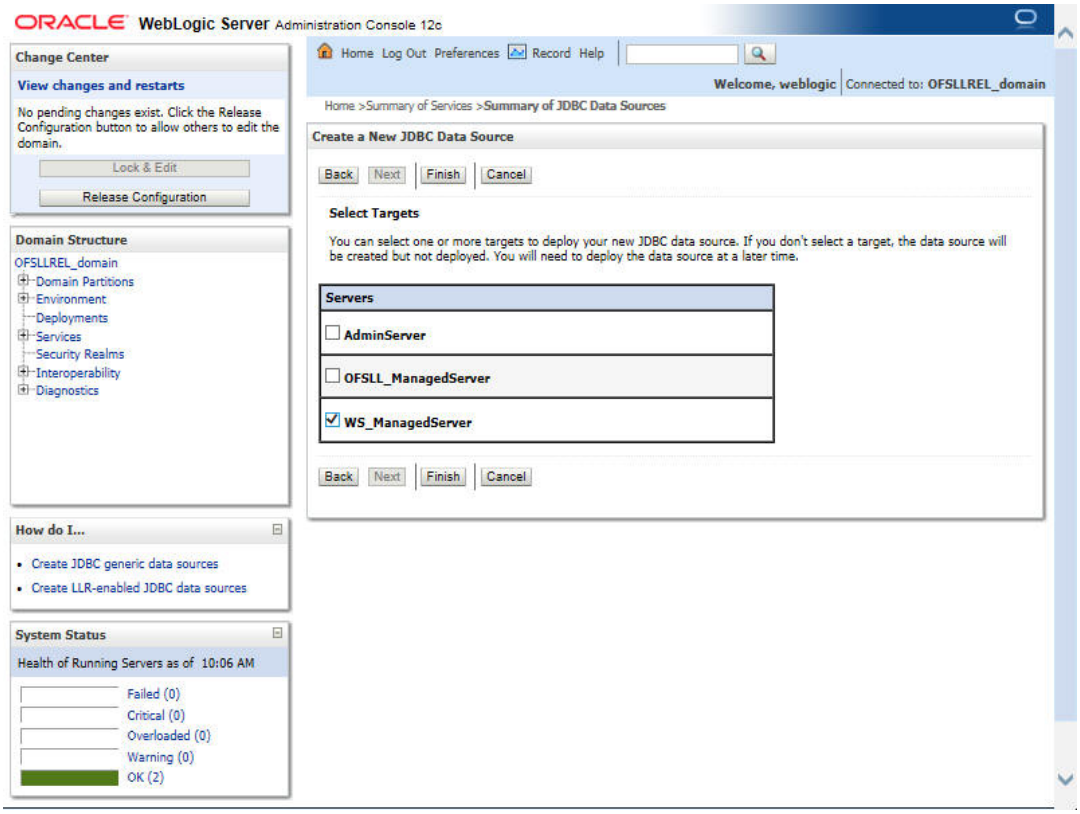
Figure 2-10 Creating Data sources 10



13. Click **Next**.

    The following window is displayed.

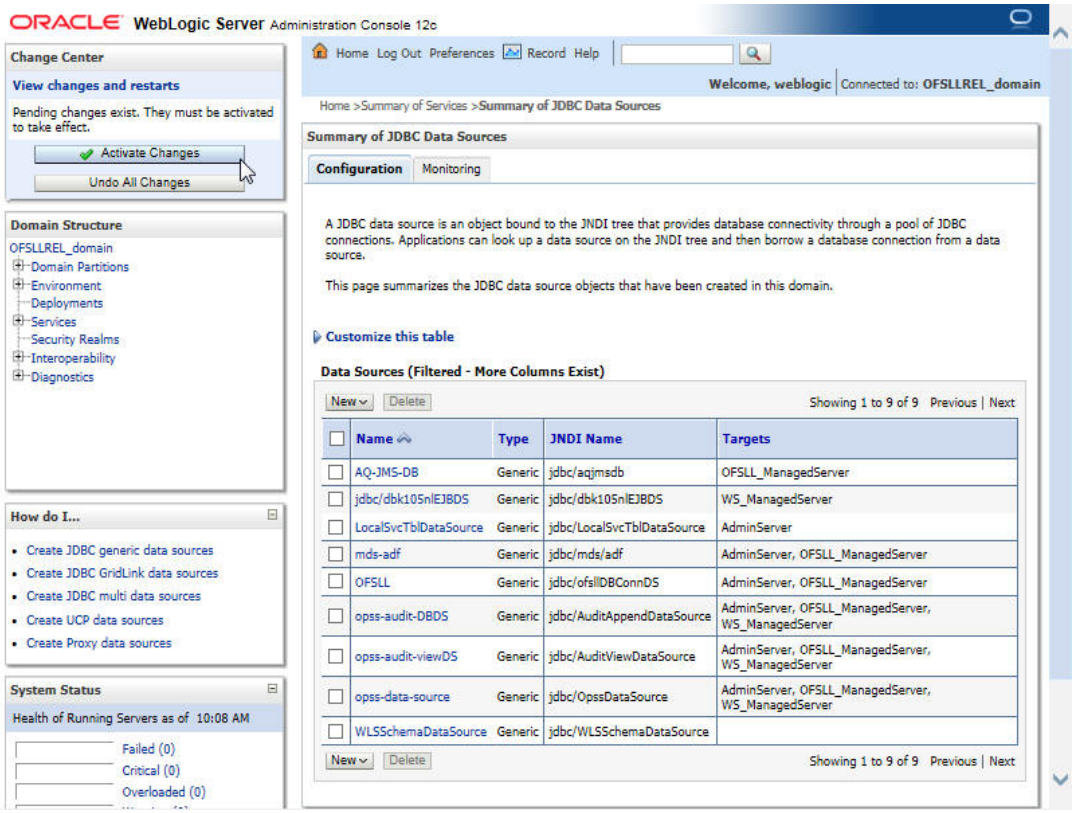
Figure 2-11 Creating Data sources 11



14. Select **WS_ManagedServer** as the target Web service Server and click **Finish**.

    The following window is displayed.

Figure 2-12 Creating Data sources 12

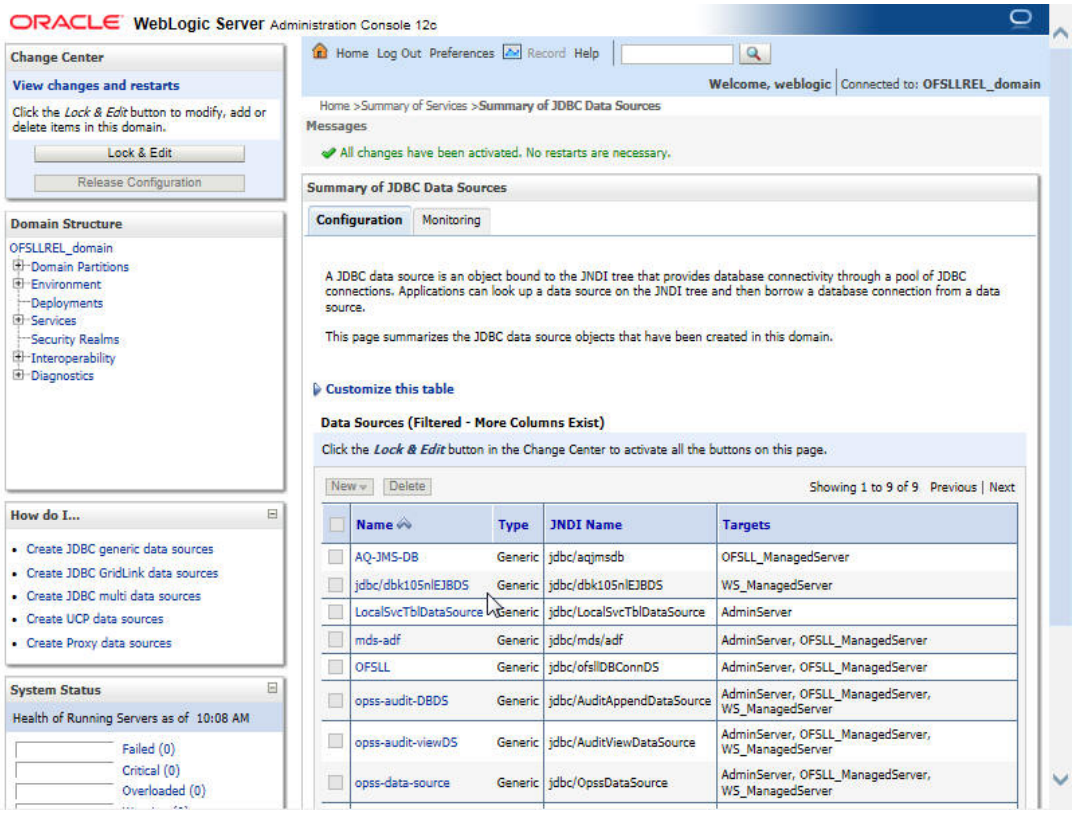


15. Click **Activate Changes**.

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

    - jdbc/dbkwsDS
    - jdbc/IN1HukWznG0b4esj

Figure 2-13 Creating Data sources 13



2.2 Work 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.

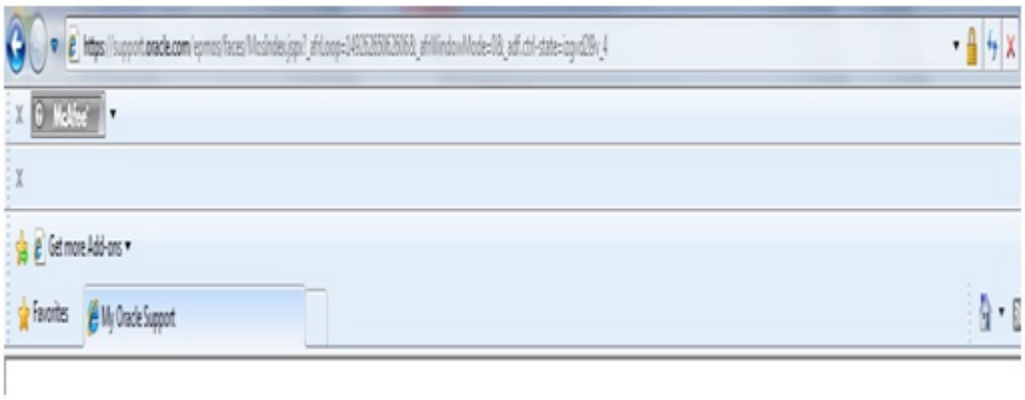
This topic consists of the following sections:

2.2.1 Importing Certificates to keystore

The following section details the steps to be followed to import certificates to keystore.

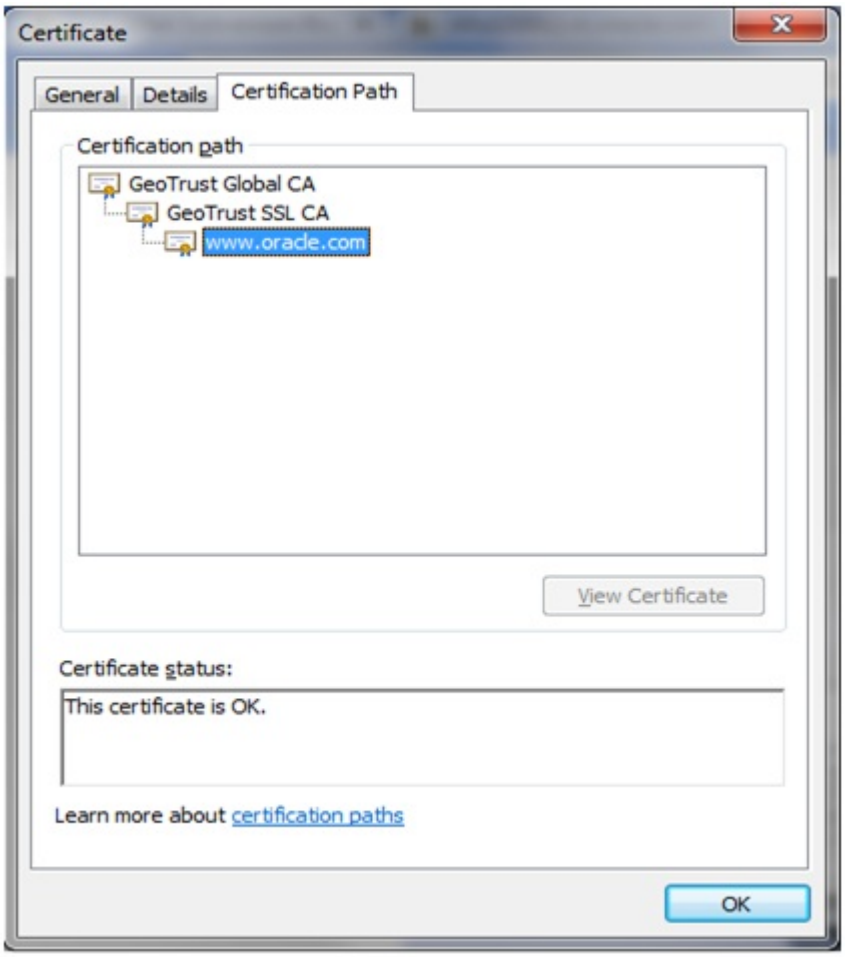
1. The JKS(dls_cacerts) should be available under /WEB-INF/classes/config.
2. Save all the certificates from the vendor website. Note to save the certificates in "Base-64 encoded X.509(.CER)" and with extn .cer. Steps for saving certificates from the vendor website:
   - Click on the padlock and choose **View Certificates**.

Figure 2-14 Importing certificates to keystore 1



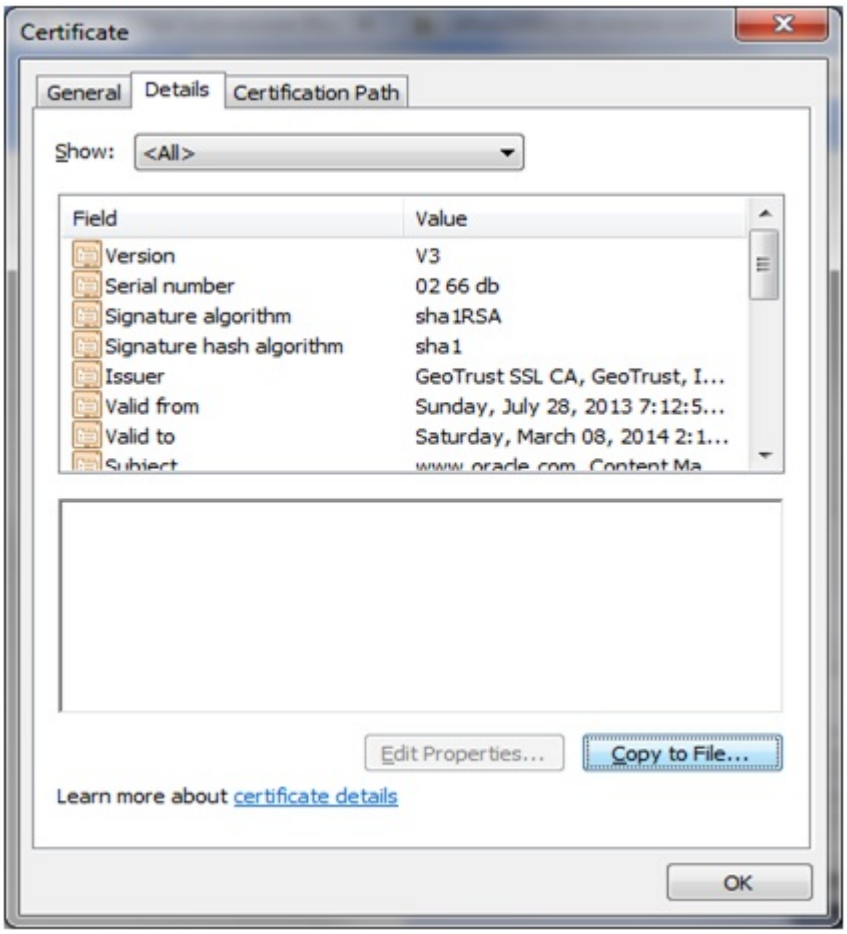
   - Click on the Certification Path tab and select the certificate with the 'Name' and 'Issued To' the same as the user-defined ID.

Figure 2-15 Importing certificates to keystore 2



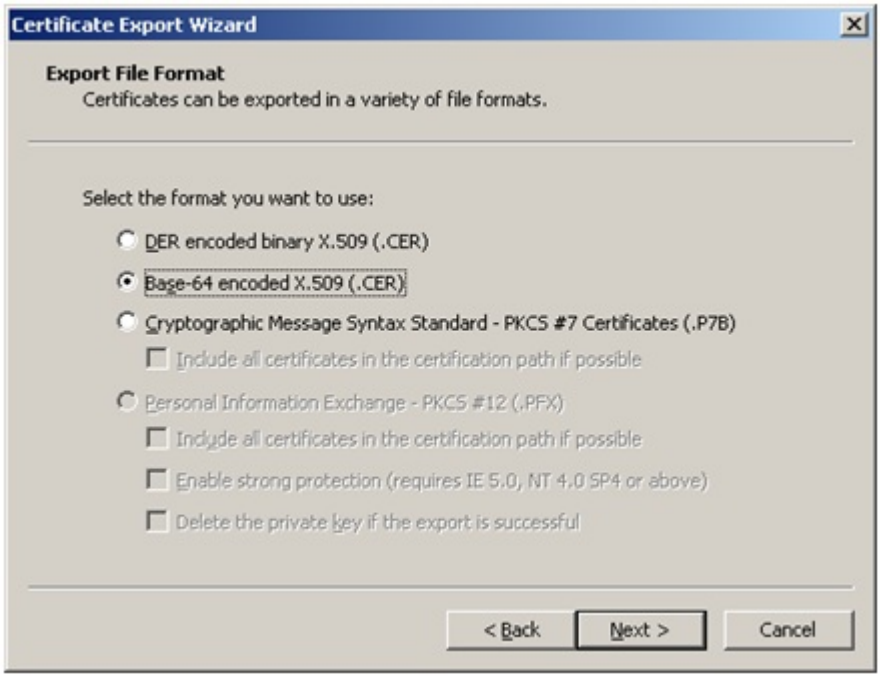
- Go to the Details tab.

Figure 2-16 Importing certificates to keystore 3



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

Figure 2-17 Importing certificates to keystore 4



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

2.2.2 Enable SSL Debugging

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

2.2.3 Connect 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.

2.3 Creating RouteOne Credentials and System Policies

This sections explains how to create RouteOne credentials and system policies.

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

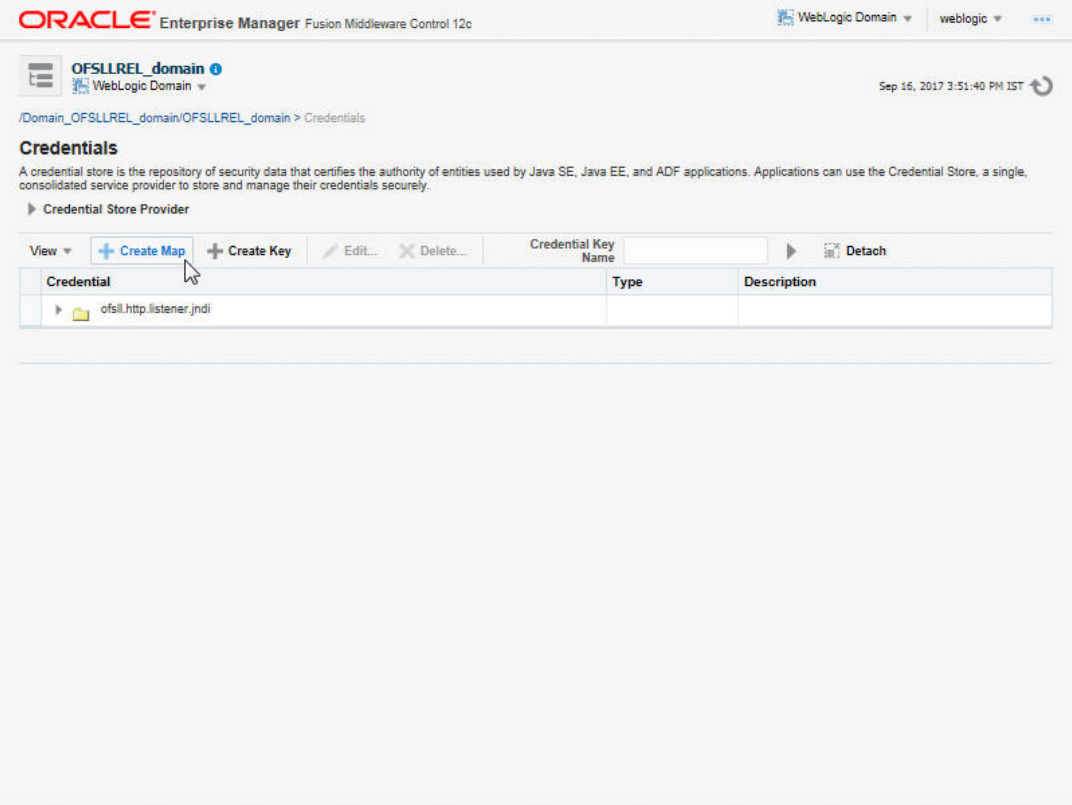
1. Login to Oracle Enterprise Manager 12c (`http://hostname:port/em`).

   (Optional) Enter the result of the step here.

2. On the left panel, right click on OFSLLREL_domain and select Security > System Policies > Credentials.

   The following window is displayed.

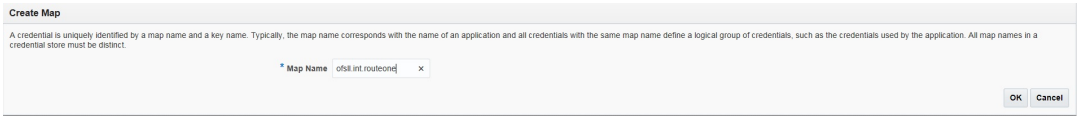
Figure 2-18 RouteOne credentials 1



3. Click **Create Map**.

   The following window is displayed.

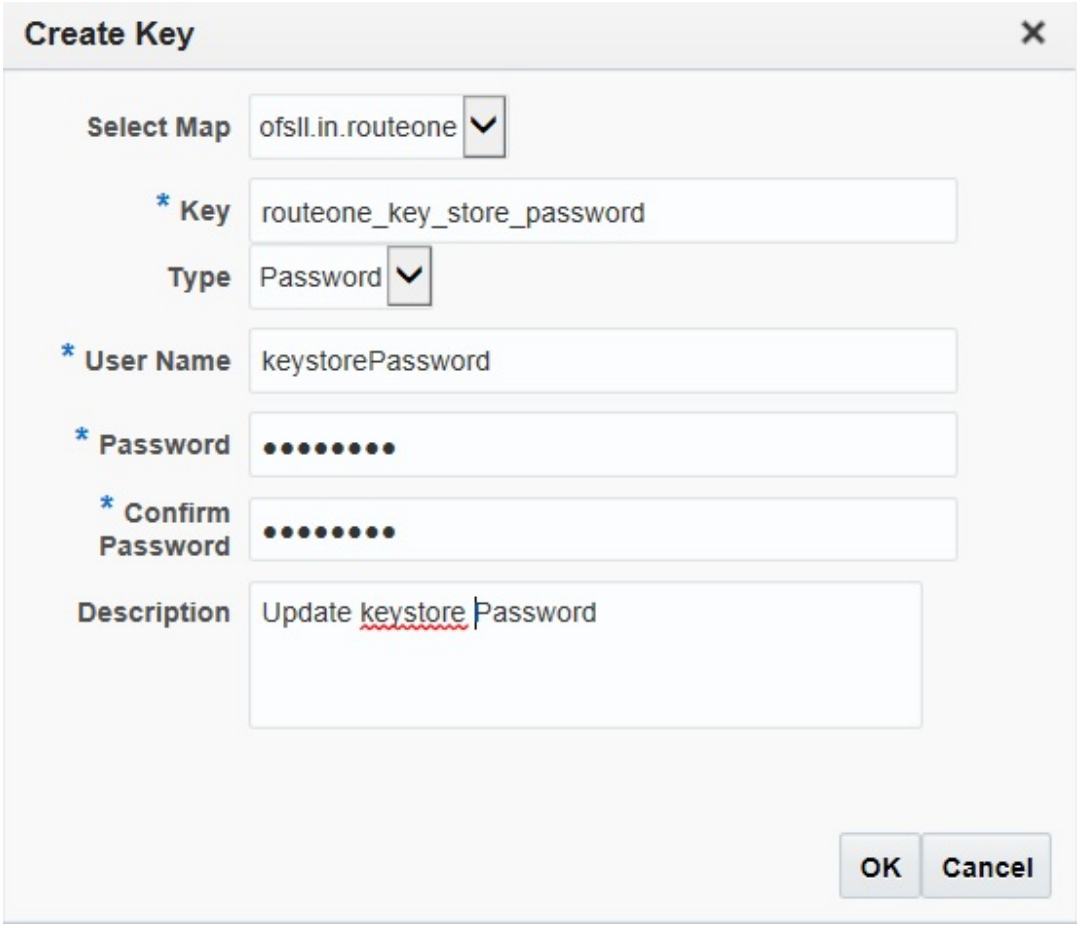
Figure 2-19 RouteOne credentials 2



4. Enter Map Name as `ofsll.int.routeone` and click **OK**.
5. Click **Create Key**.

   The following window is displayed.

Figure 2-20 RouteOne credentials 3



6. Click **OK**.

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

Table 2-1 Maps and corresponding keys

Maps	Key Name	Username	Password	Description
ofsll.int.routeone	routeone_key_store_pd	keystorePassword	setme	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_pd	sscroKeyAlias	ssc_routeone	OFSLL's private key alias name. The private key is used to sign xml response to RouteOne
	routeone_sscro_key_passsword	sscroKeyPassword	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** topic available in Application Installation Guide.

Table 2-2 Permission Class

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

2.4 Deploy Webservices

The following section details the steps to be followed for the deployment of WebServices in Enterprise Manager.

- Deploying WebServices in Enterprise Manager

2.4.1 Deploying WebServices in Enterprise Manager

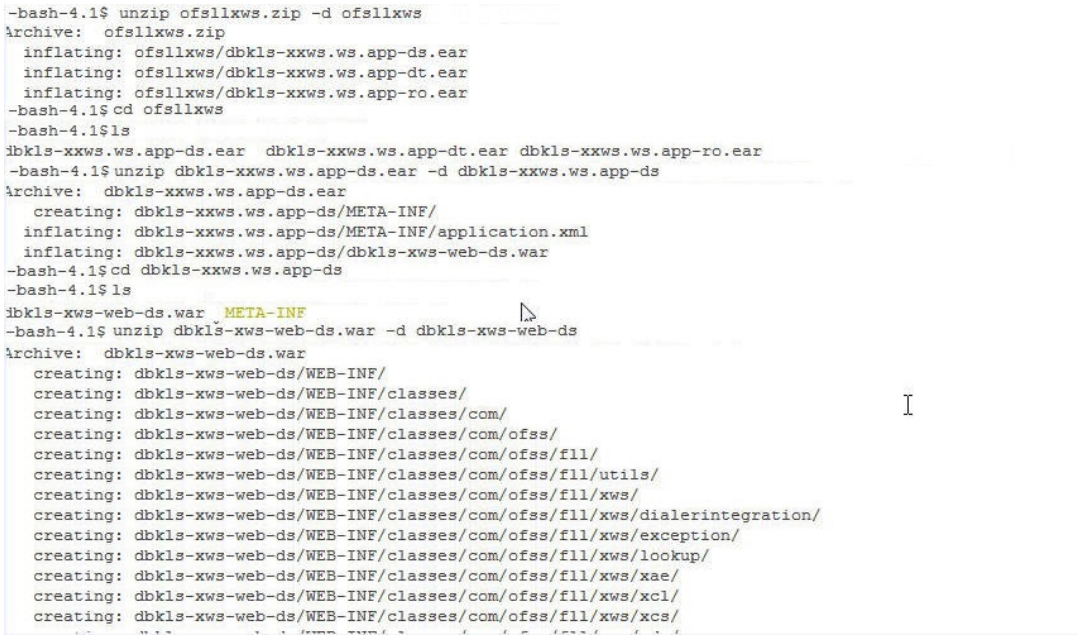
1. Download and unzip the WebServices - `ofsllxws.zip`.
2. Following is a mapping of which EAR is needed for which service:

Table 2-3 Mapping of EAR

dbkls-xxws.ws.app-dt.ear	dbkls-xxws.ws.app-ds.ear	dbkls-xxws.ws.app-ro.ear
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
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
DT Interface: To receive comments from dealer track	DS Interface: To receive edocs location update	RO Interface: To receive comments from route one
LOSPostStatusRequestService: To post comments to dealer track		LOSPostStatusRequestService: To post comments to route one
LOSPostStatusRequestService: To post application status to dealer track		LOSPostStatusRequestService: To post application status to route one
ILOSPostDealerDetailsService: To post dealer details to dealer track		ILOSPostDealerDetailsService: To post dealer details to dealer track
		LOSEContractService: To recieve contract information from RouteOne

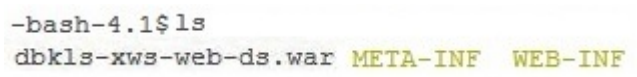
3. Unzip all the ear files present:
   - dbkls-xxws.ws.app-dt.ear
   - dbkls-xxws.ws.app-ro.ear
   - dbkls-xxws.ws.app-ds.ear
4. Open each of the unzipped files and unzip the war file.

Figure 2-21 Deployment in EnterpriseManager 1



5. It creates a WEB-INF file.

Figure 2-22 Deployment in EnterpriseManager 2



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

- / dbkls-xxws.ws.app-ds / dbkls-xxws.ws.app-dt / dbklsxxws.ws.app-ro
  WEB-INF (directory)

  ---->classes (directory)

  - config

  - class files (in package folders)

  -----> lib (directory)

  - OfsllCommonCSF.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 **Configuration parameters**section in Appendix chapter.

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

Figure 2-23 Deployment in EnterpriseManager 3

```
#### 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

Figure 2-24 Deployment in EnterpriseManager 4

```
#### 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
```

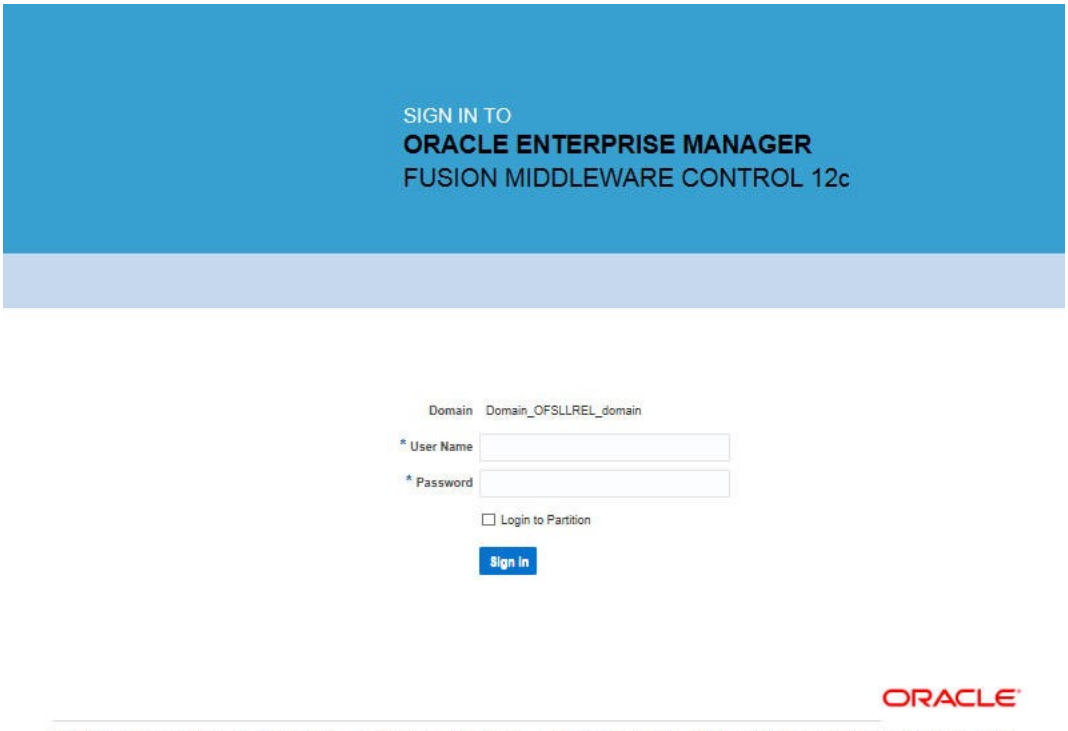
   - ro_servlet_init.conf- This is the configuration file only for route one servlet interface

Figure 2-25 Deployment in EnterpriseManager 5

```
#### 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`).

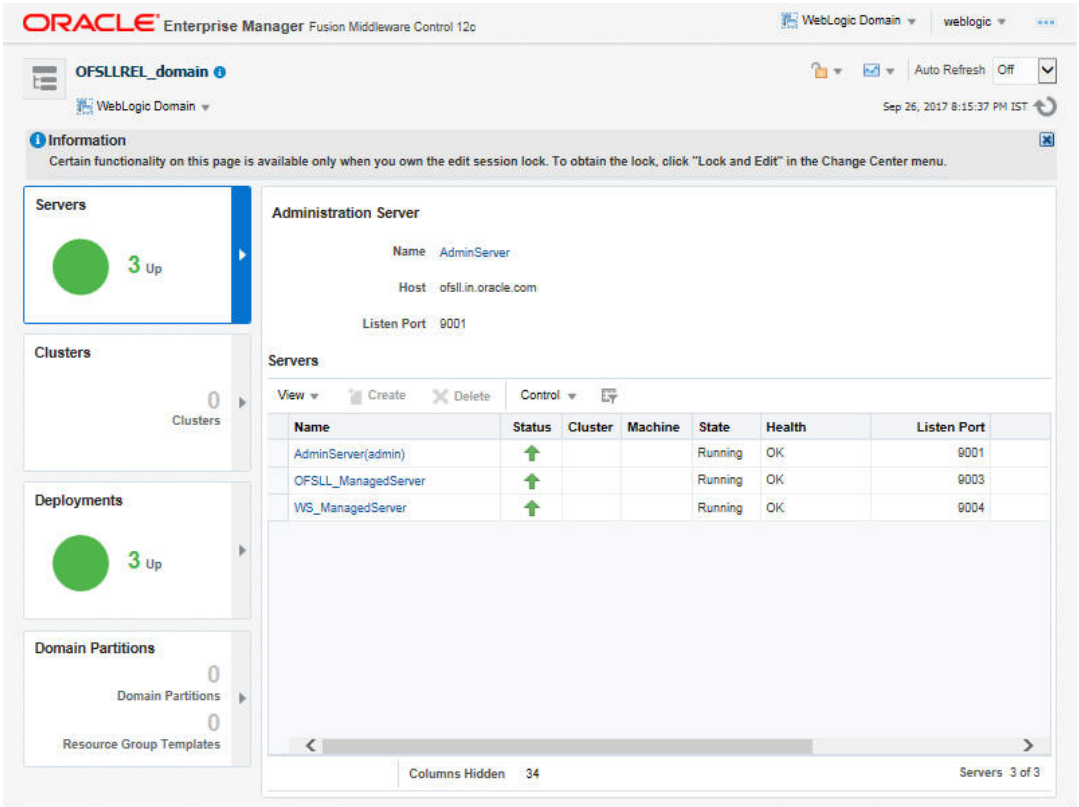
Figure 2-26 Deployment in EnterpriseManager 6



8. Enter valid login credentials.

   The following window is displayed.

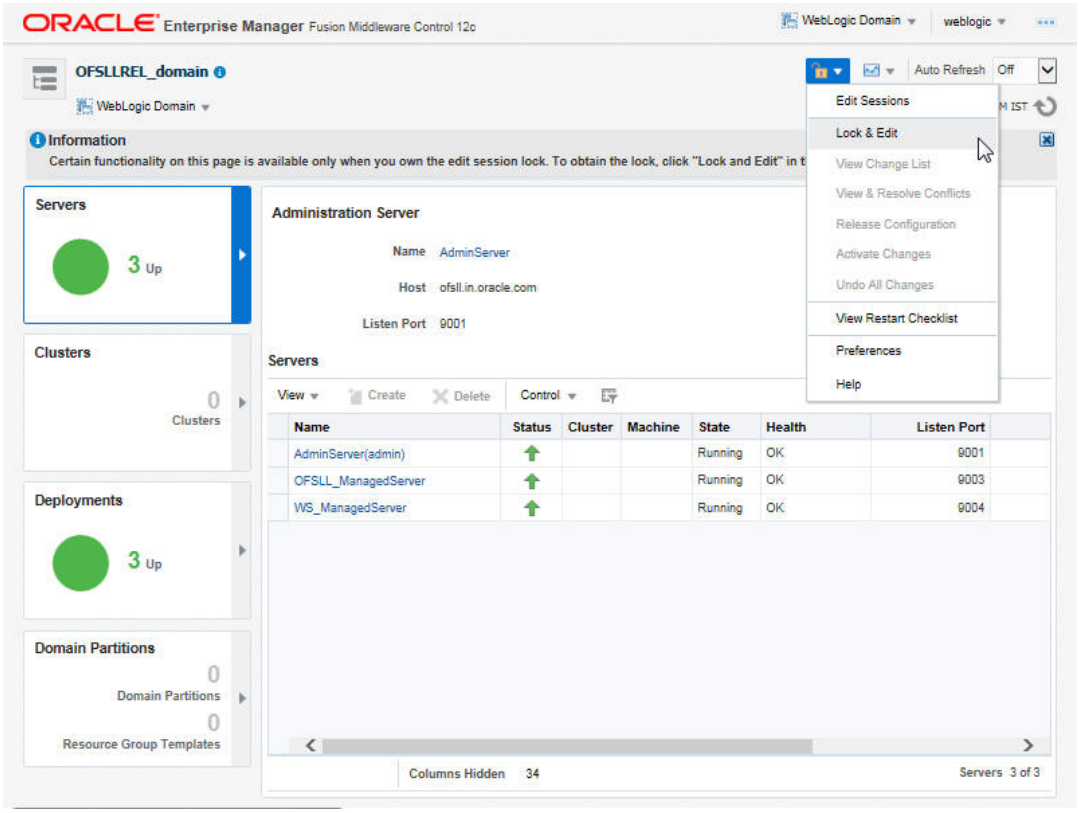
Figure 2-27 Deployment in EnterpriseManager 7



9. Expand the weblogic domain present in the left pane.

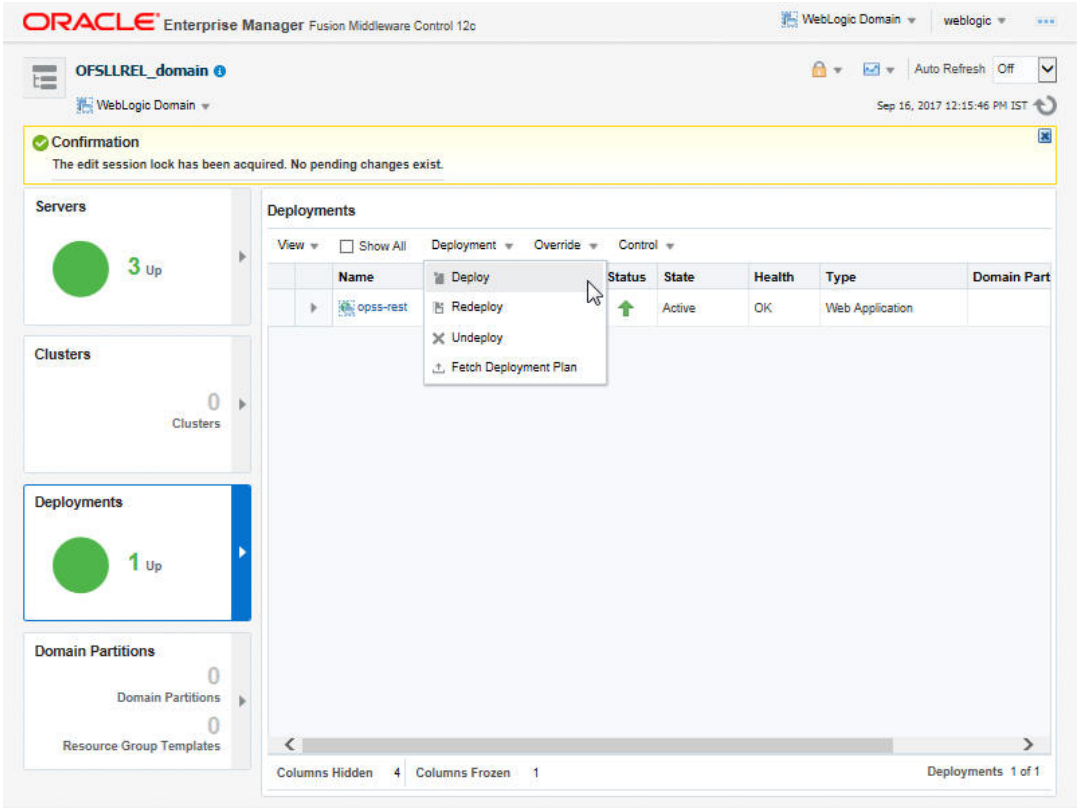
   The following window is displayed.

Figure 2-28 Deployment in EnterpriseManager 8



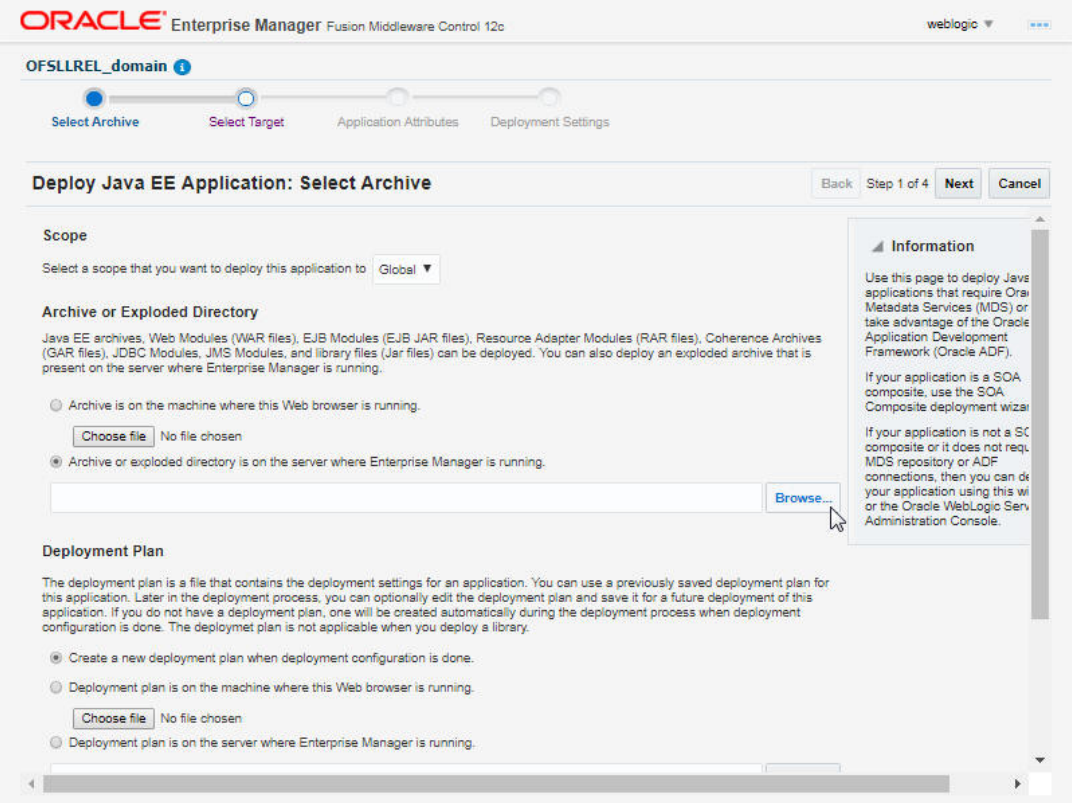
10. Right click on **WS_ManagedServer** in left panel, select Application Deployment > Deploy.

Figure 2-29 Deployment in EnterpriseManager 9



11. The following window is displayed.

Figure 2-30 Deployment in EnterpriseManager 10

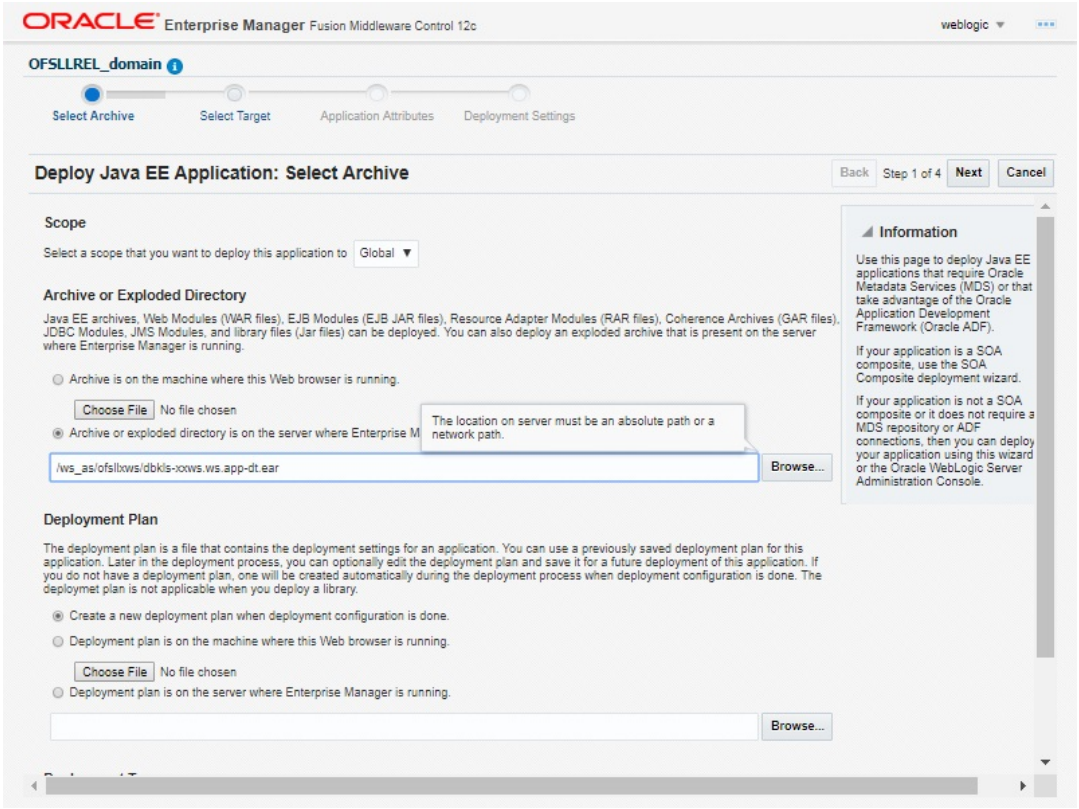


12. Browse to the folder containing the WebService. Eg:

    /ws-as/ofsllxws/dbkls-xxws.ws.app

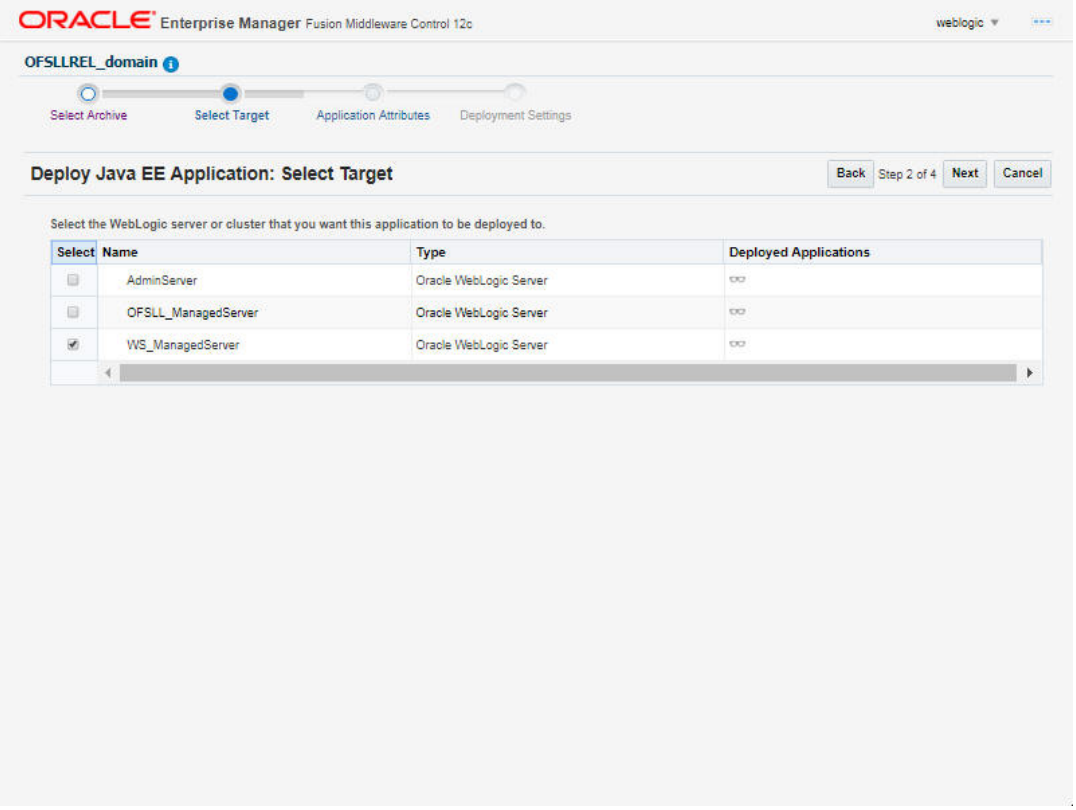
    Click **Next**.

Figure 2-31 Deployment in EnterpriseManager 11



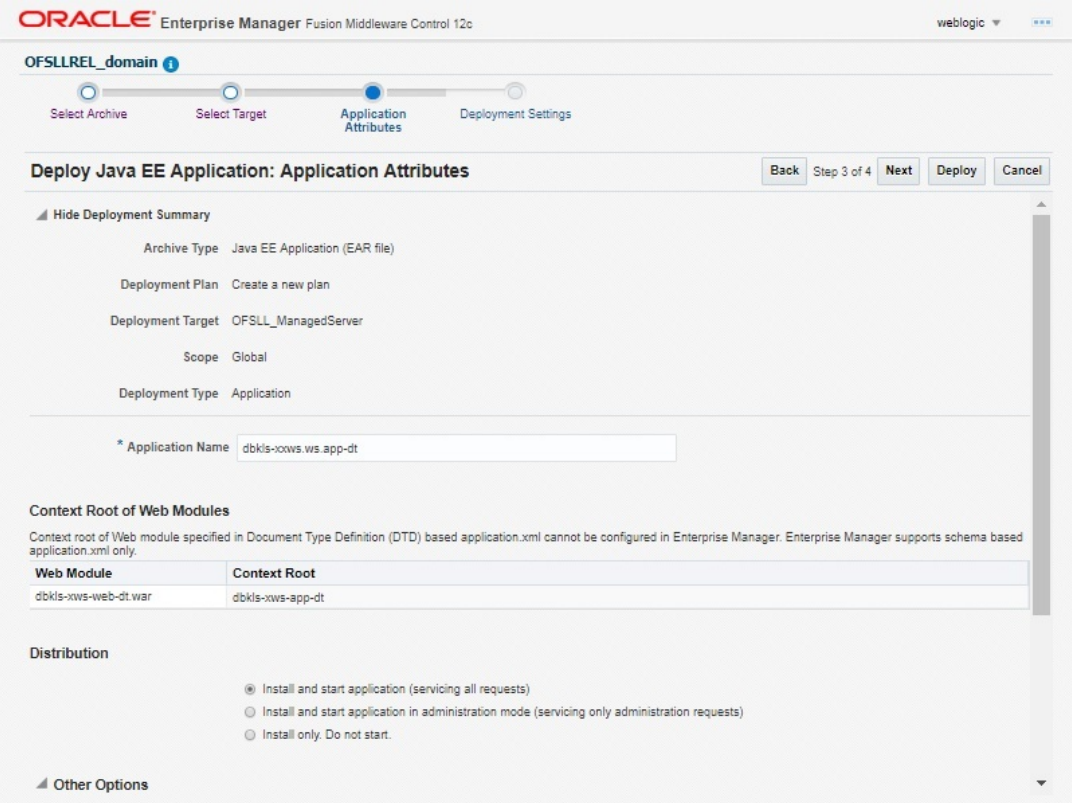
13. The following window is displayed. Select the server on which the WebService needs to be deployed. Click **Next**.

Figure 2-32 Deployment in EnterpriseManager 12



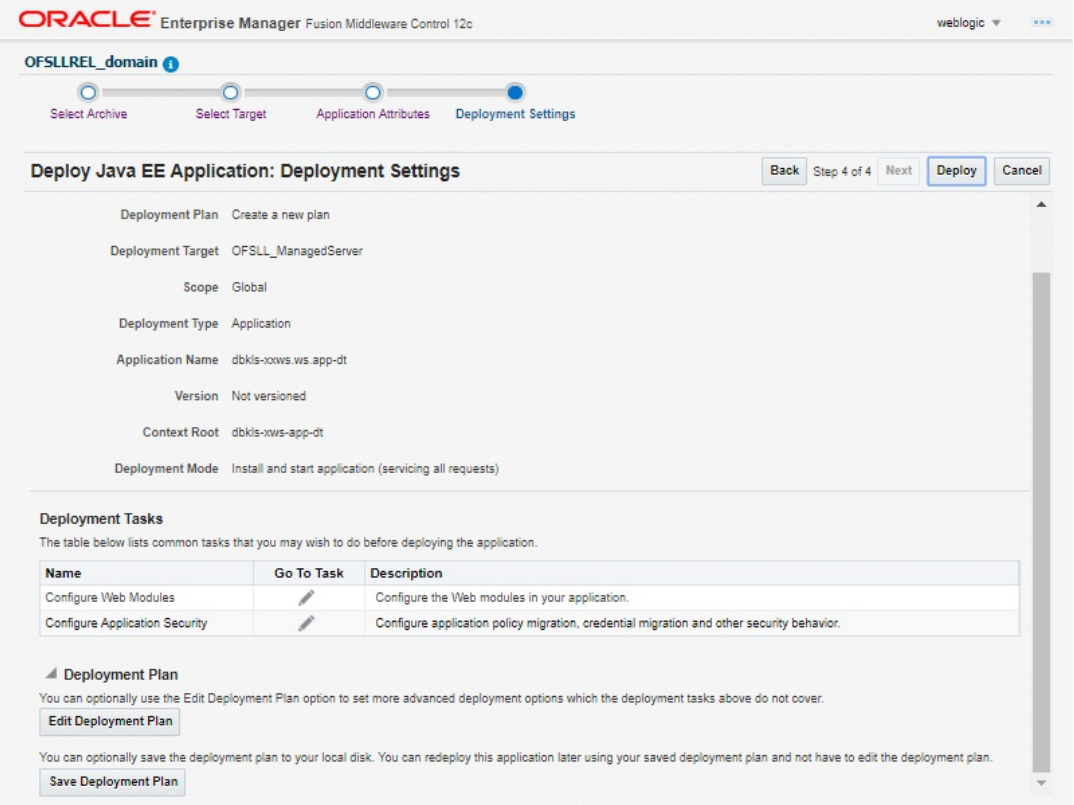
14. The following window is displayed. Check the context root and select the Distribution option as **Install and start application (servicing all requests)**. Click **Next**.

Figure 2-33 Deployment in EnterpriseManager 13



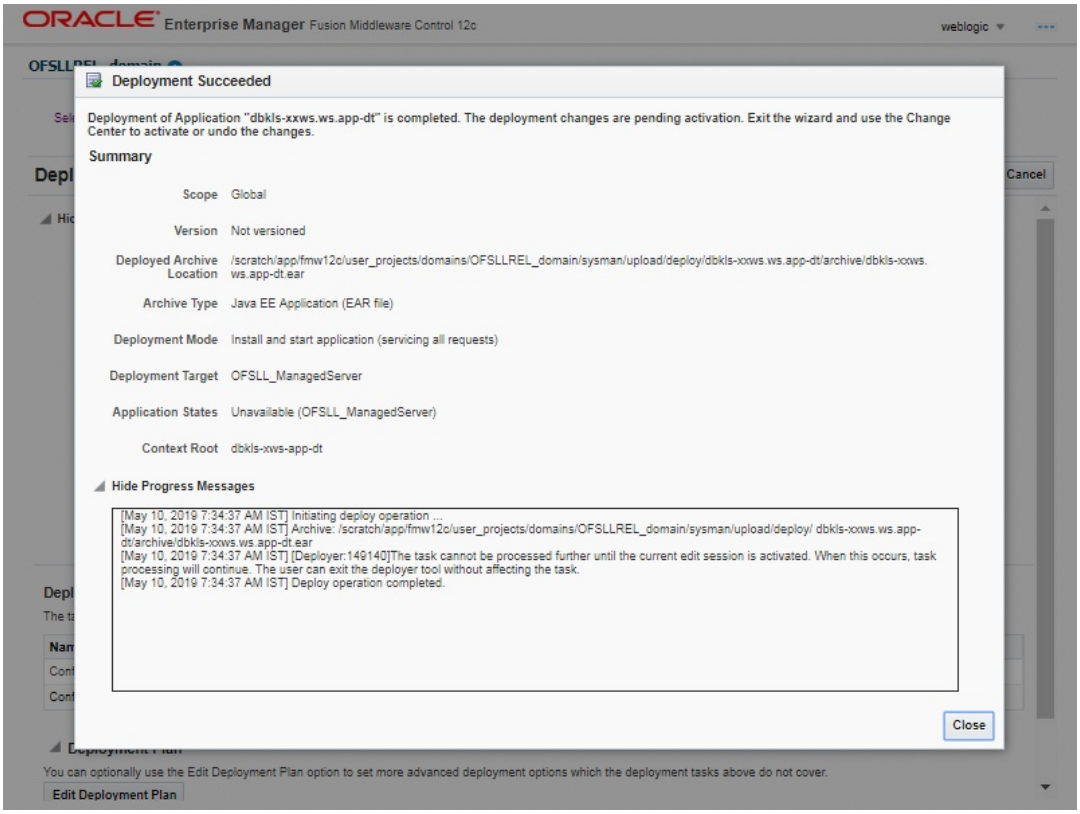
15. The following window is displayed. Click **Deploy**.

Figure 2-34 Deployment in EnterpriseManager 14



16. The following window is displayed. Click **Close**.

Figure 2-35 Deployment in EnterpriseManager 15



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:

Table 2-4 Service EAR

Service	Service EAR	Recommended Context	Remark
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

Table 2-4 (Cont.) Service EAR

Service	Service EAR	Recommended Context	Remark
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**.

3
Configure Weblogic Policy on WebServices

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

- Configuring Weblogic Policy on WebServices

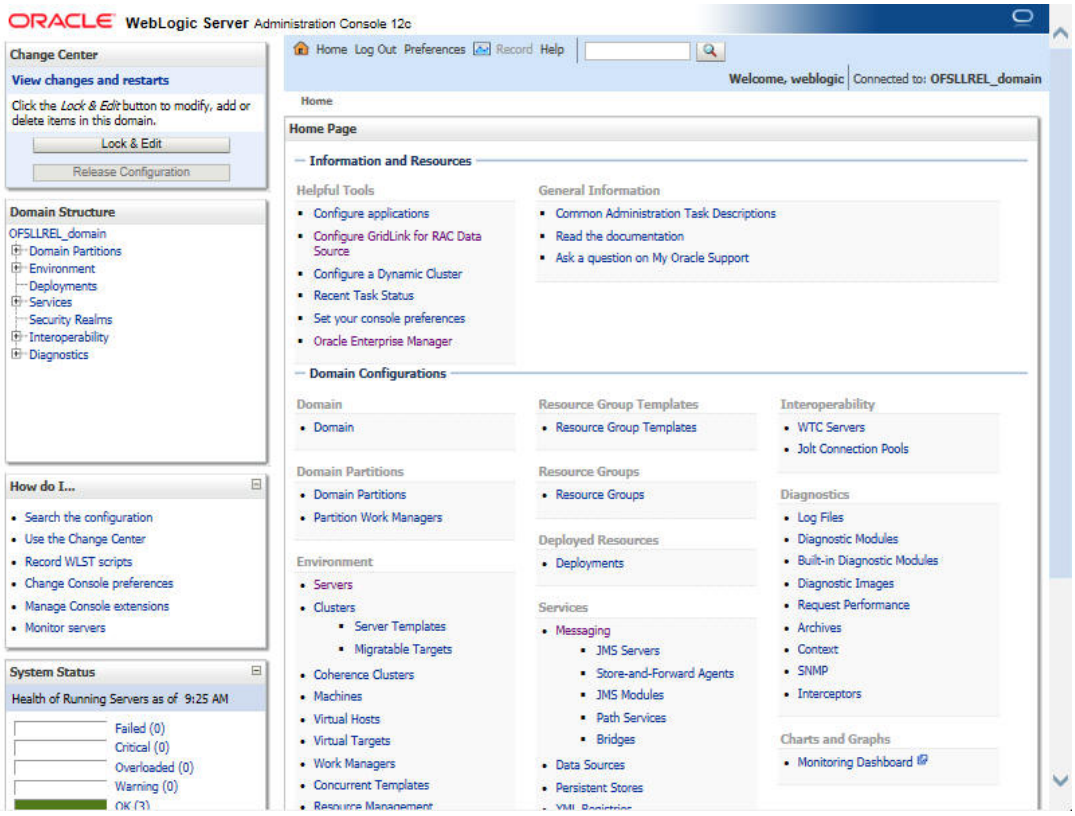
3.1 Configuring Weblogic Policy on WebServices

The following steps explains the procedure to configure Weblogic policy on WebServices.

1. Login to WebLogic application server console (`http://hostname:port/console`).

   The following window is displayed.

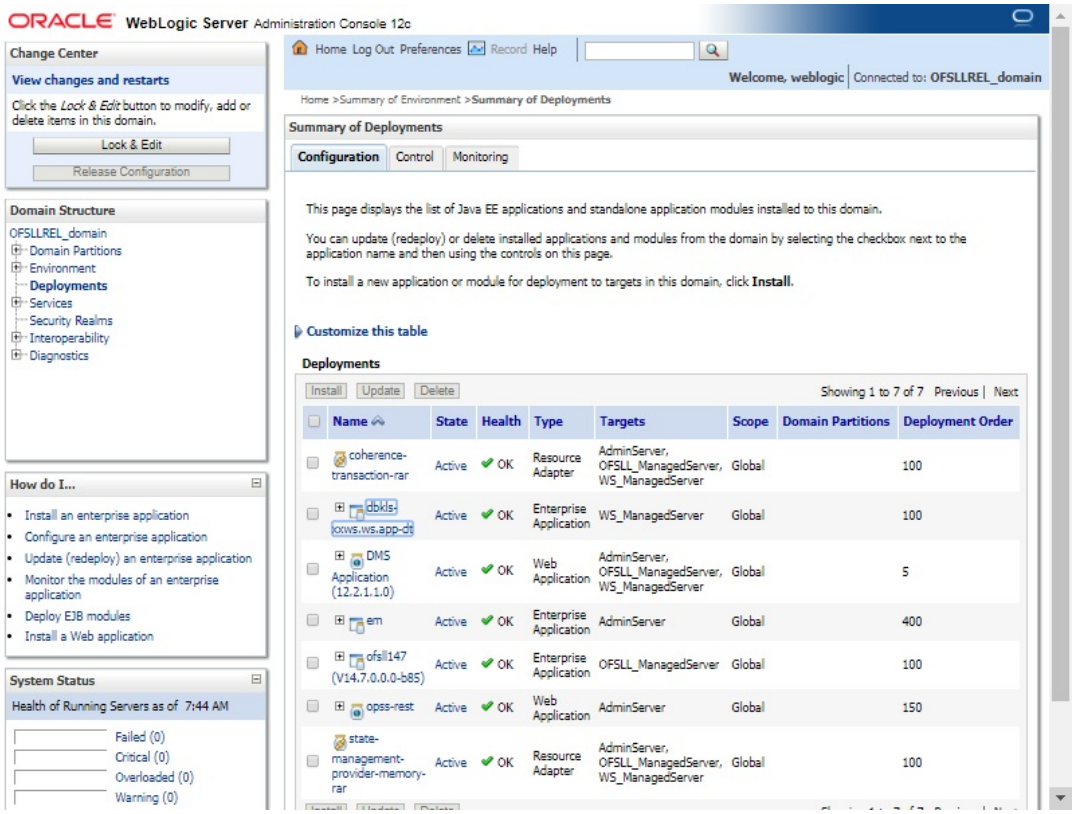
Figure 3-1 Configuring Weblogic Policy 1



2. Click **Deployments** which is available on both side panels as marked above.

   The following window is displayed.

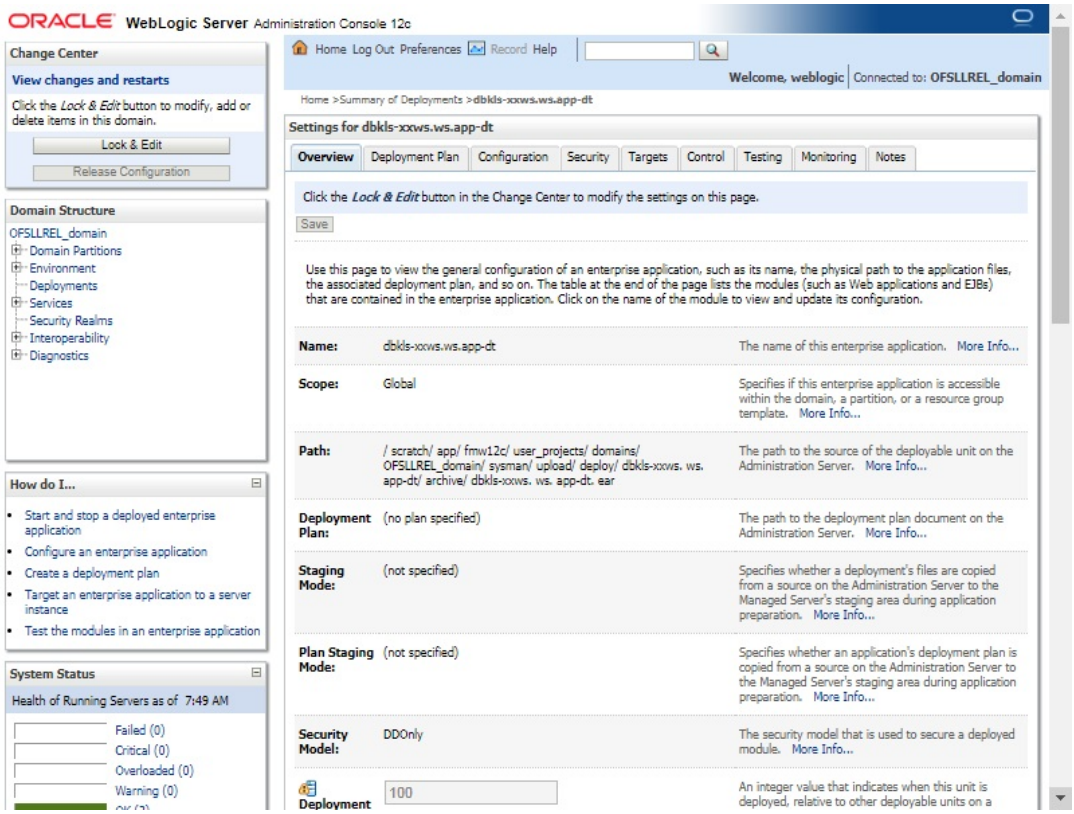
Figure 3-2 Configuring Weblogic Policy 2



3. Click on application name **dbkls-xws-app** on right side panel.

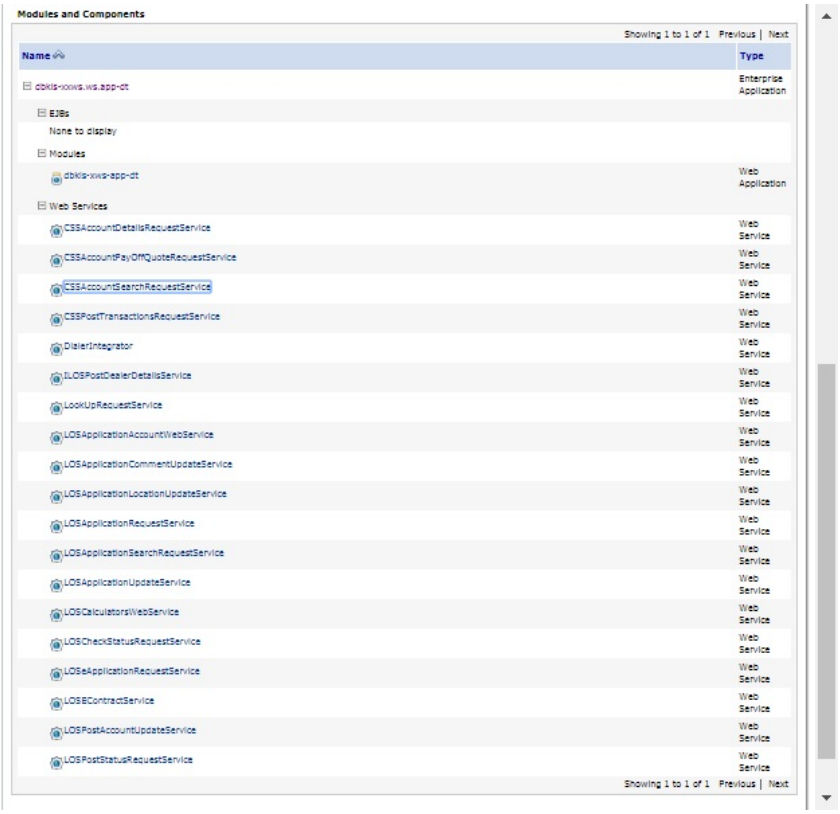
   The following window is displayed.

Figure 3-3 Configuring Weblogic Policy 3



4. Scroll down the page.

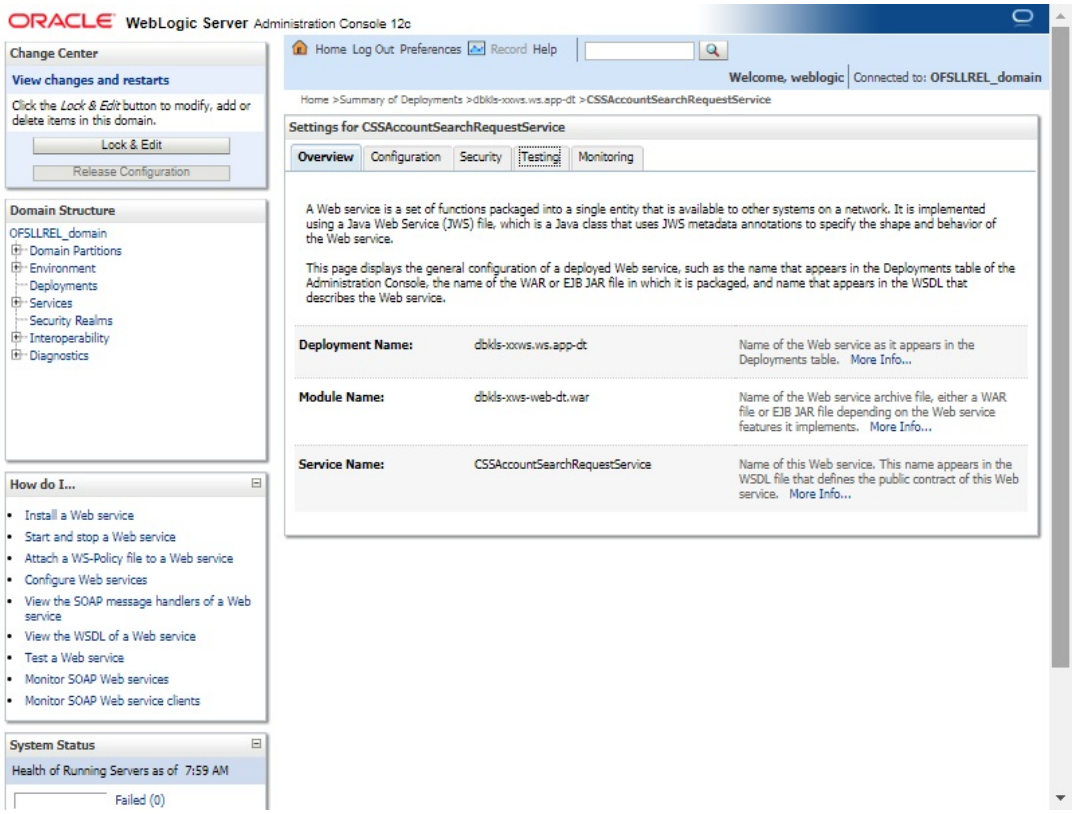
Figure 3-4 Configuring Weblogic Policy 4



5. Click **WebServices CSSAccountSearchRequestService**.

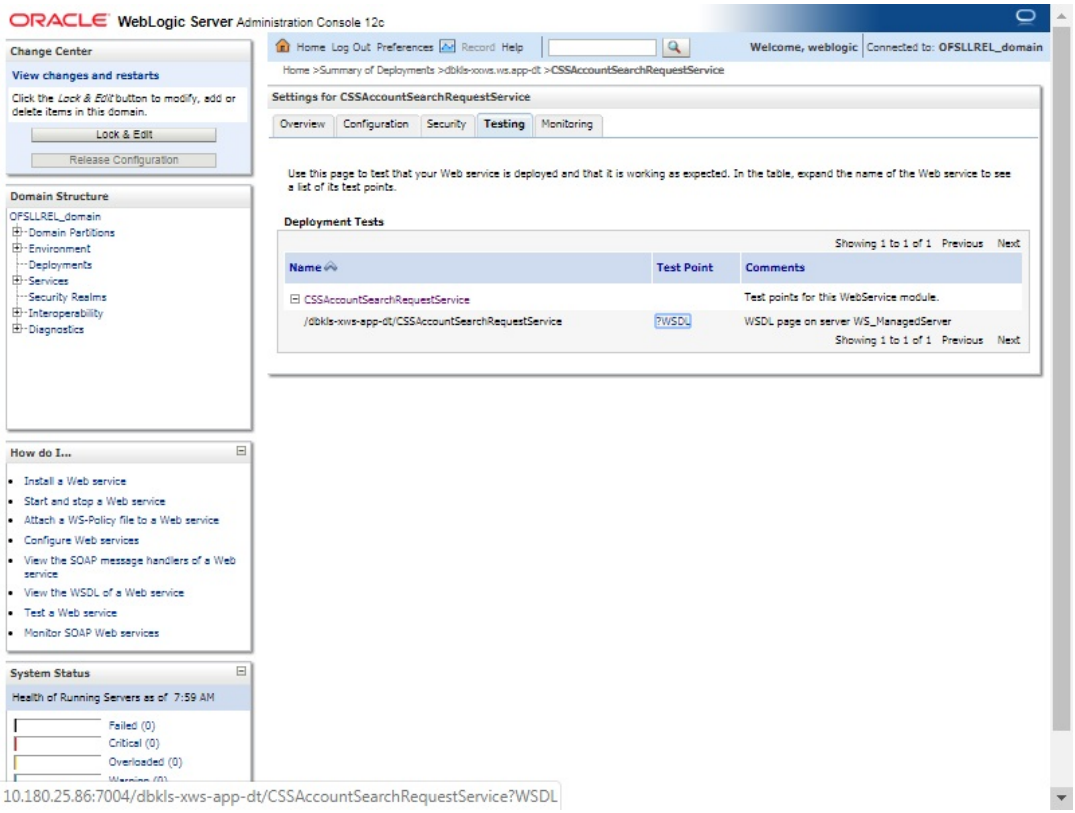
   The following window is displayed.

Figure 3-5 Configuring Weblogic Policy 5



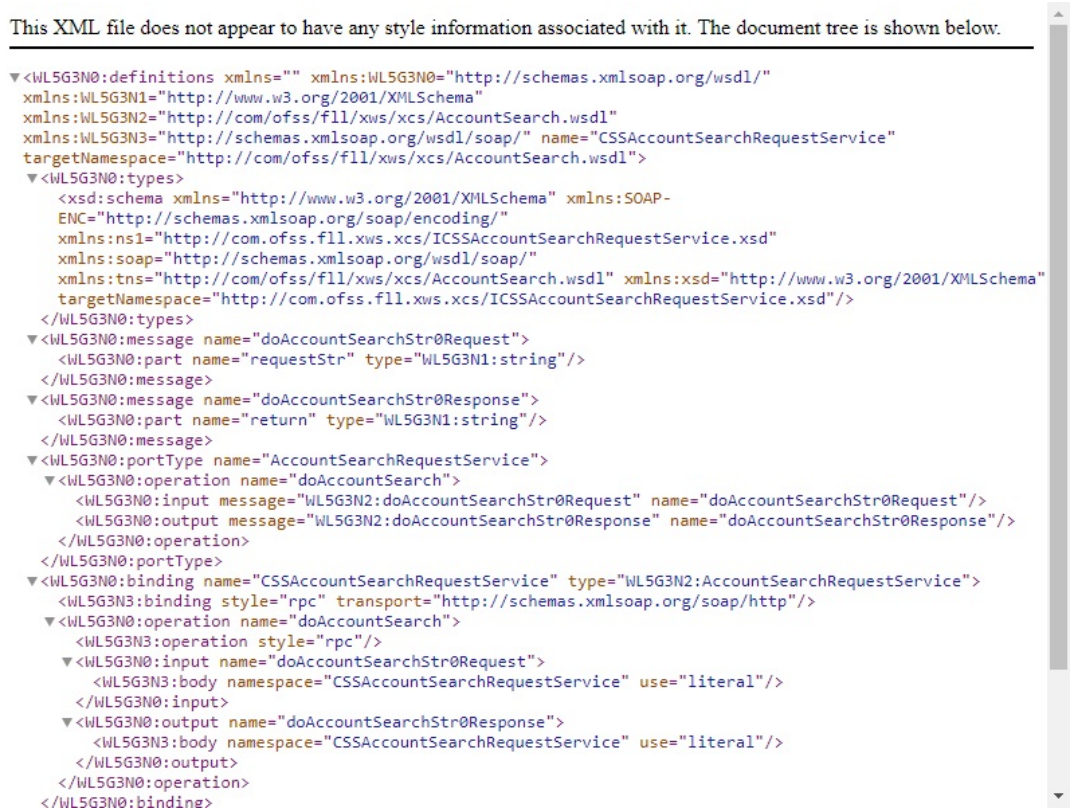
6. Select **Testing** tab and click **?WSDL** link in Test Point column adjacent to the service name in the table.

Figure 3-6 Configuring Weblogic Policy 6



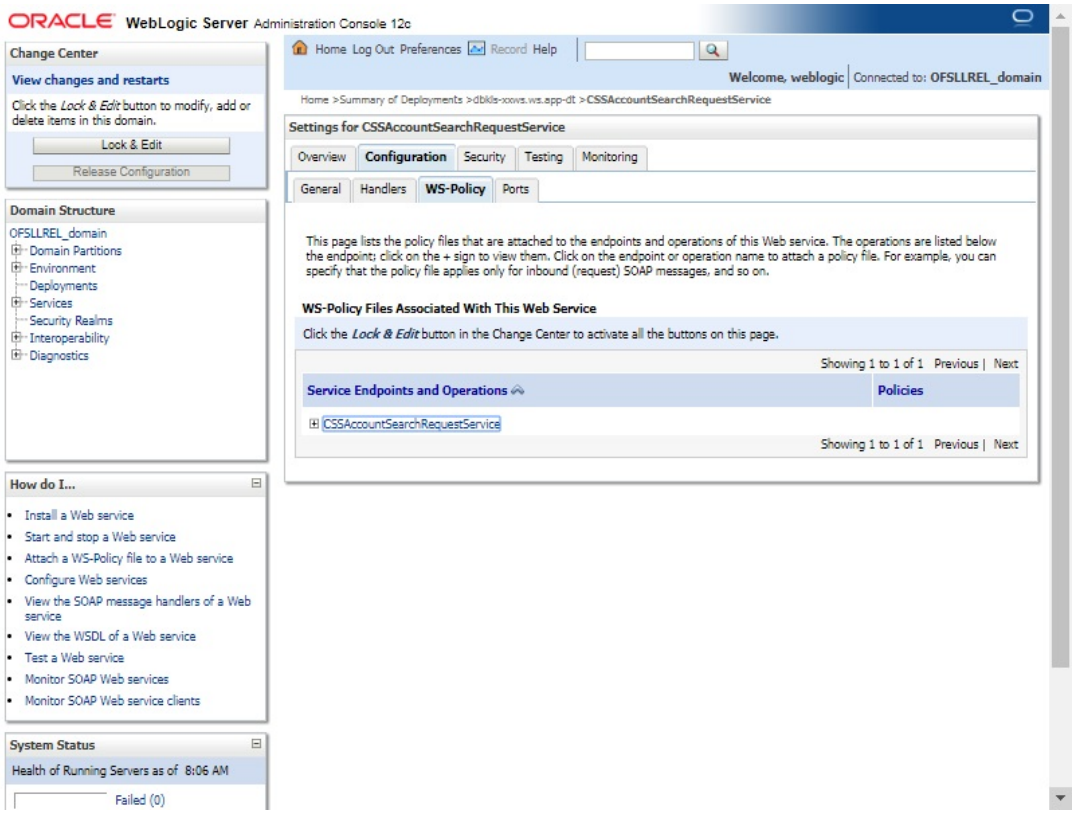
7. The following window is displayed. The WSDL will be accessible on http before applying WS-Policy.

Figure 3-7 Configuring Weblogic Policy 7



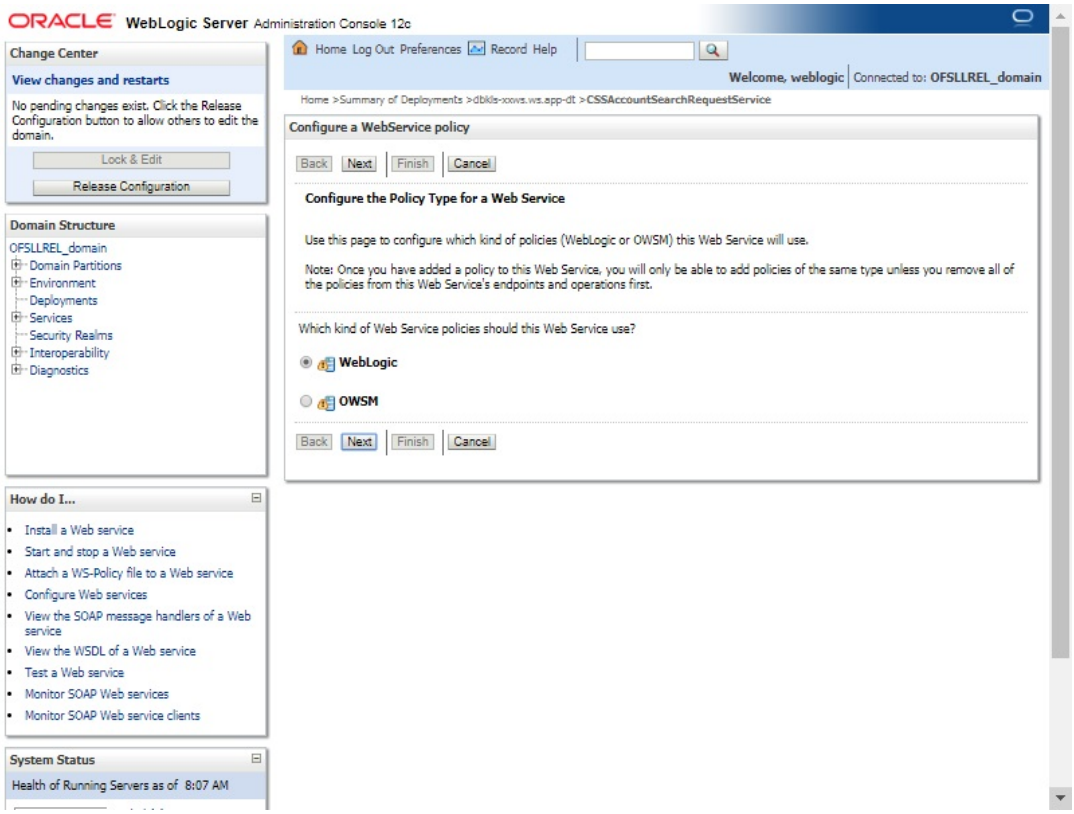
8. Navigate to Configuration > WS-Policy tab.

Figure 3-8 Configuring Weblogic Policy 8



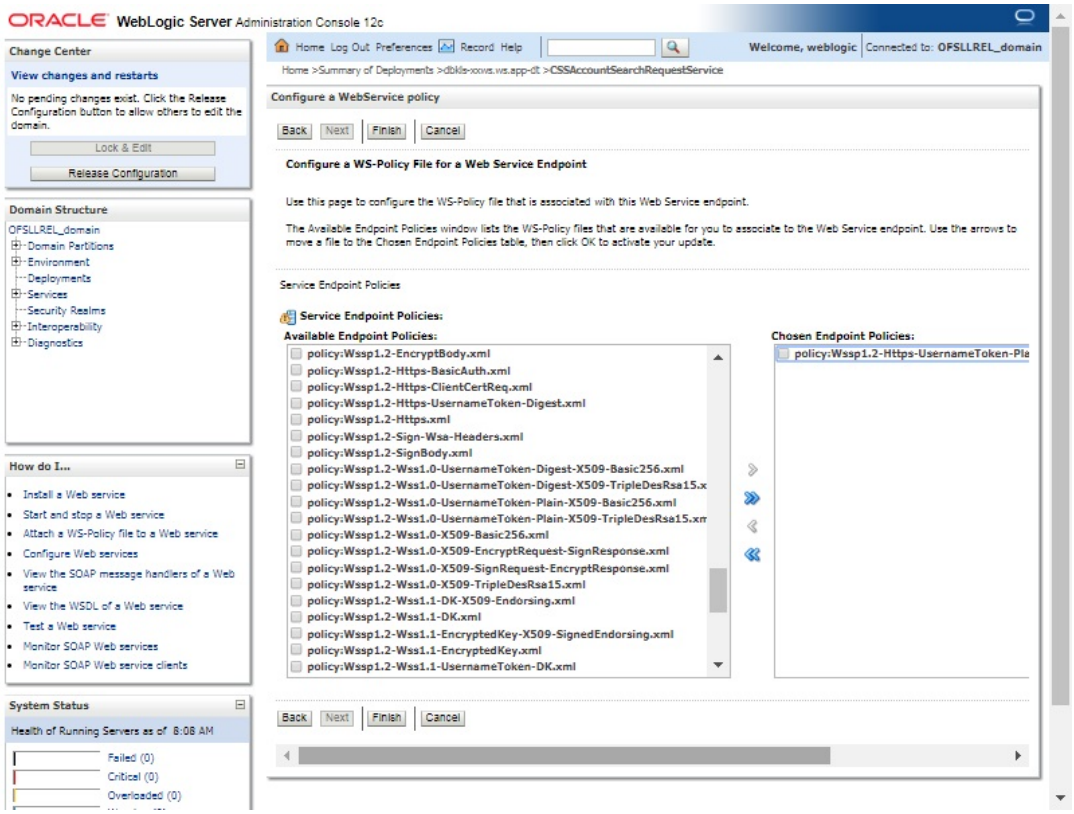
9. Click **CSSAccountSearchRequestService**.
10. In the below window, select **Weblogic** and click **Next**.

Figure 3-9 Configuring Weblogic Policy 9



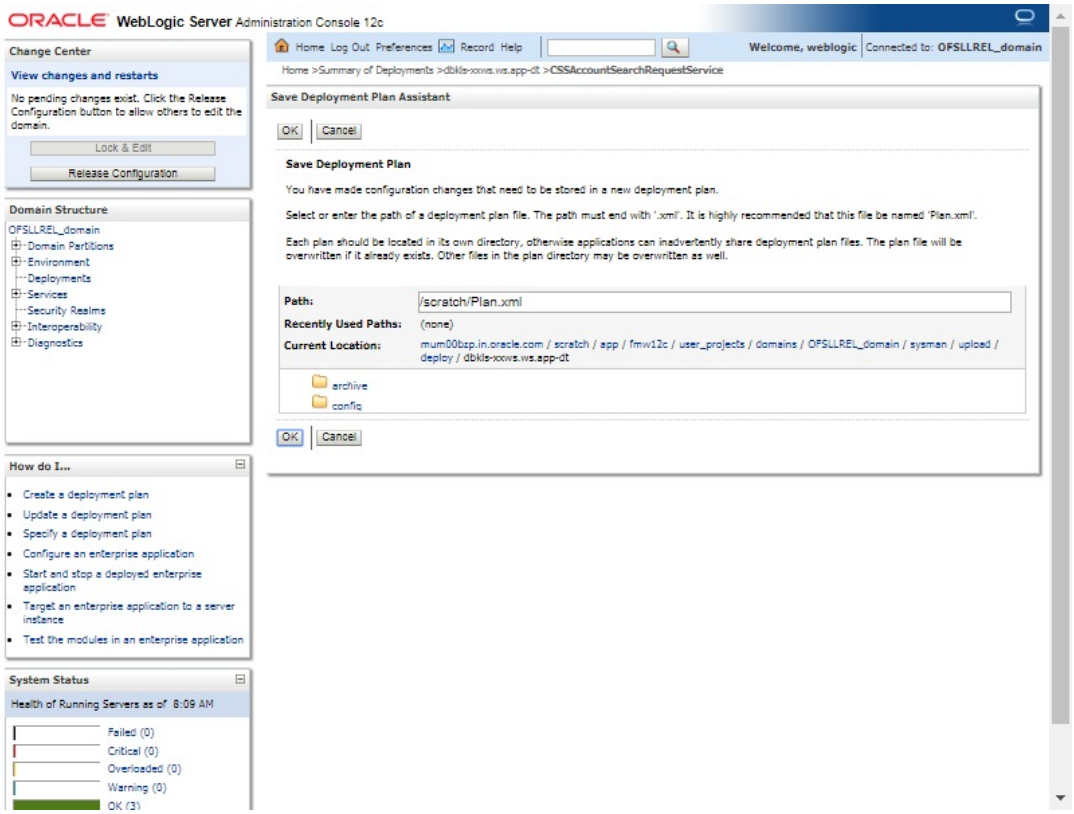
11. Select **policy:Wssp1.2-Https-UsernameToken-Plain.xml** and click right arrow to move it to the selected policies list.

Figure 3-10 Configuring Weblogic Policy 10



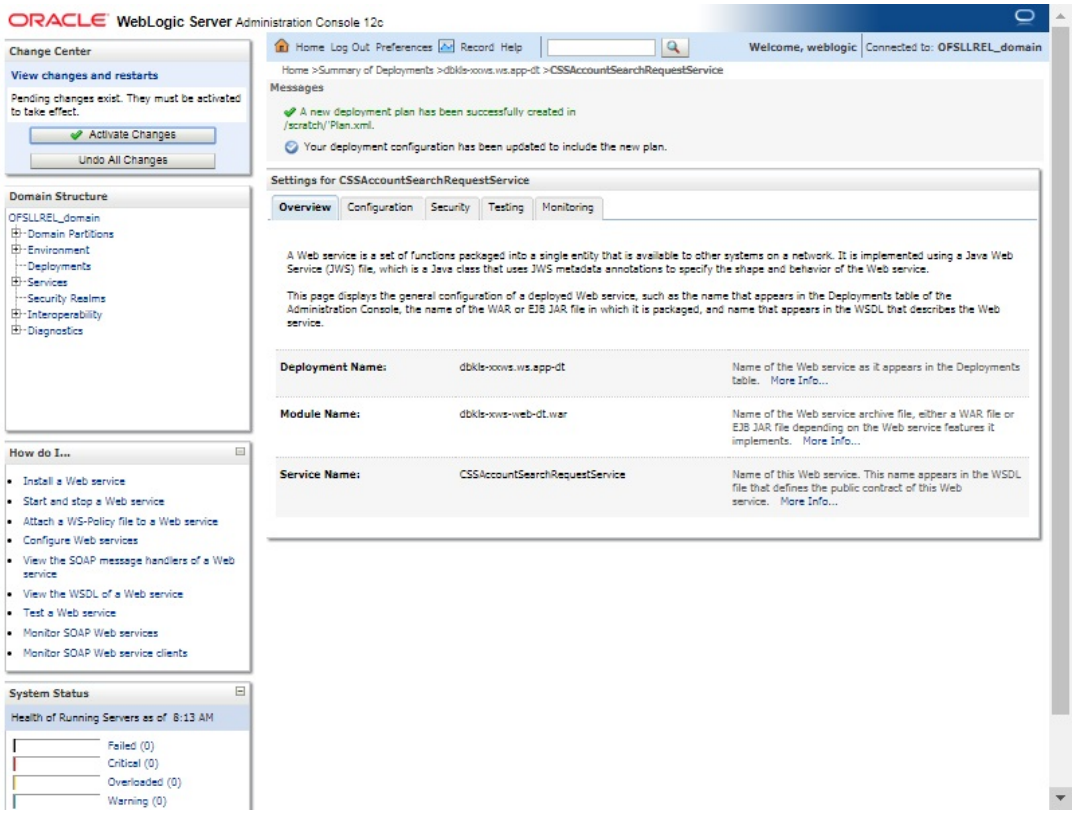
12. Click **OK**.

Figure 3-11 Configuring Weblogic Policy 11



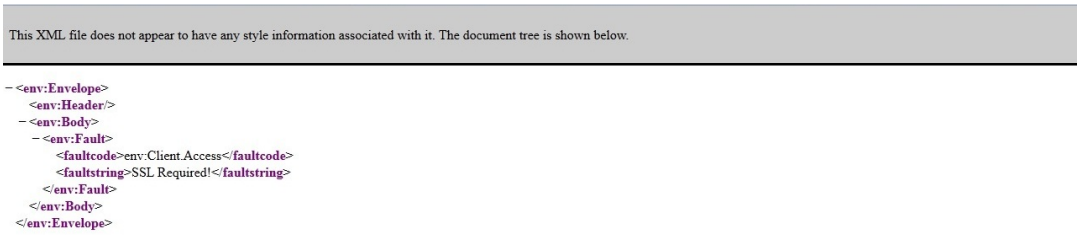
13. Browse to any folder to save Deployment Plan and click **OK**.

Figure 3-12 Configuring Weblogic Policy 12



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

Figure 3-13 Configuring Weblogic Policy 13

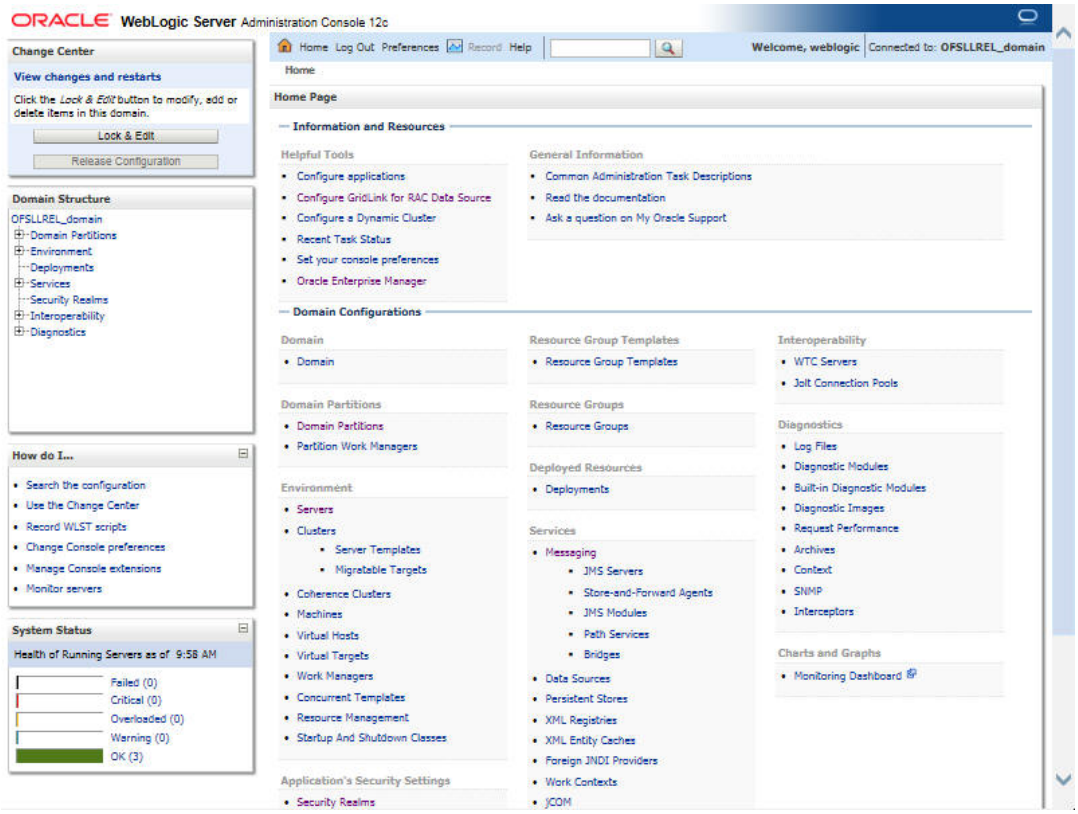


> **Note:**
>
> For SSL communication, the vendor servers seek public certificates. Hence, you need to download the certificates from vendor website and import into your java keystore. You then need to configure Weblogic to present the certificates to vendor servers for successful handshake.

15. Select the environment on the left pane.

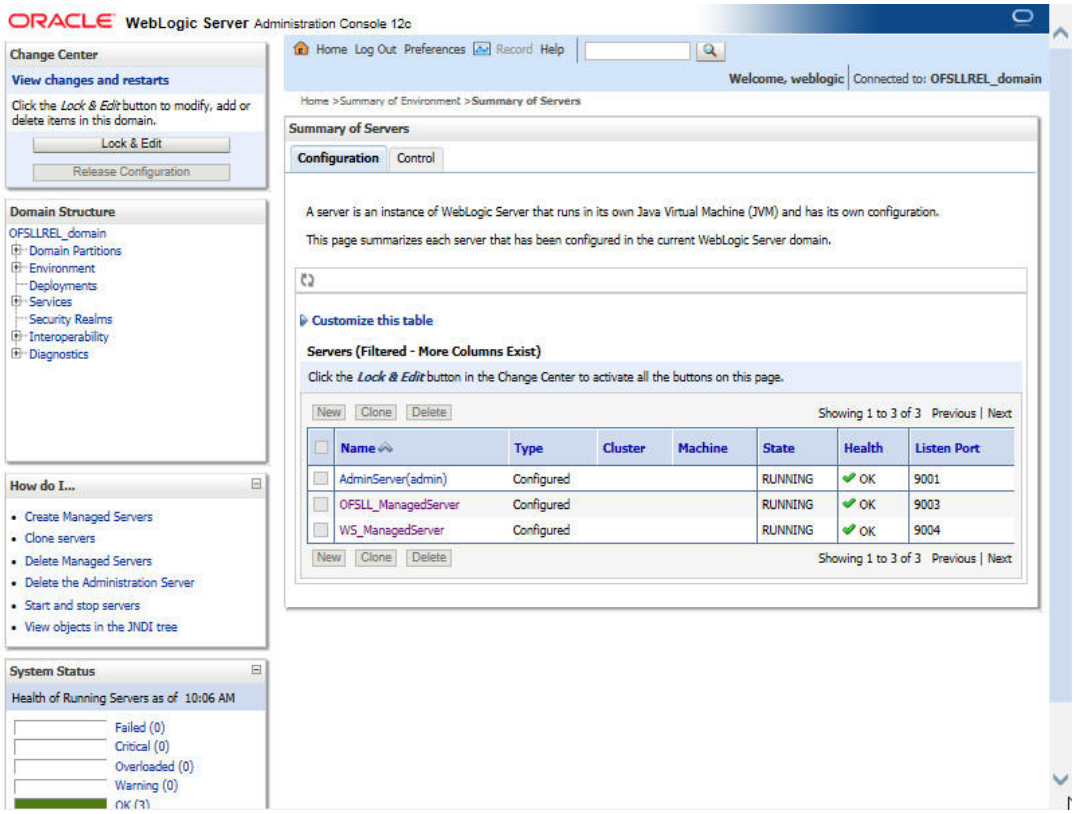
    The following window is displayed.

Figure 3-14 Configuring Weblogic Policy 14



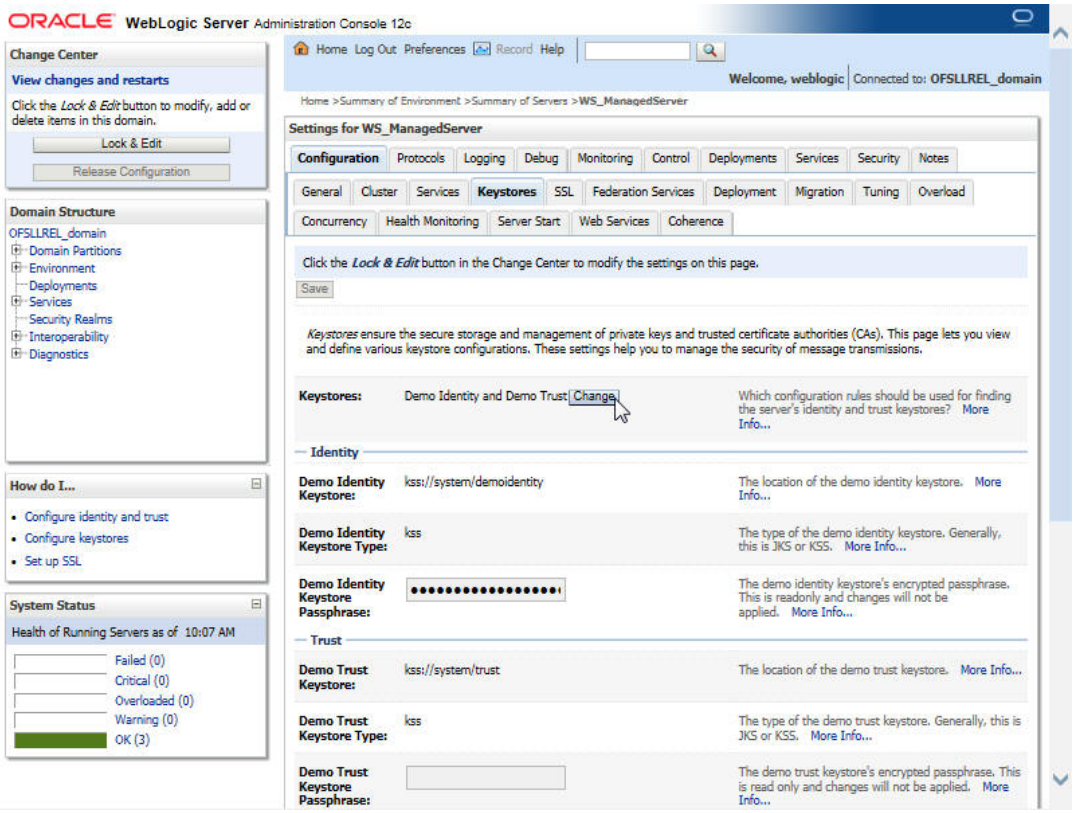
16. Click **Servers**.

Figure 3-15 Configuring Weblogic Policy 15



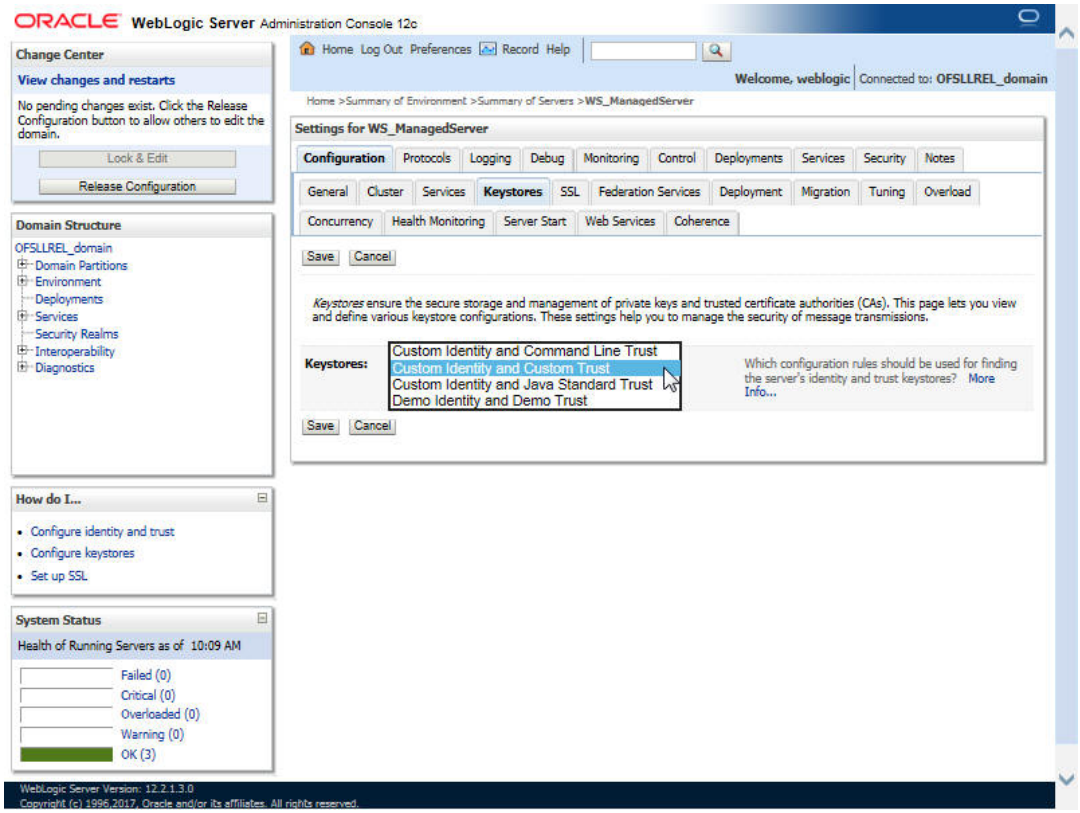
17. Select the Server into which the WebServices are deployed.

Figure 3-16 Configuring Weblogic Policy 16



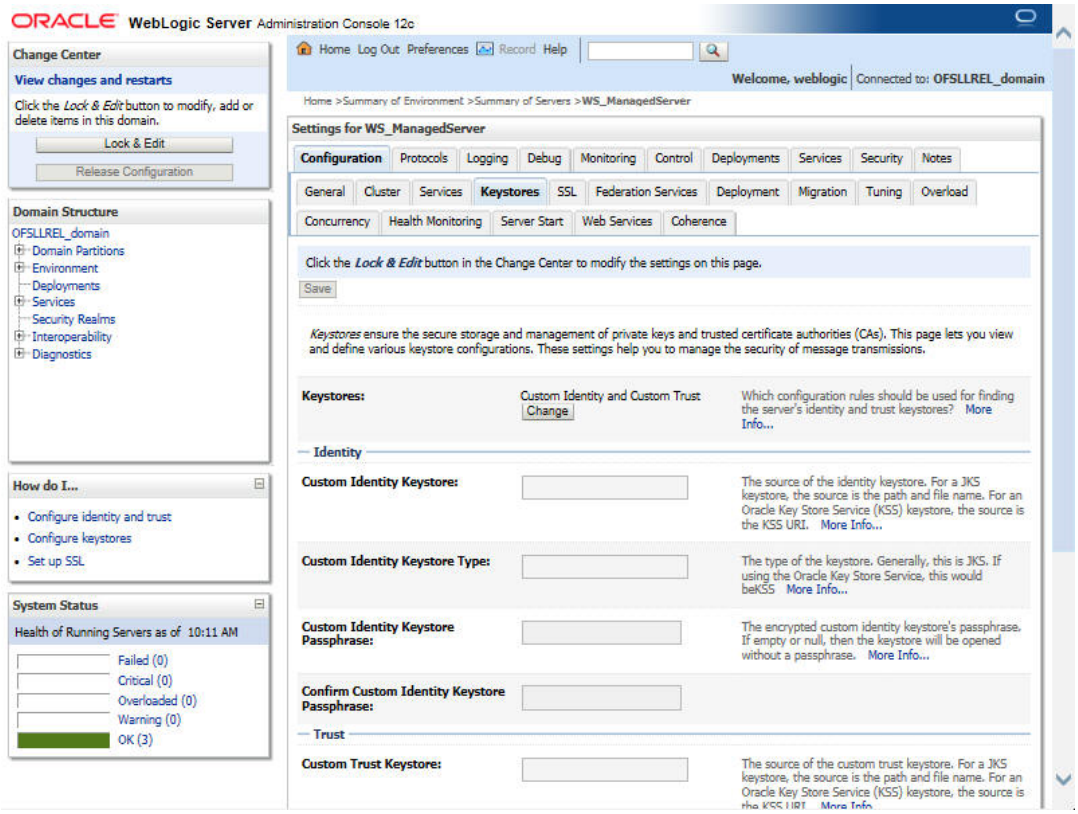
18. Select **Keystores**.

Figure 3-17 Configuring Weblogic Policy 17



19. Click **Change** and select **Custom Identity and Custom trust** from the drop-down list.

Figure 3-18 Configuring Weblogic Policy 18

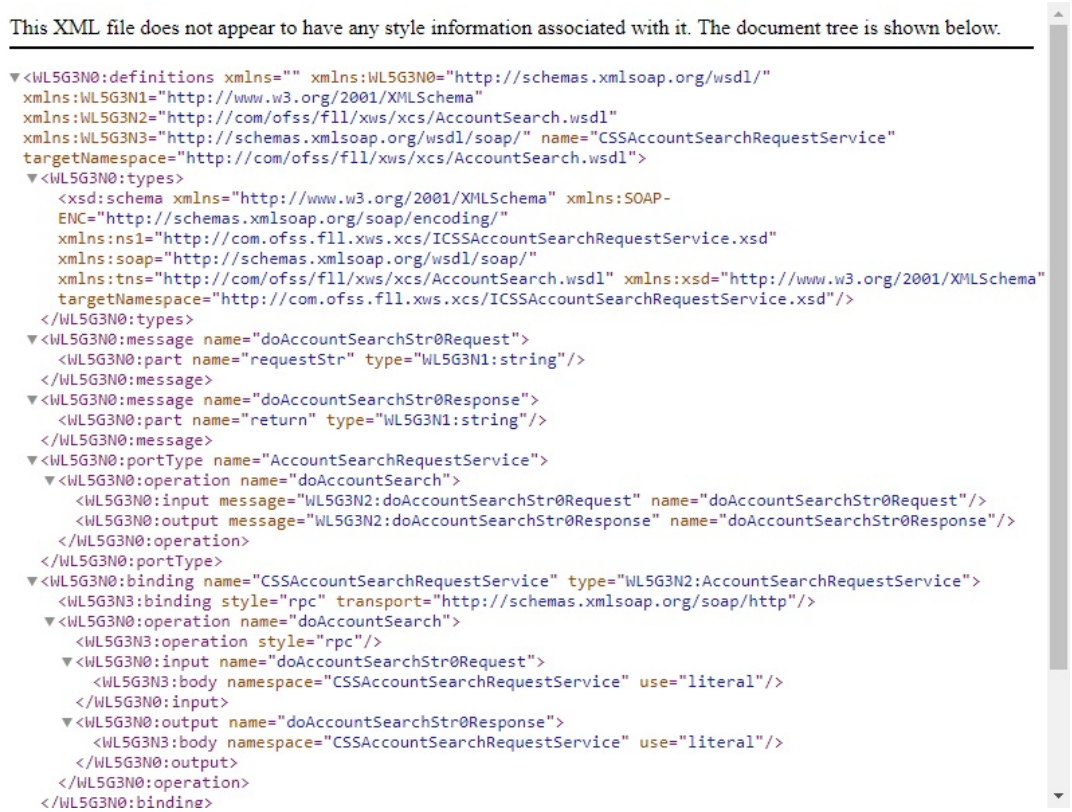


Specify

- 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

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

Figure 3-19 Configuring Weblogic Policy 19



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

- CSSAccountDetailsRequestService
- CSSAccountPayOffQuoteRequestService
- CSSAccountSearchRequestService
- CSSPostTransactionsRequestService

4

Verify Successful Installation

The following section details the steps to be followed to verify successful Webservices deployment and verify successful Interface (Route One) deployment

- Verifying Successful Webservices deployment:
- Verifying Successful Interface (Route One) deployment:

4.1 Verifying Successful Webservices deployment:

The following are the steps to verify successful Webservices deployment.

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

4.2 Verifying Successful Interface (Route One) deployment:

The following are the steps to verify successful Interface (Route One) deployment.

1. Ensure that the state is ACTIVE and health is 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"

5
Enable Logging

The following sections explain the process of enabling WebService Log and enabling SQL Log for WebServices Component.

- Enabling WebService Log
- Enabling SQL Log for WebServices Component

5.1 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-dt.ear/dbkls-xws-web-dt.war/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:

Table 5-1 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

5.2 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.

Table 5-2 Service and Debug level mapping

Service			
DialerIntegrator: xcsadi_em_100_01	To receive loan application from dealer track: XAEPRC_EM_100_01	To receive edocs application update : XAEEDS_EM_100_01	To receive loan application from route one:XAEPRC_EM_100_01
CSSAccountDetailsRequestService: XCSPRC_EM_100_01	To receive loan application/deal update from dealer track: XAEUPD_EM_100_01	To receive edocs comment update : XAEEDS_EM_100_01	To receive loan application/deal refresh from route one : XAEUPD_EM_100_01
CSSAccountSearchRequestService: XCSACS_EM_100_01	To receive comments from dealer track : XAEPRC_EM_100_01	To receive edocs location update : XAEEDS_EM_100_01	To receive comments from route one: XEAUPD_EM_100_01
CSSPostTransactionsRequestService: XCSUPD_EM_100_01	To post comments to dealer track: XAECHK_EM_100_01		To post comments to route one: XAECHK_EM_100_01
CSSAccountPayOffQuoteRequestService: N/A	To post application status to dealer track: XAECHK_EM_100_01		To post application status to route one: XAECHK_EM_100_01
LOSeApplicationRequestService: XAEEDS_EM_100_01	To post dealer details to dealer track: XPRPRC_EM_100_01		To post dealer details to dealer track: XPRPRC_EM_100_01
LOSApplicationRequestService: XAEPRC_EM_100_01			
LOSApplicationSearchRequestService: XAEQUE_EM_100_01			
LOSApplicationLocationUpdateService: XAEEDS_EM_100_01			
LOSApplicationCommentUpdateService: XAEEDS_EM_100_01			
LOSPostStatusRequestService: XAECHK_EM_100_01			
LOSApplicationUpdateService: XAEUPD_EM_100_01			
LOSCheckStatusRequestService: XAECHK_EM_100_01			
LOSEcontractService: XACPRC_EM_100_01			

6
Configure RESTful WebService

Follow the below steps to configure RESTful WebService.

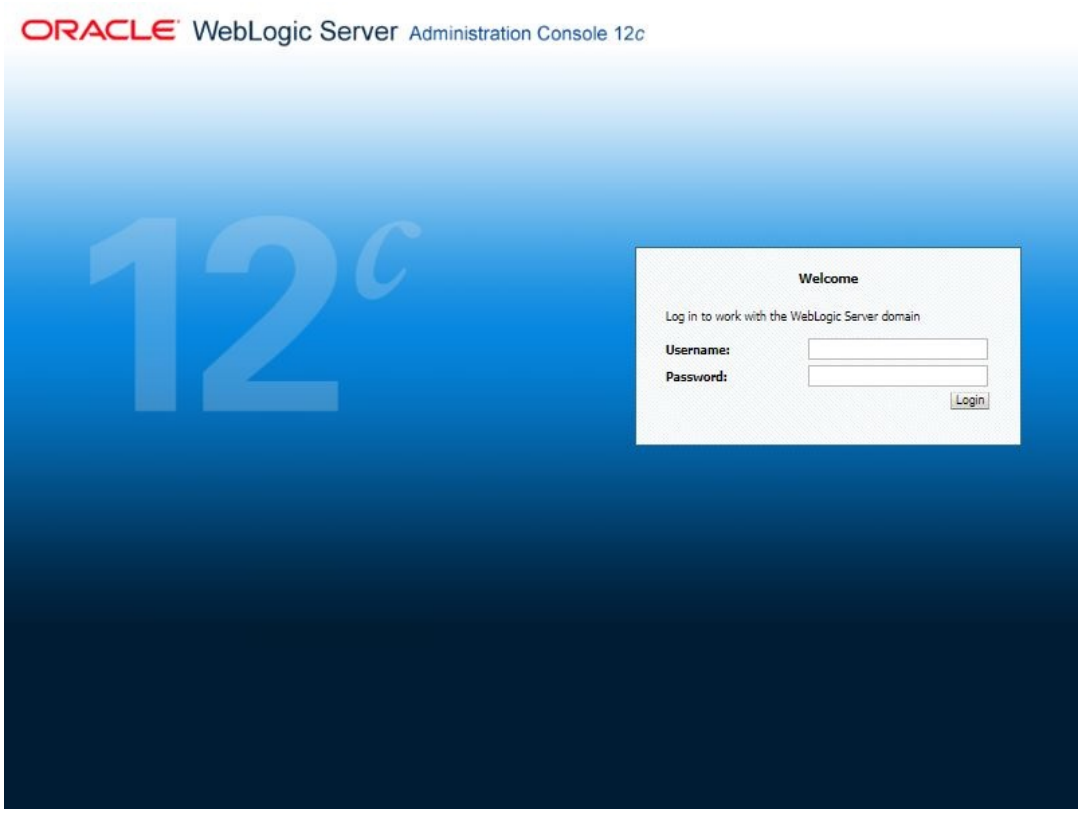
- Creating Data Sources for RESTful WebService
- Statement Timeout Configuration
- OAuth Implementation
- Deploying RESTful WebService
- Deploying RESTful Credit Bureau WebService

6.1 Creating Data Sources for RESTful WebService

Please follow the below steps to create data Sources for RESTful WebService.

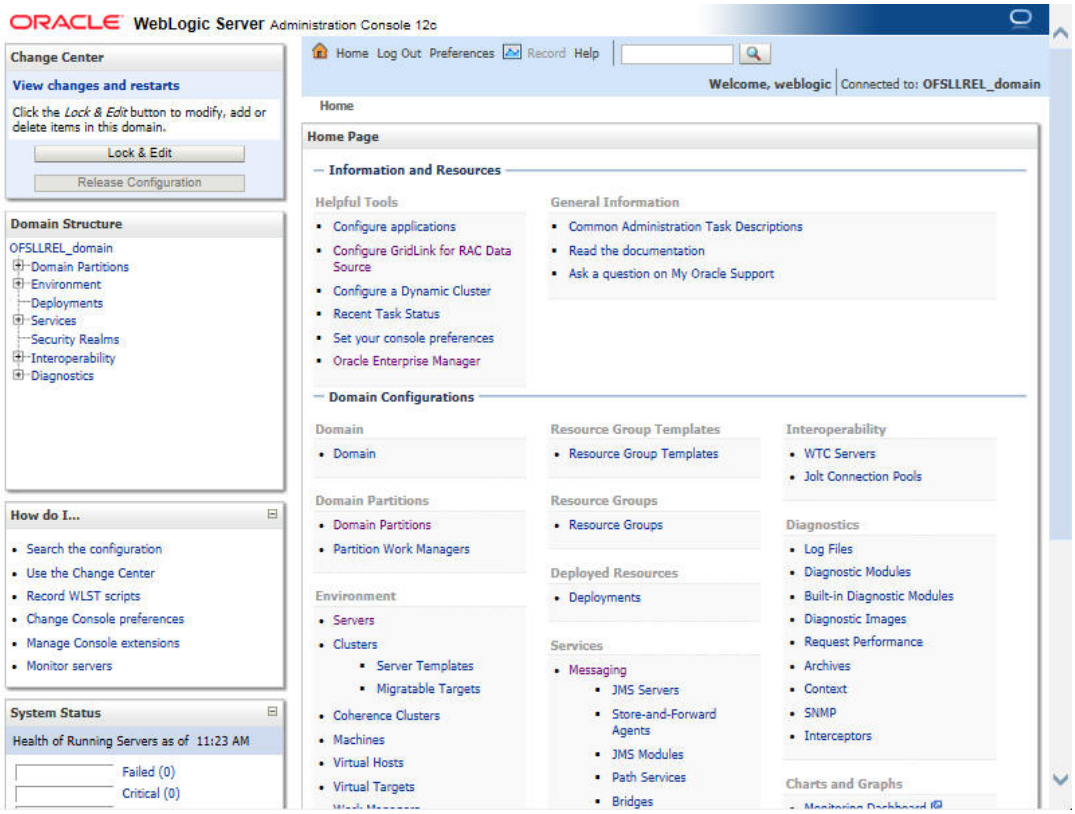
1. Login to Oracle Weblogic 12c console (`http://hostname:port/console`).

Figure 6-1 Create RESTful WebService 1



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

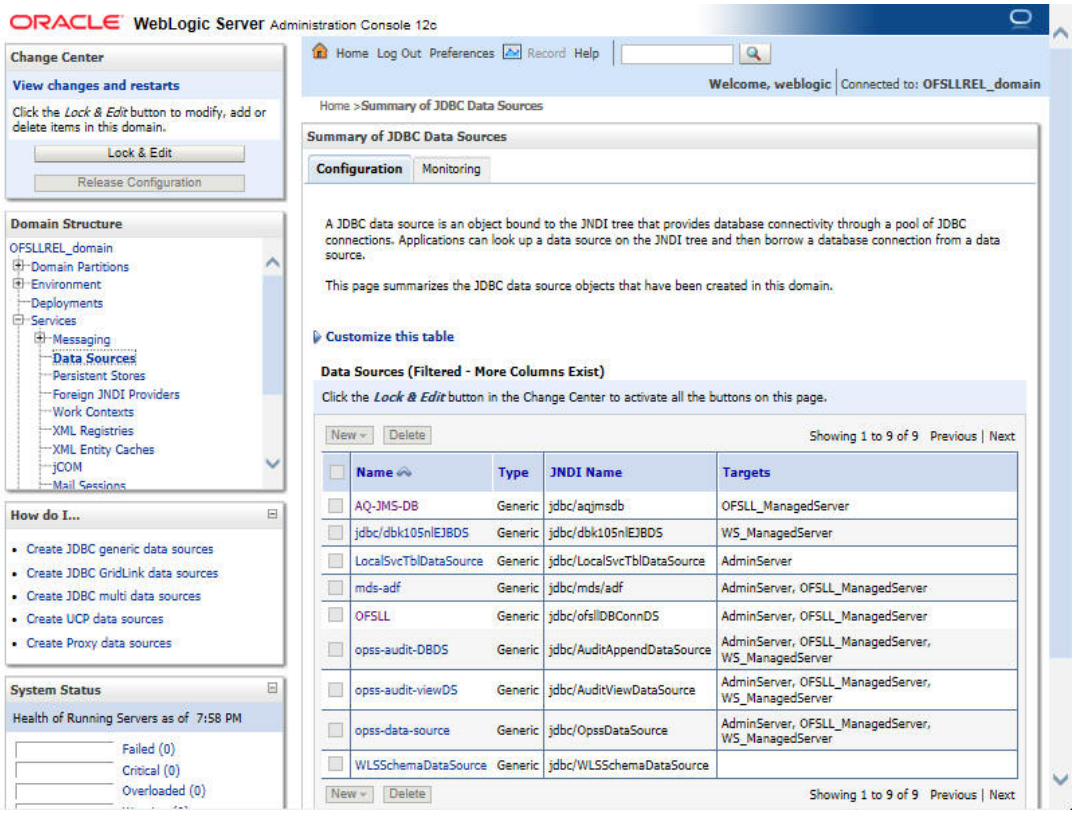
Figure 6-2 Create RESTful WebService 2



3. Click Domain Name > Services > Data Sources.

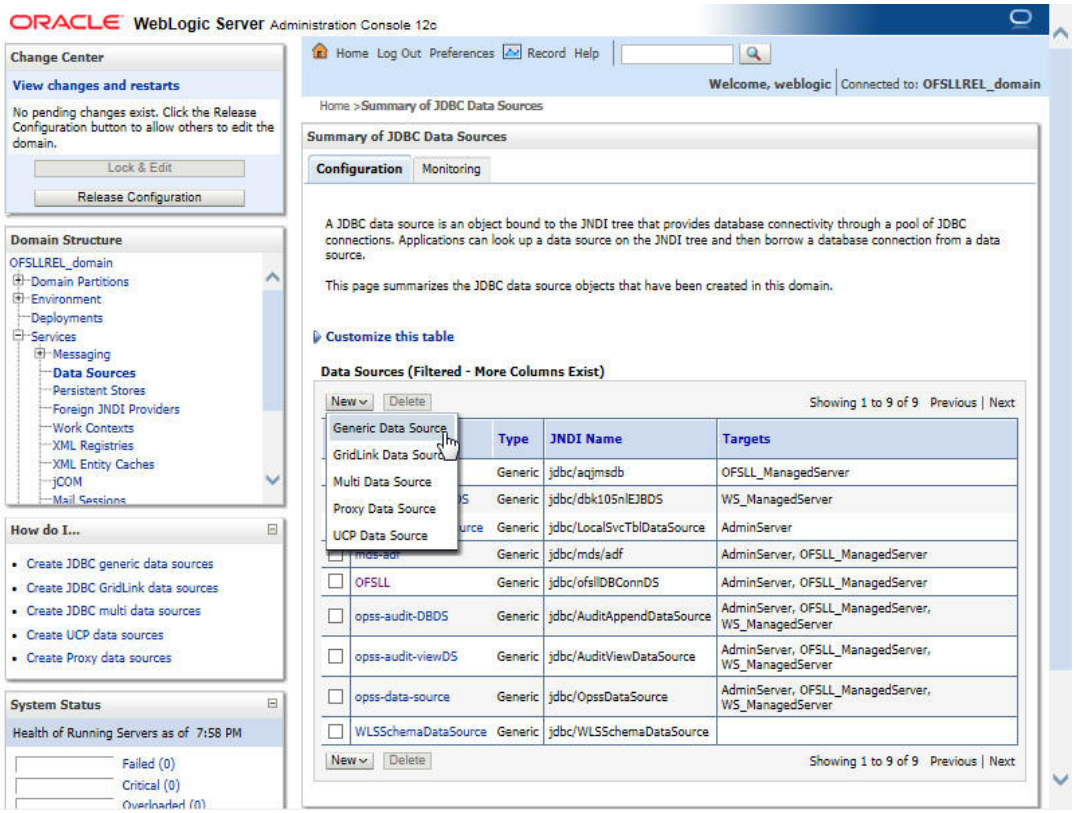
   The following window is displayed.

Figure 6-3 Create RESTful WebService 3



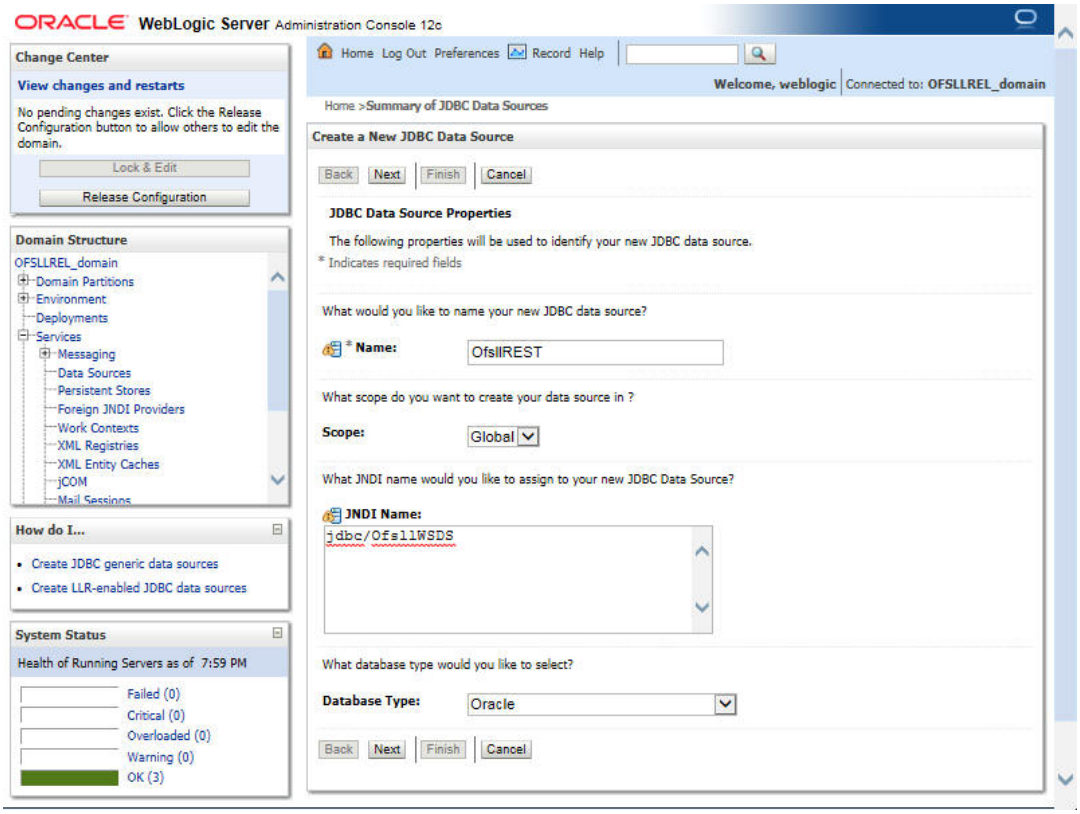
4. Click **Lock & Edit** button on the left panel. Click **New** on right panel and select **Generic Data Source**.

Figure 6-4 Create RESTful WebService 4



5. The following window is displayed.

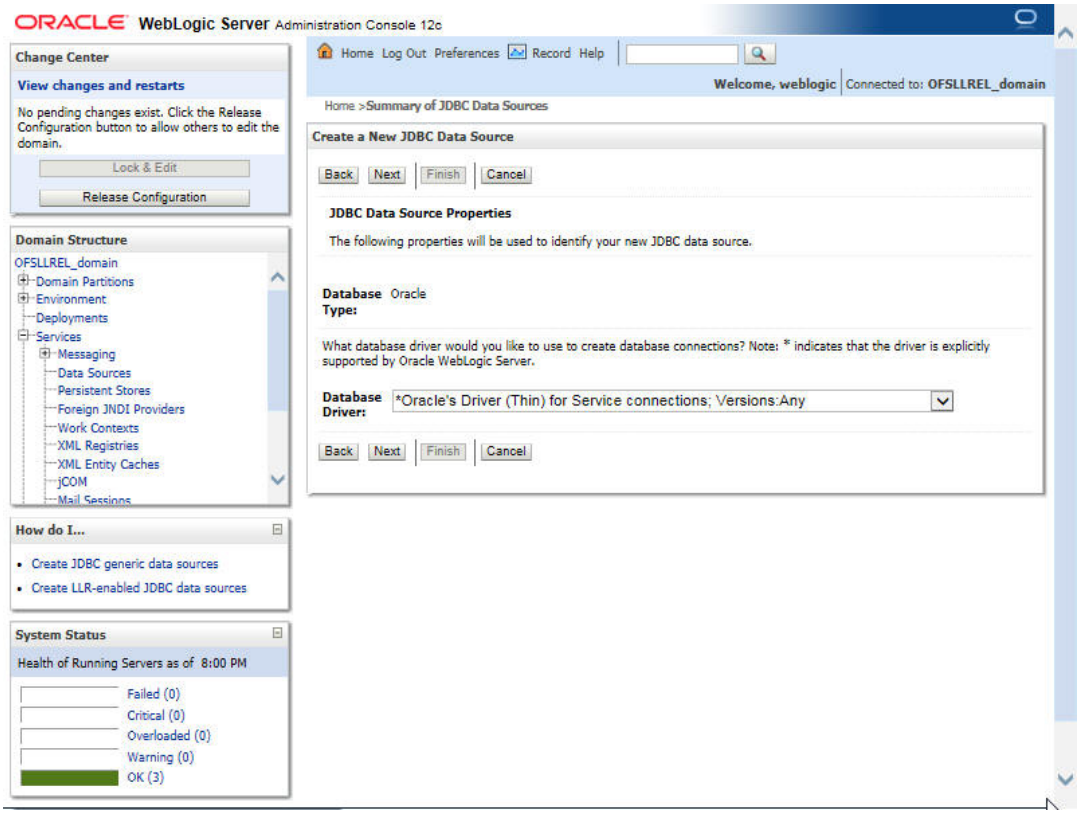
Figure 6-5 Create RESTful WebService 5



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.

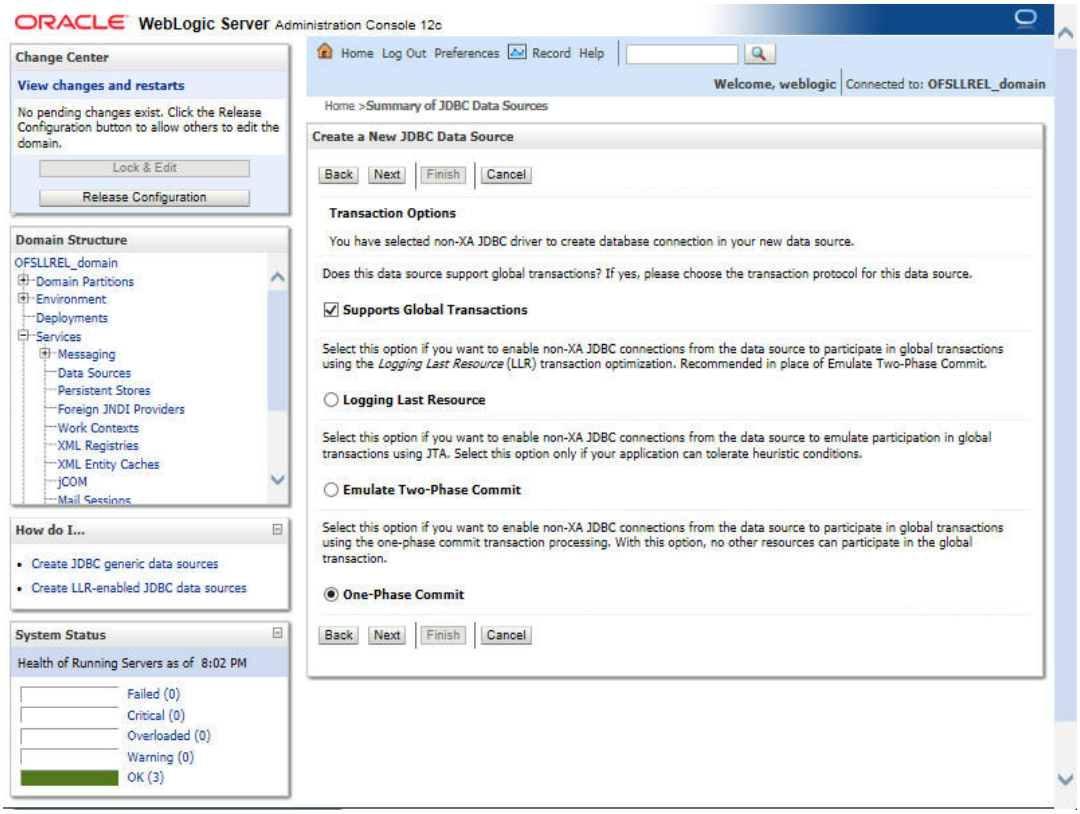
Figure 6-6 Create RESTful WebService 6



8. Select the Database Driver **Oracle's Driver(Thin) for Services connections;Versions:Any**.
9. Click **Next**.

   The following window is displayed.

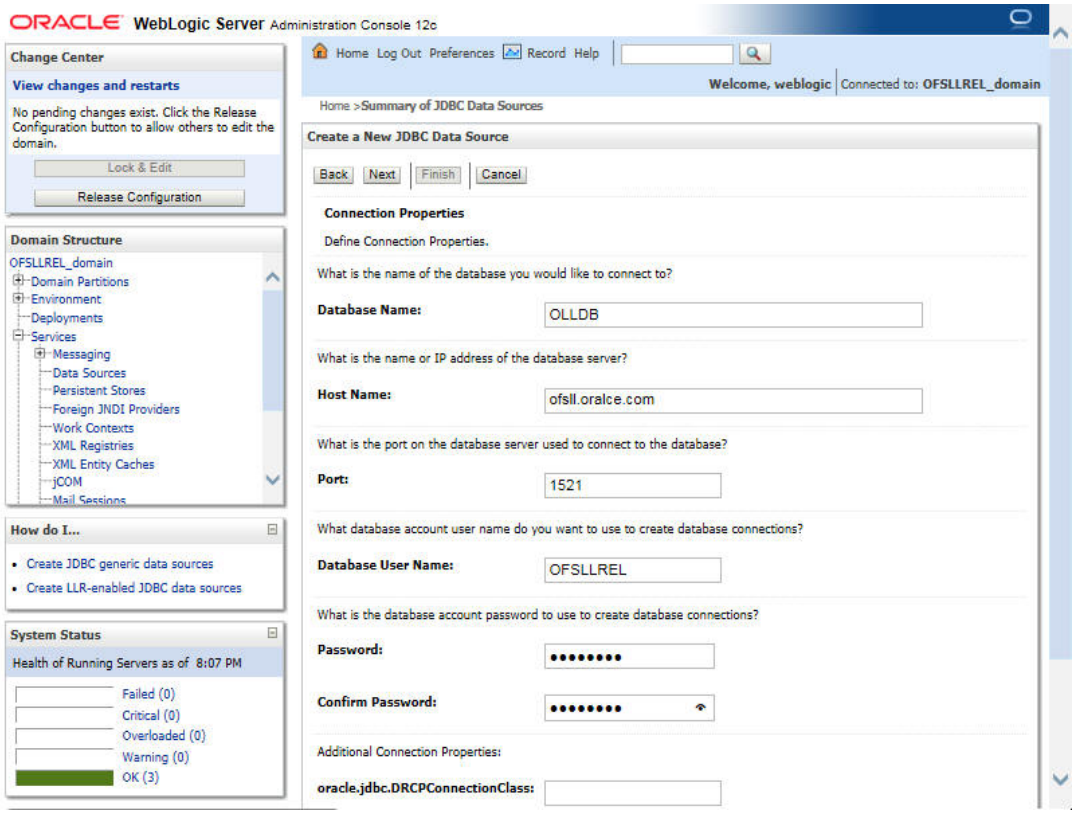
Figure 6-7 Create RESTful WebService 7



10. Click **Next**.

    The following window is displayed.

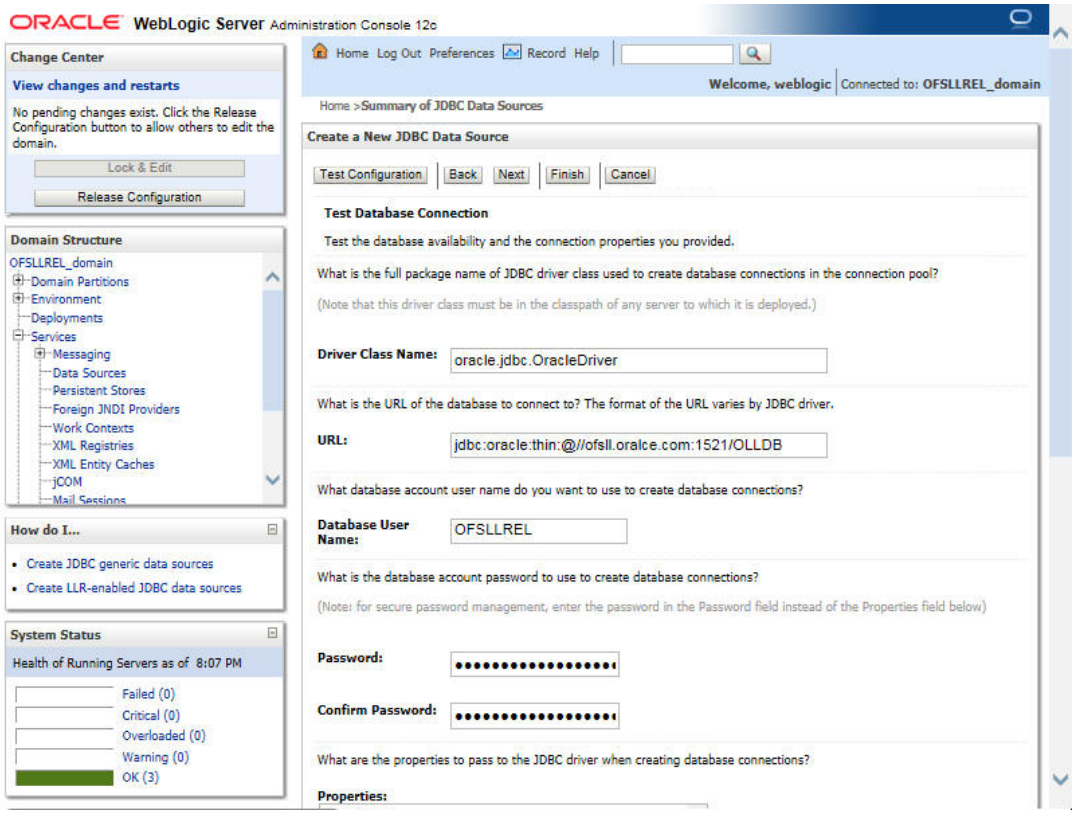
Figure 6-8 Create RESTful WebService 8



11. Enter the Database details.
12. Click **Next**.

    The following window is displayed.

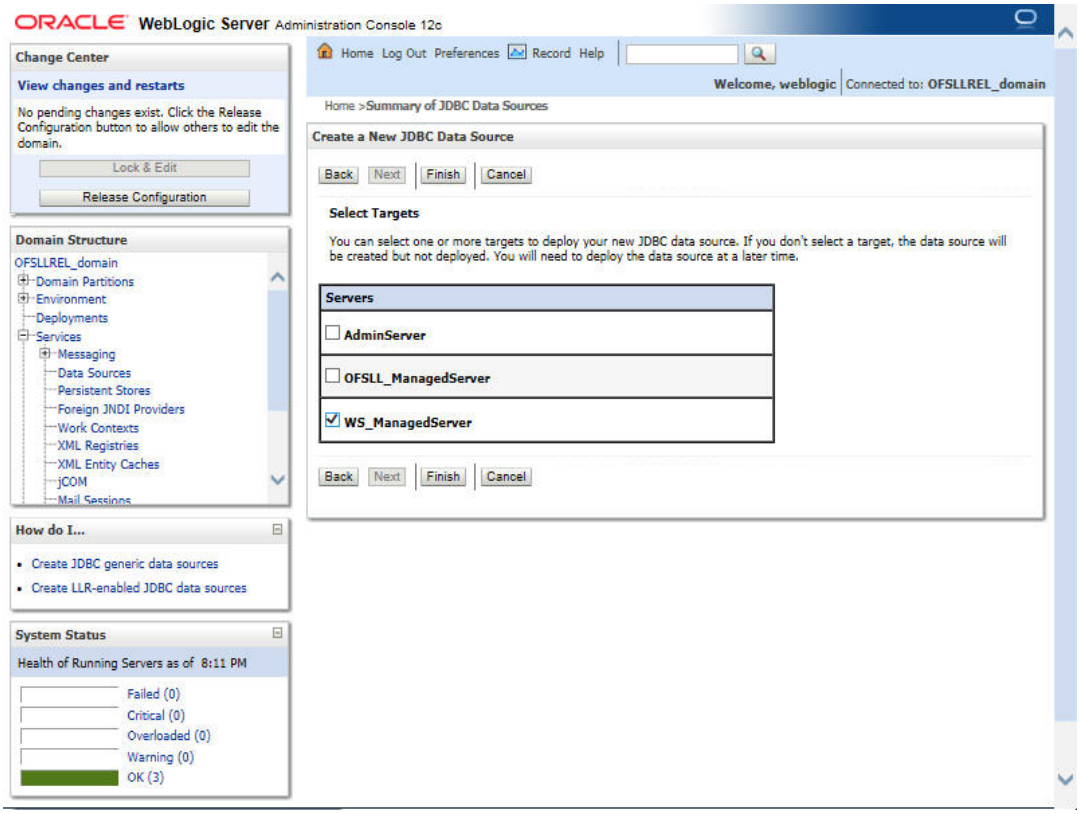
Figure 6-9 Create RESTful WebService 9



13. Click **Test Configuration**. On completion, displays a confirmation message as **Connection test succeeded**.
14. Click **Next**.

    The following window is displayed.

Figure 6-10 Create RESTful WebService 10



15. Select target Server as **WS_ManagedServer**.
16. Click Advanced button and update the **Inactive Connection Timeout** to 300 seconds.
17. Click **Finish** to activate the changes.

6.2 Statement Timeout Configuration

When APIs are integrated with Client systems, you may need to specify how long your client system waits for an API call to complete before a timeout occurs. If the Client system times out earlier than the API call, you may see inappropriate responses.

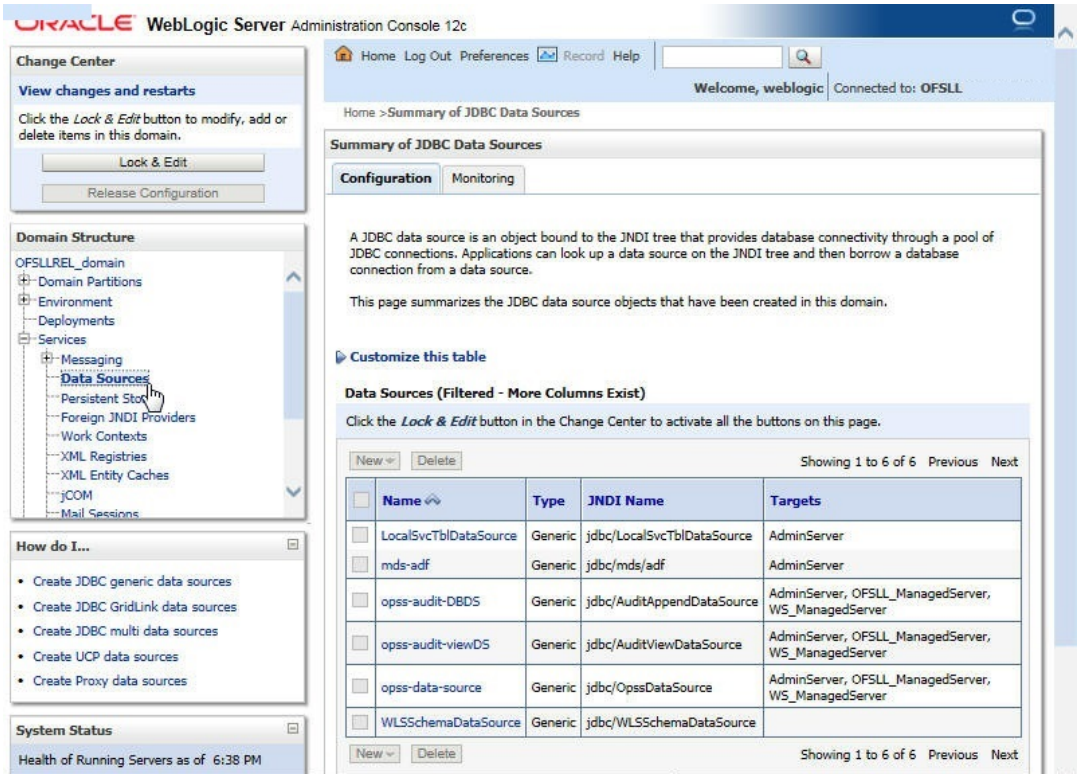
Hence, a client timeout value higher than the API response time is required to avoid such a situation.

Ensure that the time out settings is always defined in decreasing order. Which means, the time out value of managed components configured between the 'Client Server' and 'OFSLL Managed Servers' should have decreasing value so that last managed server before OFSLL has least timeout value.

Follow the below steps to set the statement timeout value.

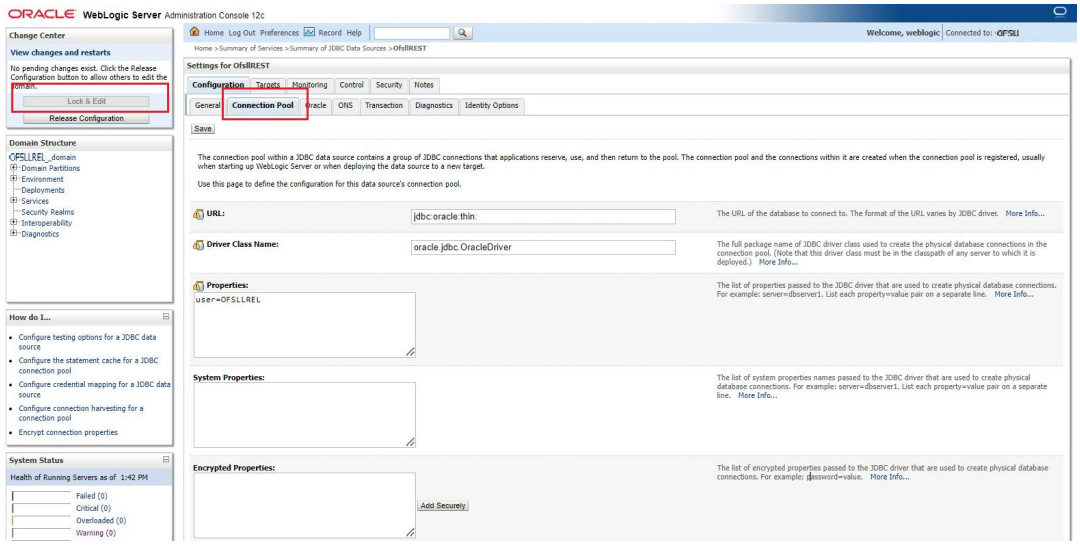
1. Login to WebLogic Server 12c console (http://hostname:port/console) using the valid credentials.
2. Click **Domain Name > Services**. The following screen is displayed:

Figure 6-11 JDBC Data Sources window



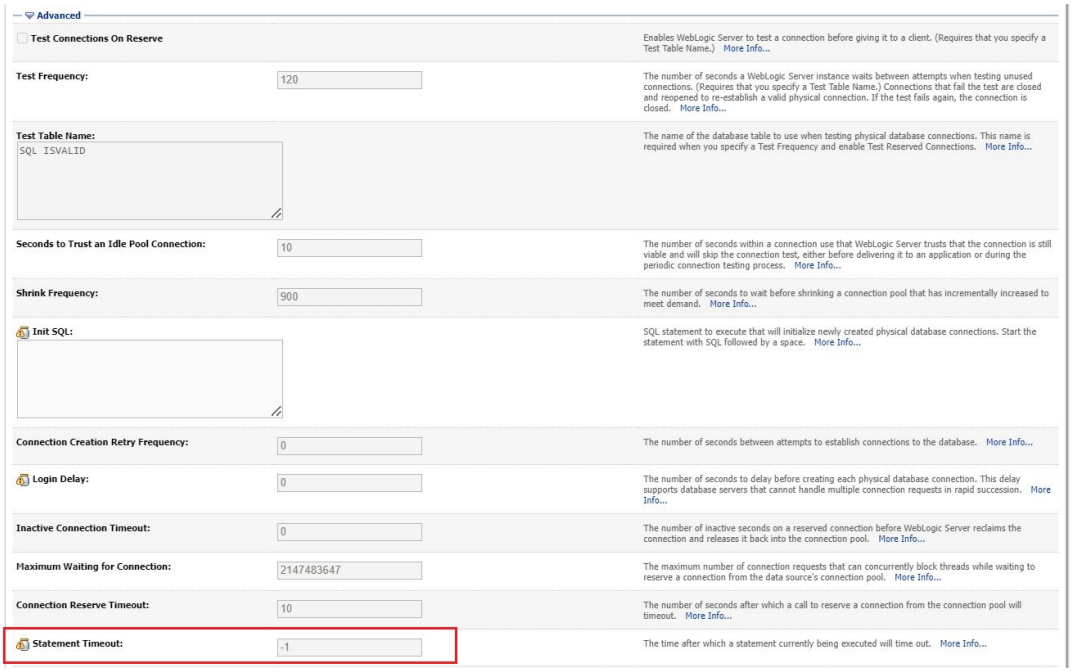
3. Click the **Data Source** from the LHS menu and click on the configured Restful data source (for example 'OfsllREST').
4. Click **Connection Pool** tab. The following screen is displayed.

Figure 6-12 Connection Pool tab



5. Click **Lock & Edit** option from the Change Center menu.
6. Scroll down and click the **Advance** option. The following screen is displayed.

Figure 6-13 Advance option - Statement Timeout



7. Set the **Statement Timeout** value as appropriate. This is the time after which a statement currently being executed will time out. For more information, refer to "**Guidelines_OFSLL API Timeout Period.pdf**" shared along with fix.
8. Once done, for changes to take effect, you need to restart the Data Source. Click on the **View changes and restarts** from the Change Center menu.

6.3 OAuth Implementation

(Optional) To extend OFSLL SaaS, OAuth2 can be used for securing OFSLL web services user access Authentication.

Web services authentication using OAuth2 is one of the best approach for securing user authentication to extend OFSLL SaaS. This uses Oracle / Non-Oracle PaaS to authenticate service access request from an external partner application without sharing OFSLL environment access credentials (UID / Password) and leverages the built-in support for OAuth 2.0.

OAuth 2.0 is an open standard token-exchange technology for verifying a user's identity across multiple systems and domains without risking the exposure of a password.

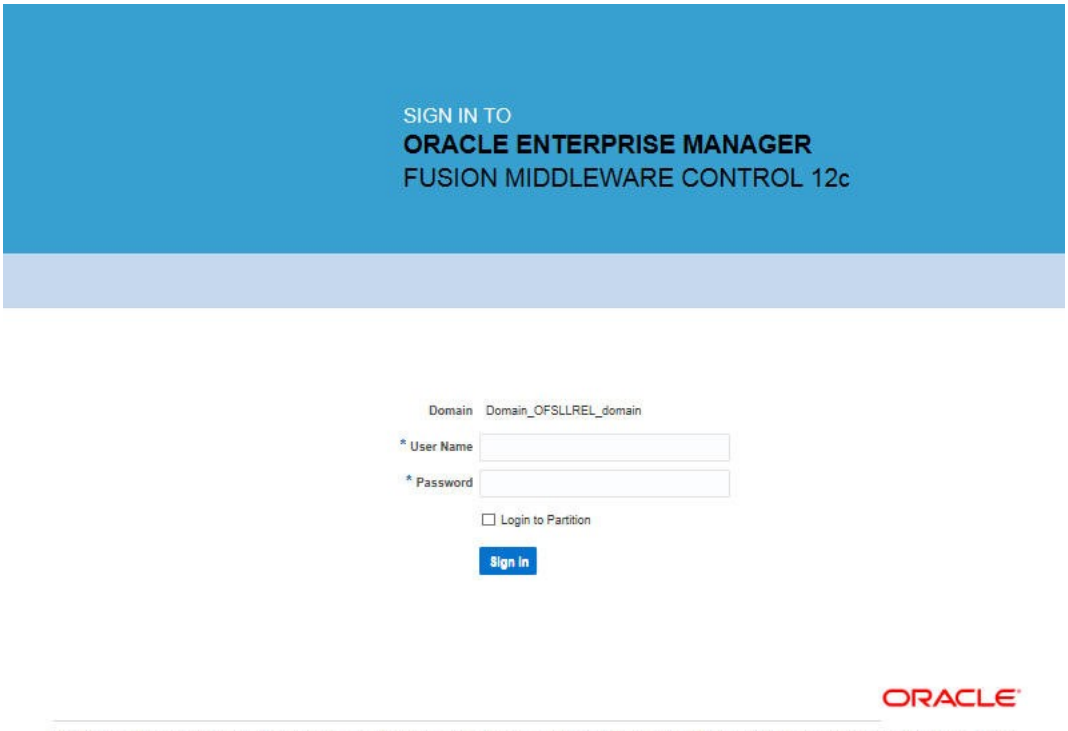
For detailed information, refer to the OAuth Implementation Guide shared in OTN library.

6.4 Deploying RESTful WebService

Please follow the below steps to deploy RESTful WebService.

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

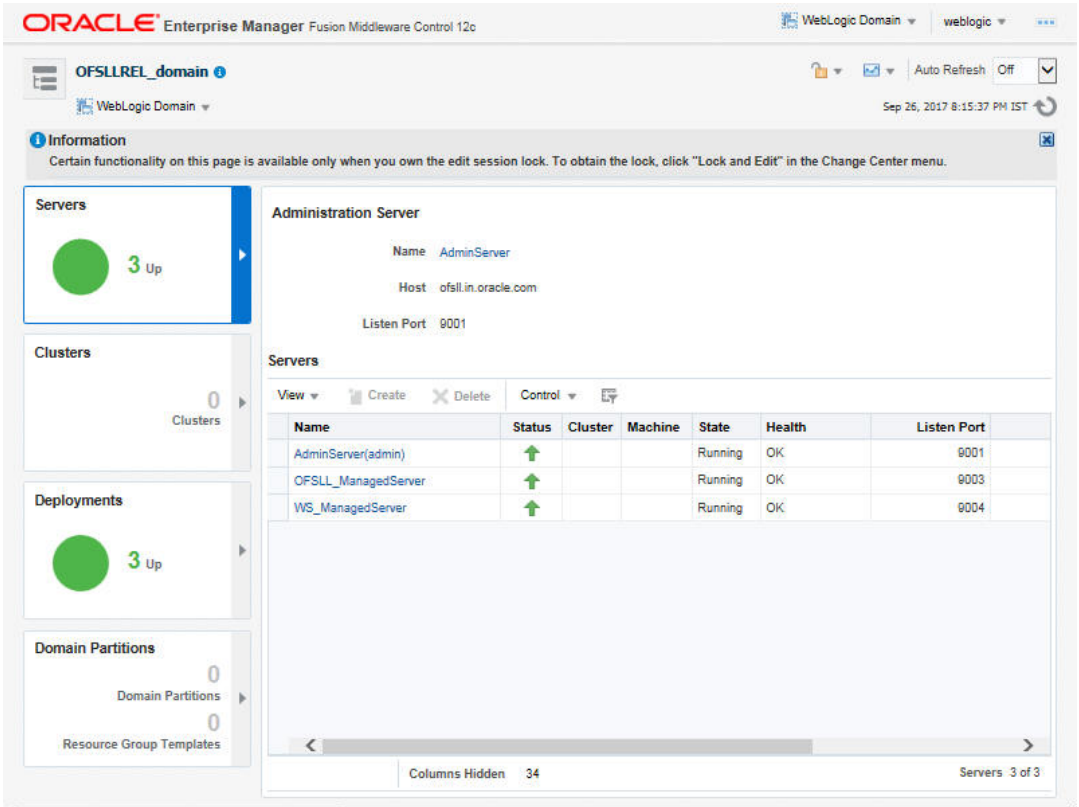
Figure 6-14 Deploy RESTful WebService 1



2. Enter valid login credentials.

   The following window is displayed.

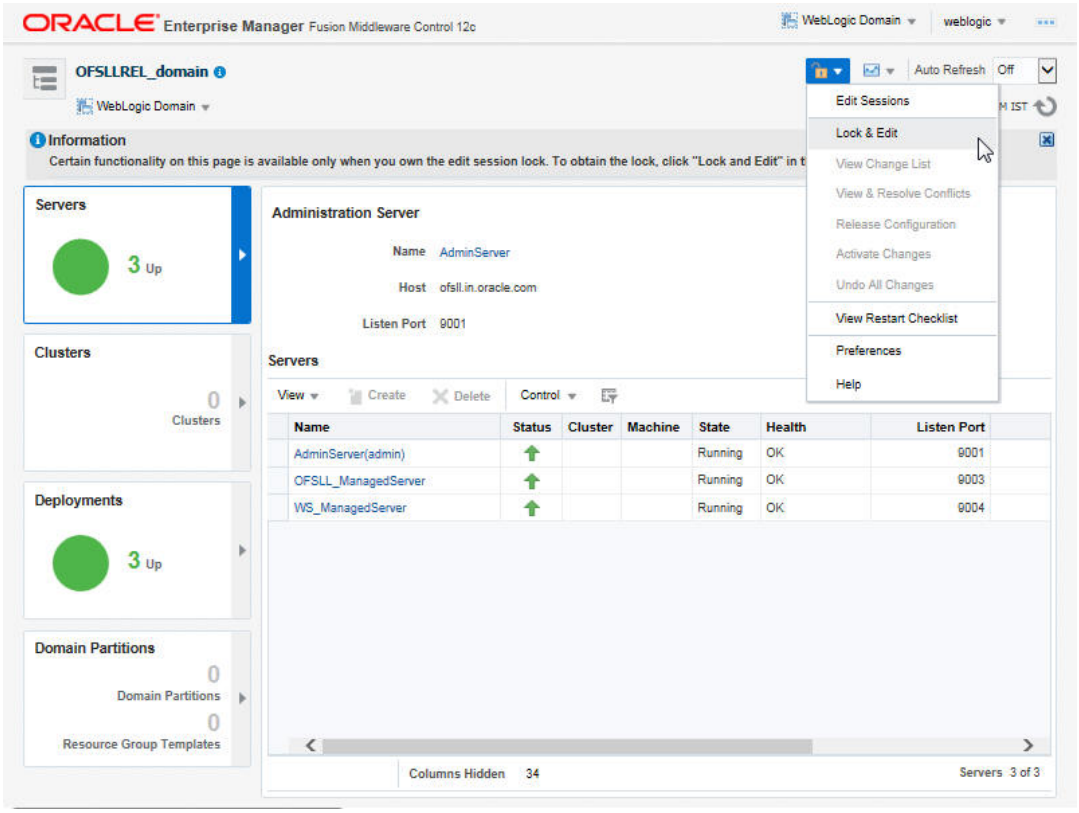
Figure 6-15 Deploy RESTful WebService 2



3. Select **Lock & Edit** option in the lock drop-down list available in the header.
4. Click **Deployment** in the left panel. Select **Lock & Edit**option in the lock drop-down list available in the header.

   The following window is displayed.

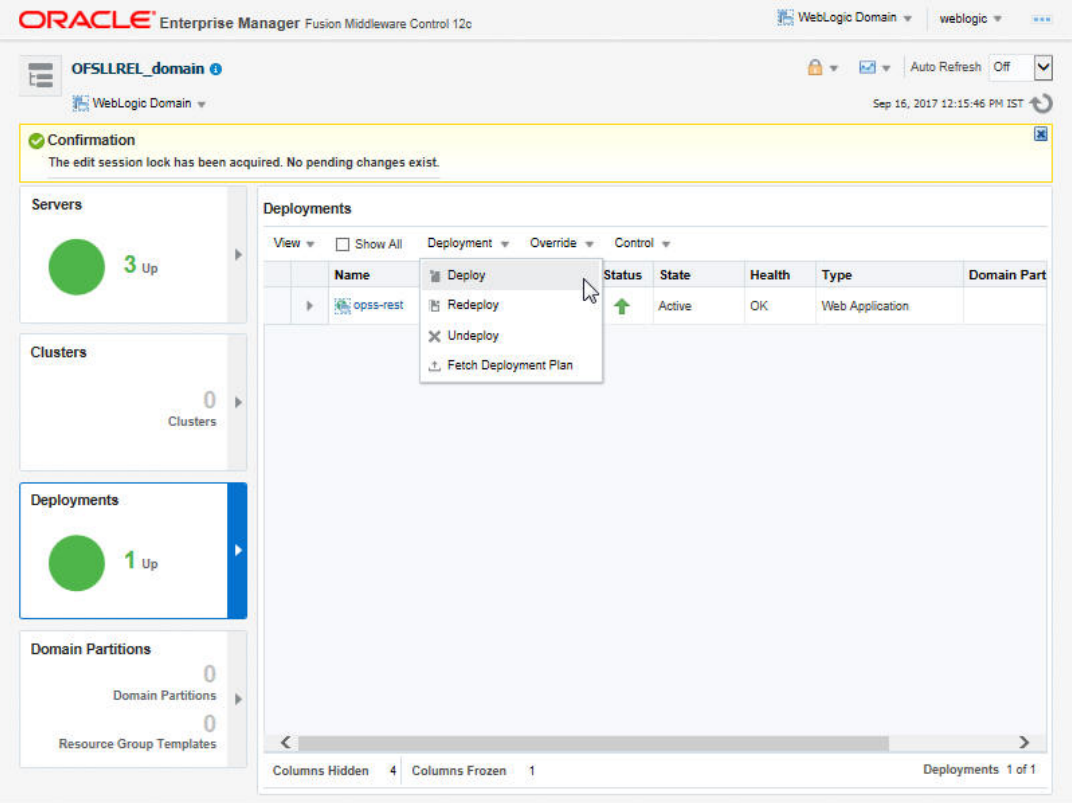
Figure 6-16 Deploy RESTful WebService 3



5. Select **Deploy** from the Deployment drop-down list.

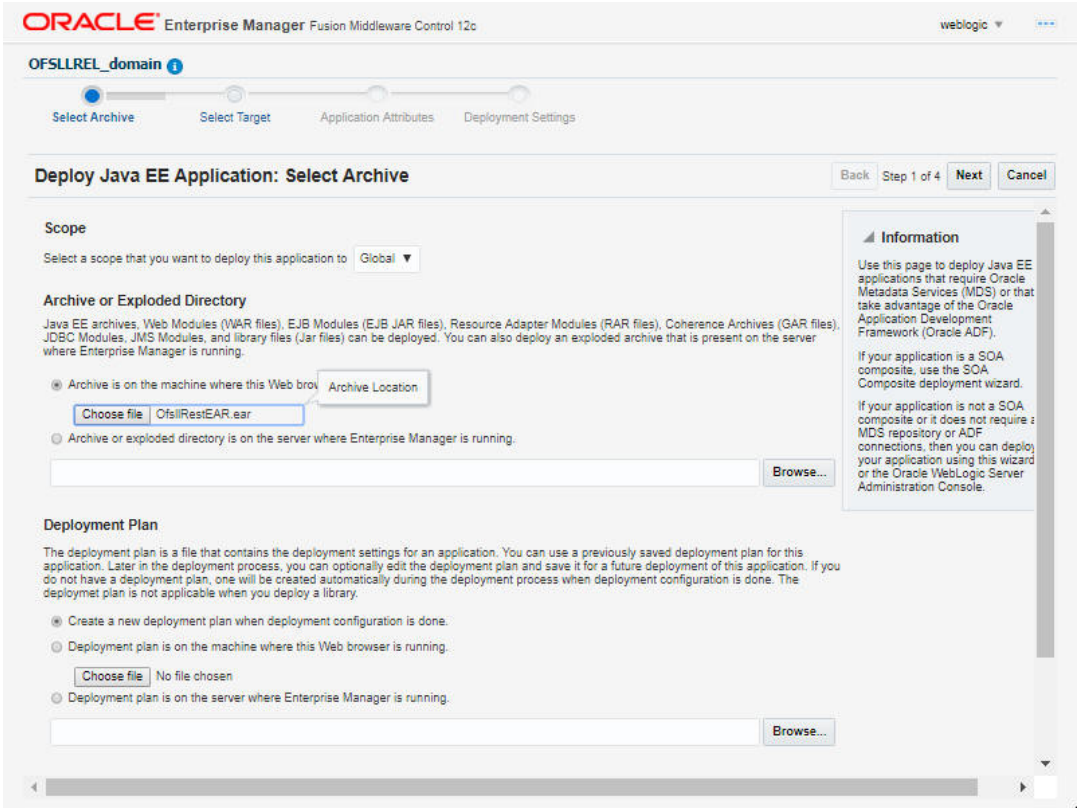
   The following window is displayed.

Figure 6-17 Deploy RESTful WebService 4



6. The following window is displayed.

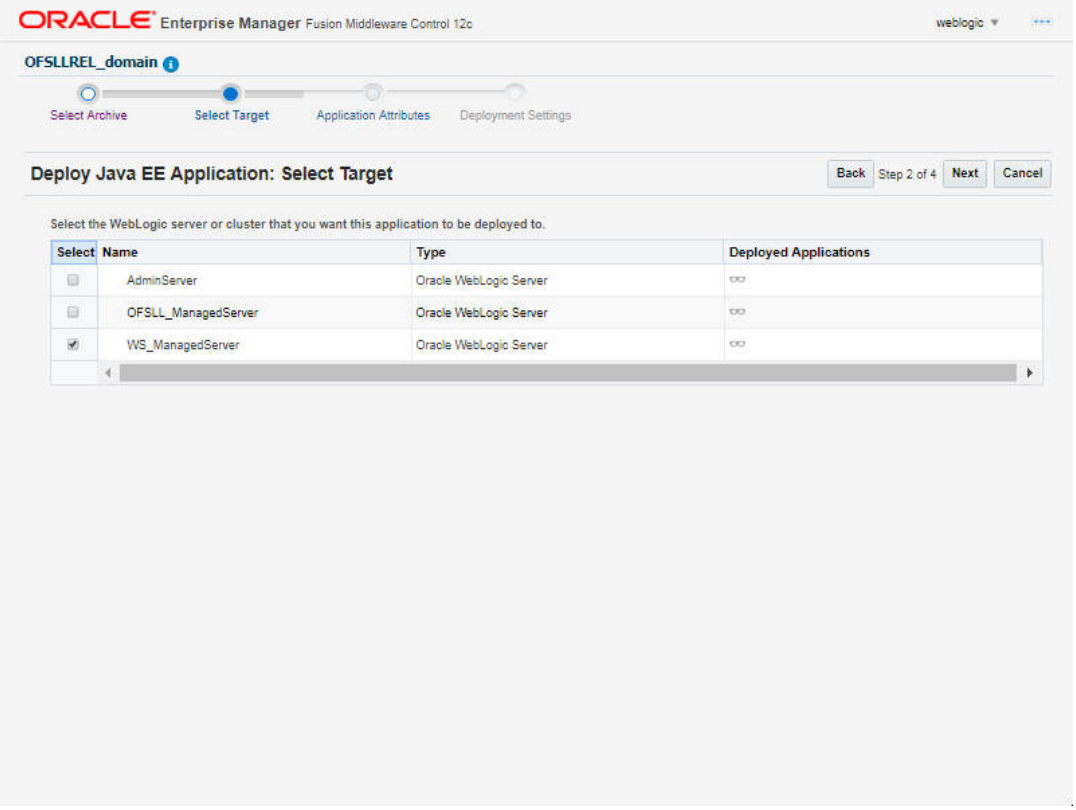
Figure 6-18 Deploy RESTful WebService 5



7. Browse to the folder containing the WebService. Eg: `C:/OfsllRestEAR.ear`
8. Click **Next**.

   The following window is displayed.

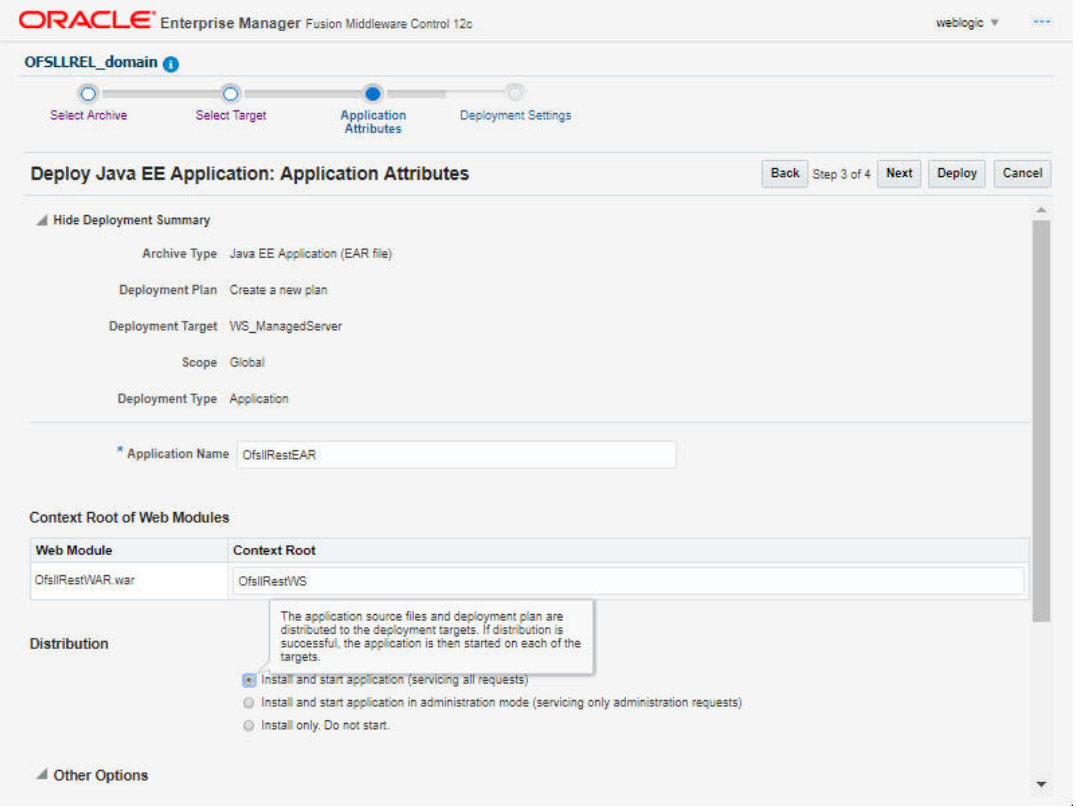
Figure 6-19 Deploy RESTful WebService 6



9. Select the server on which the WebService needs to be deployed.
10. Click **Next**.

    The following window is displayed.

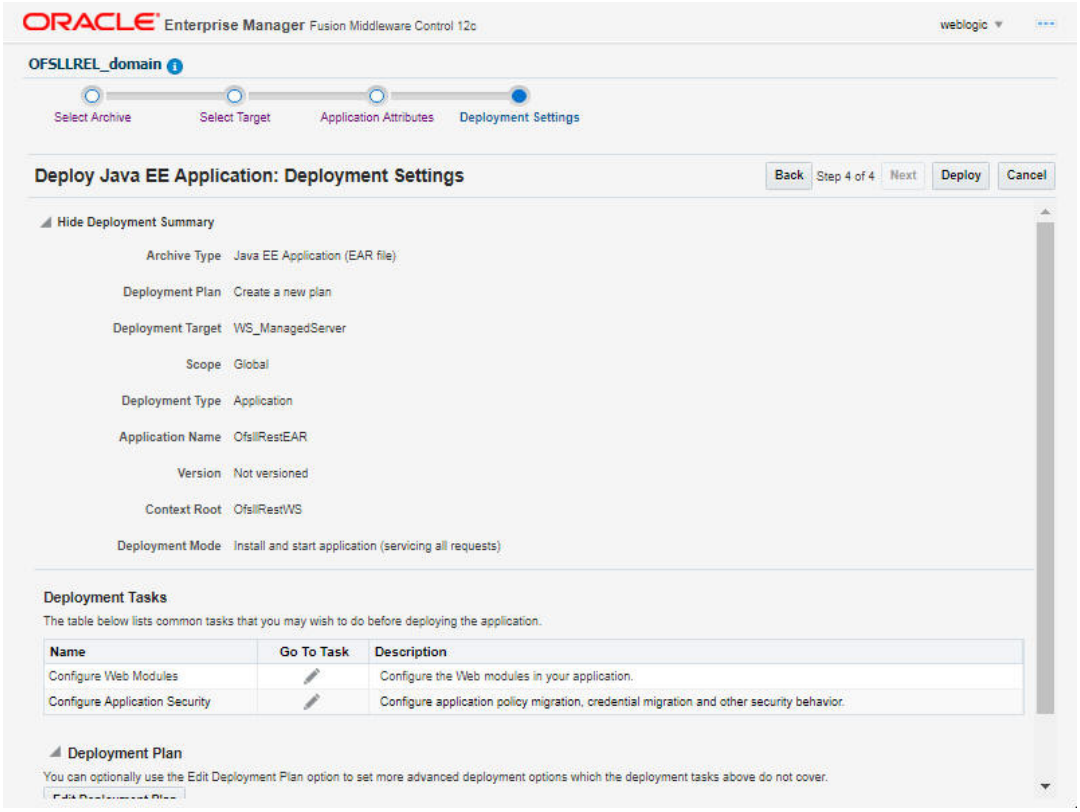
Figure 6-20 Deploy RESTful WebService 7



11. Select the option **Install and start application (servicing all requests)**.
12. Check the context root and click **Next**.

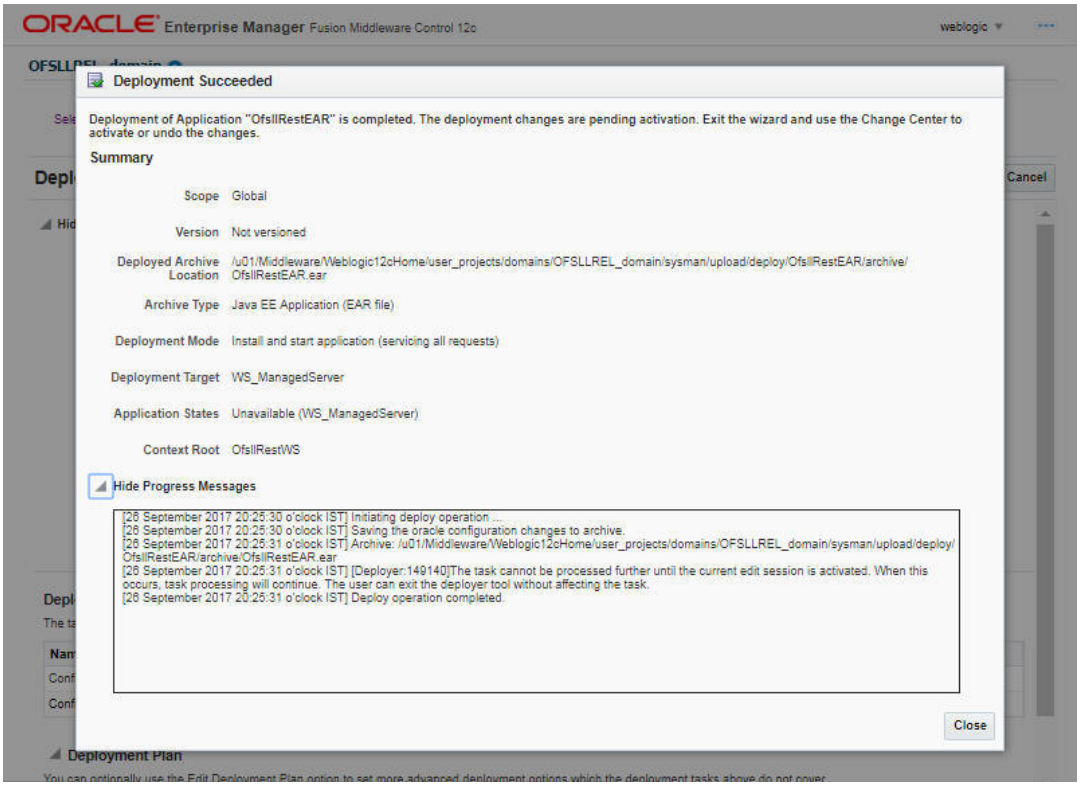
    The following window is displayed.

Figure 6-21 Deploy RESTful WebService 8



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

Figure 6-22 Deploy RESTful WebService 9



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.

The next step is to **Identifying the RESTful Webservice URL**.

- Identifying the RESTful Webservice URL

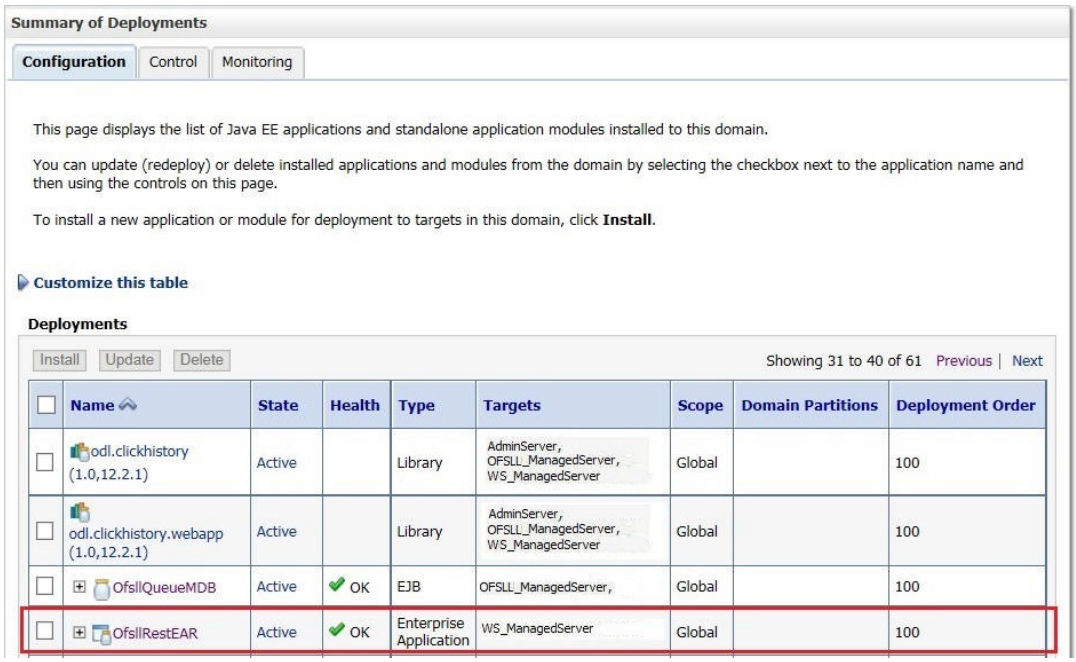
6.4.1 Identifying the RESTful Webservice URL

The following section briefs how 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.

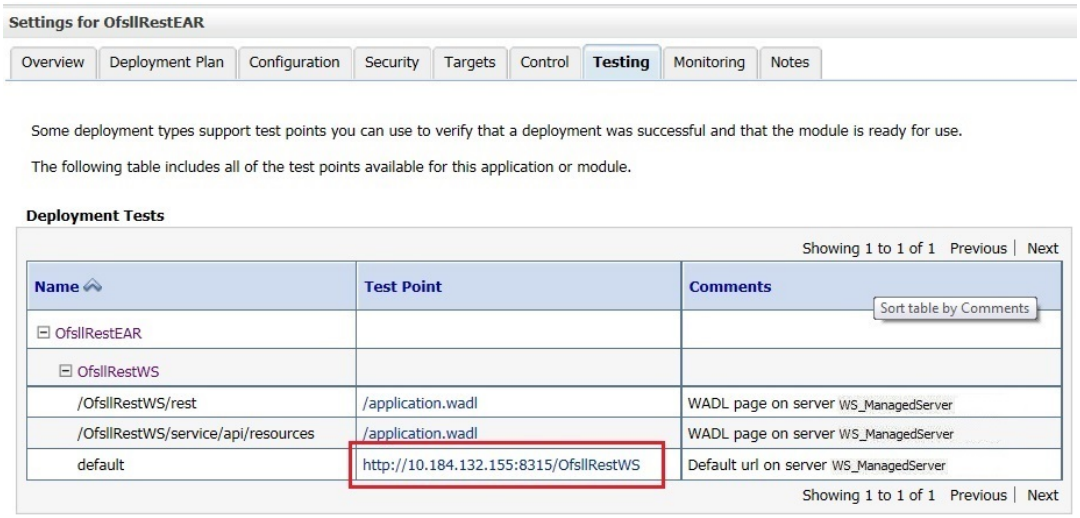
Figure 6-23 Identifying URL 1



3. Click **Testing** tab and expand **OfsllRestWS**.

   The following window is displayed.

Figure 6-24 Identifying URL 2



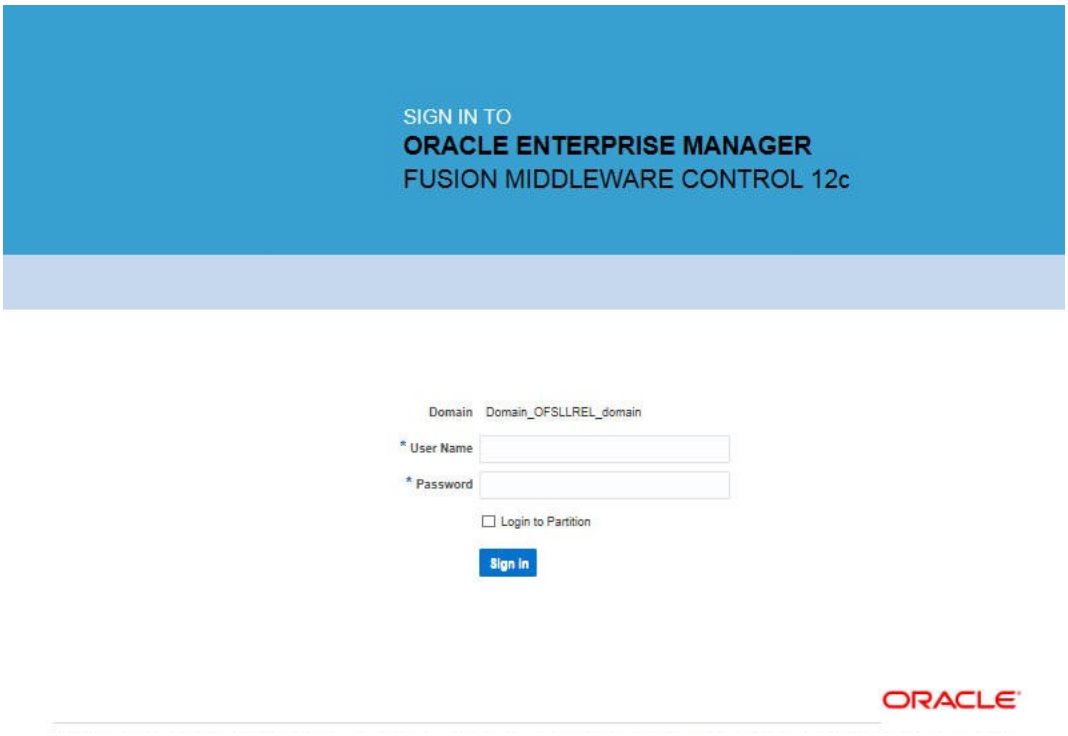
4. You can view the **OfsllRestful** Services URL as shown.
5. Swagger documentation for RESTful web services can be accessed using the following URL type - http://<server_name>;<port>/<Application_context>/swagger.json

6.5 Deploying RESTful Credit Bureau WebService

Please follow the below steps to deploy RESTful Credit Bureau WebService.

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

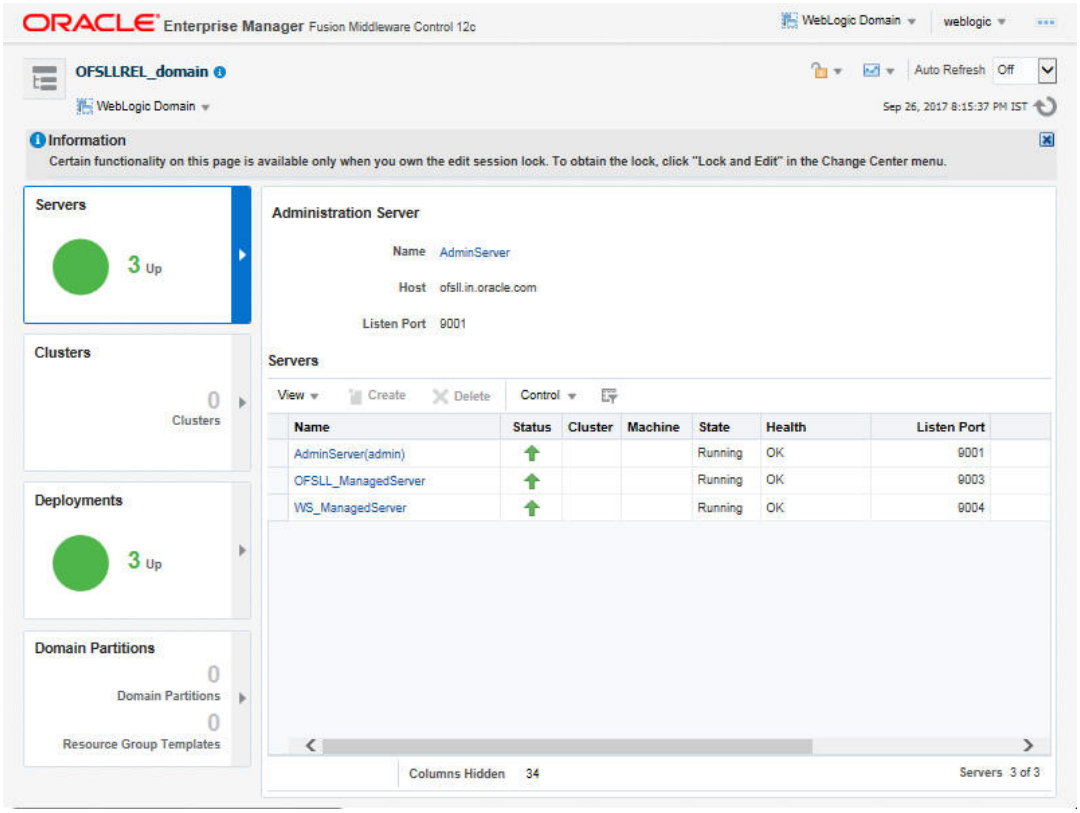
Figure 6-25 Deploy Bureau WebService 1



2. Enter valid login credentials.

   The following window is displayed.

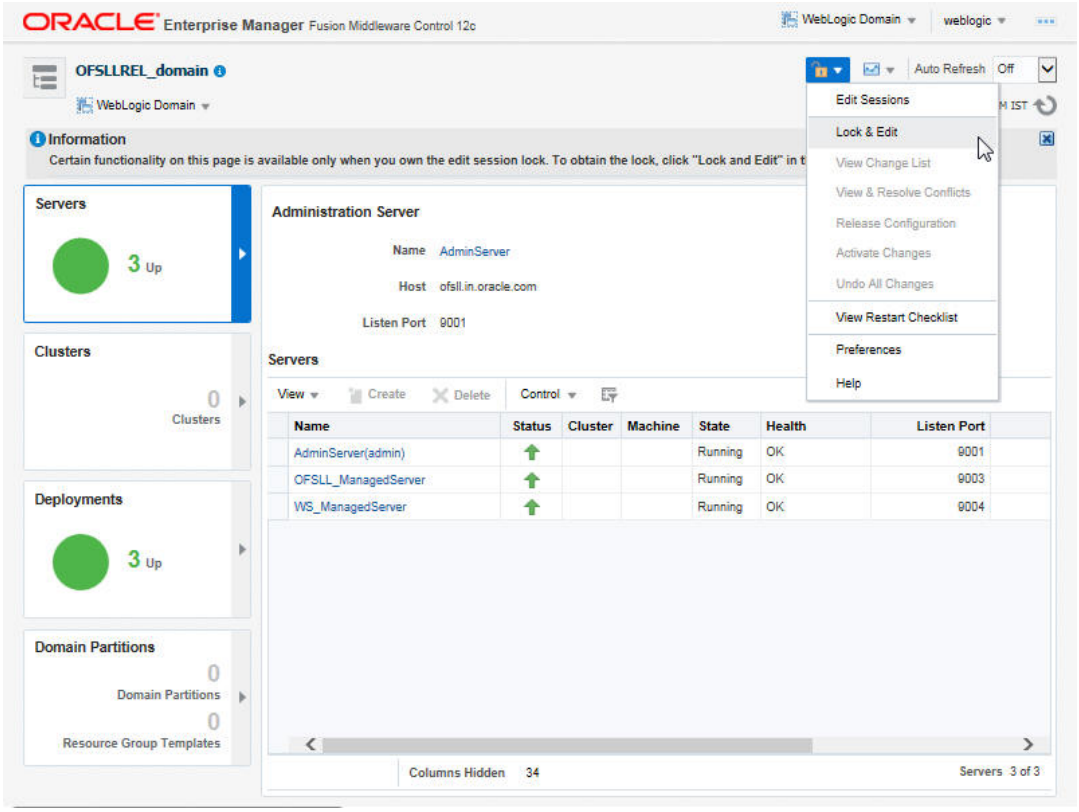
Figure 6-26 Deploy Bureau WebService 2



3. Select **Lock & Edit** option in the lock drop-down list available in the header.
4. Click **Deployment** in the left panel.

   The following window is displayed.

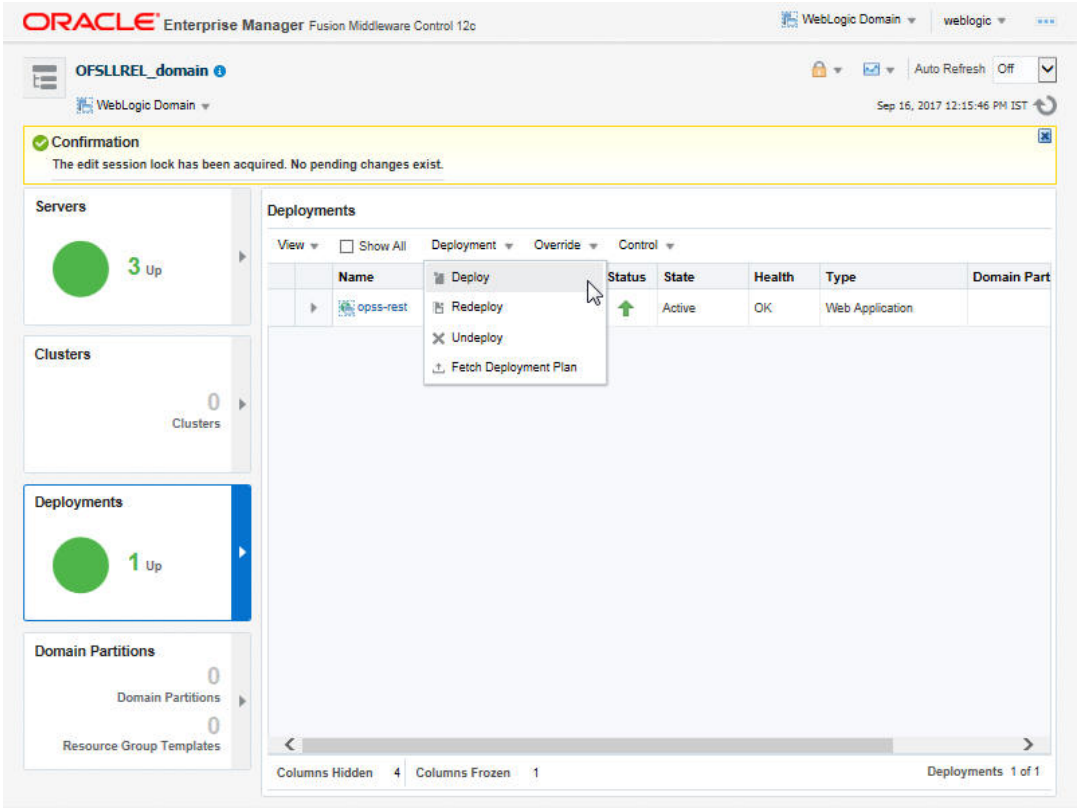
Figure 6-27 Deploy Bureau WebService 3



5. Select **Deploy** from the Deployment drop-down list.

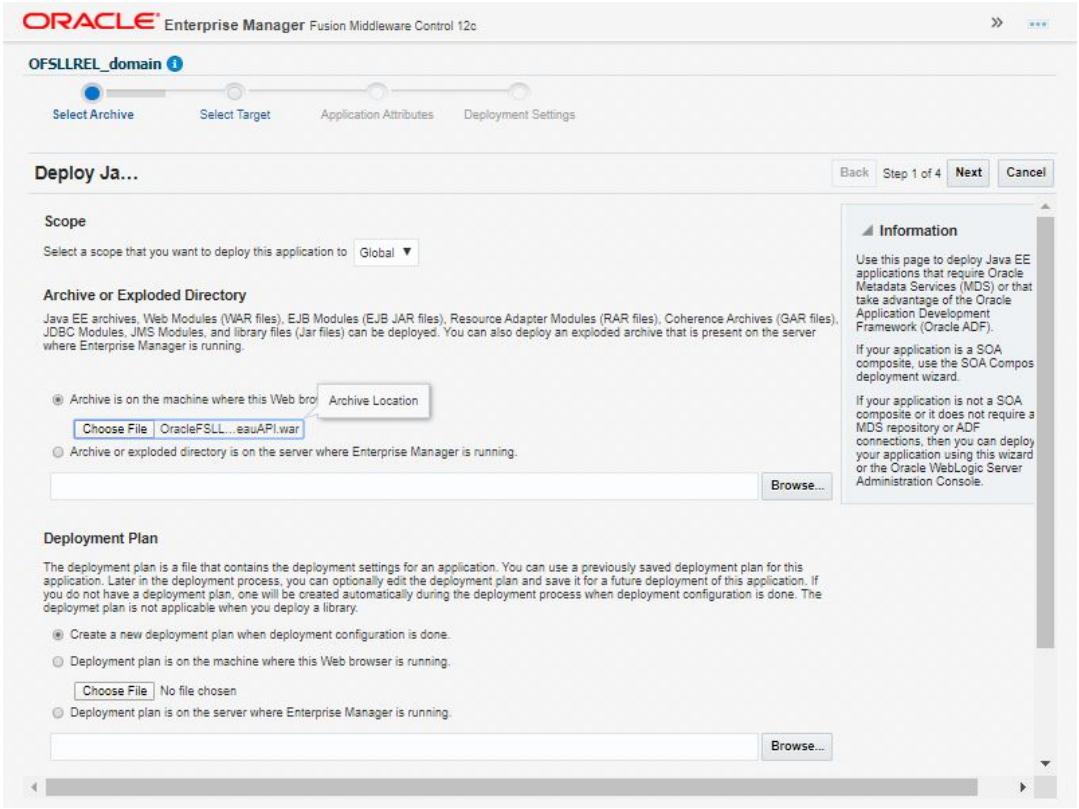
   The following window is displayed.

Figure 6-28 Deploy Bureau WebService 4



6. The following window is displayed.

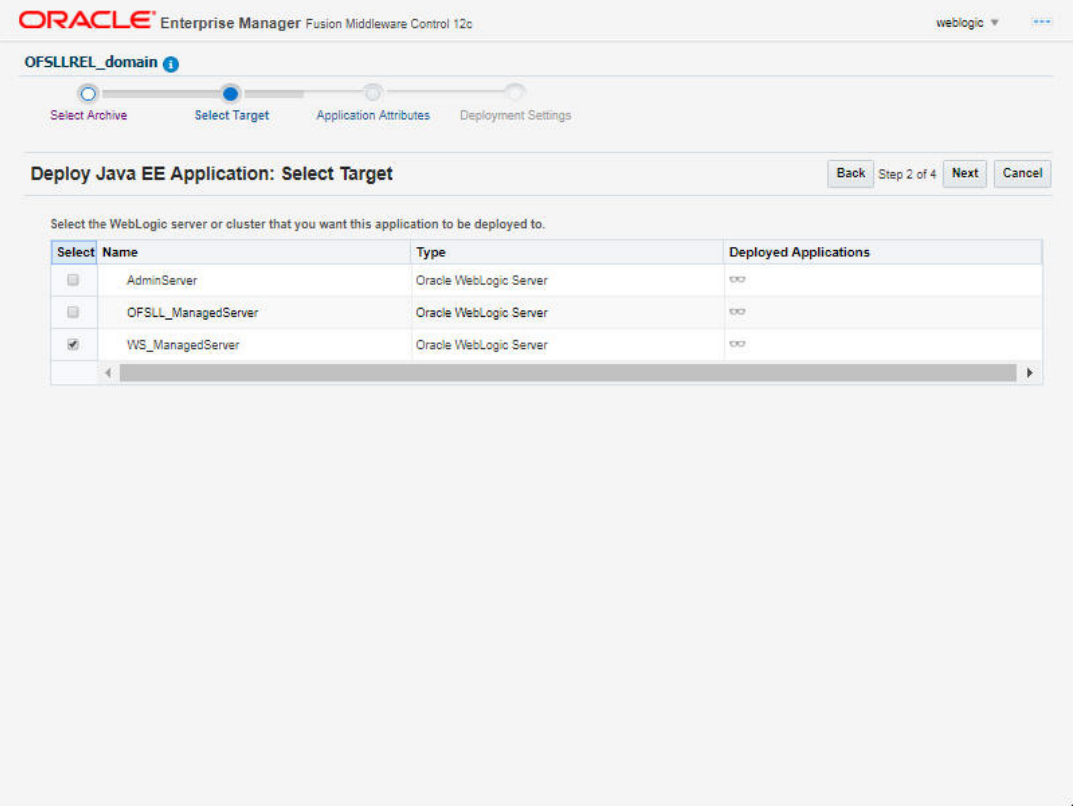
Figure 6-29 Deploy Bureau WebService 5



7. Browse to the folder containing the Credit Bureau WebService. For example: `C:/OracleFSLLBureauAPI.war`
8. Click **Next**.

   The following window is displayed.

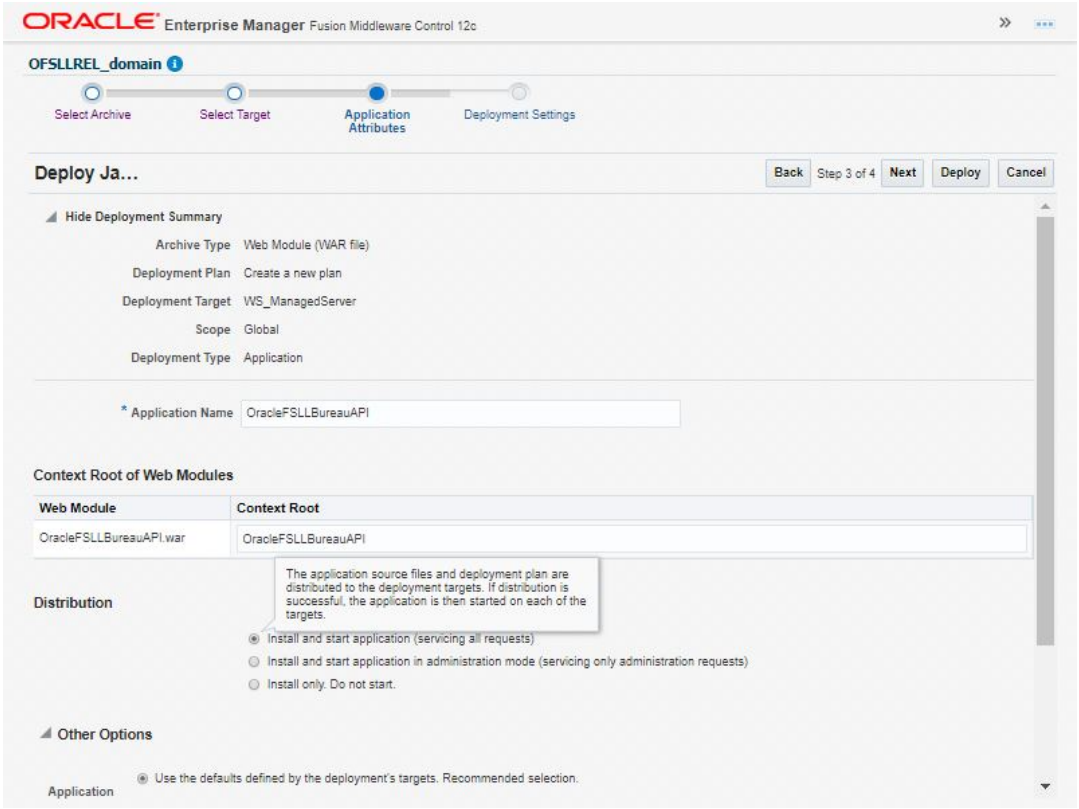
Figure 6-30 Deploy Bureau WebService 6



9. Select the server on which the WebService needs to be deployed.
10. Click **Next**.

    The following window is displayed.

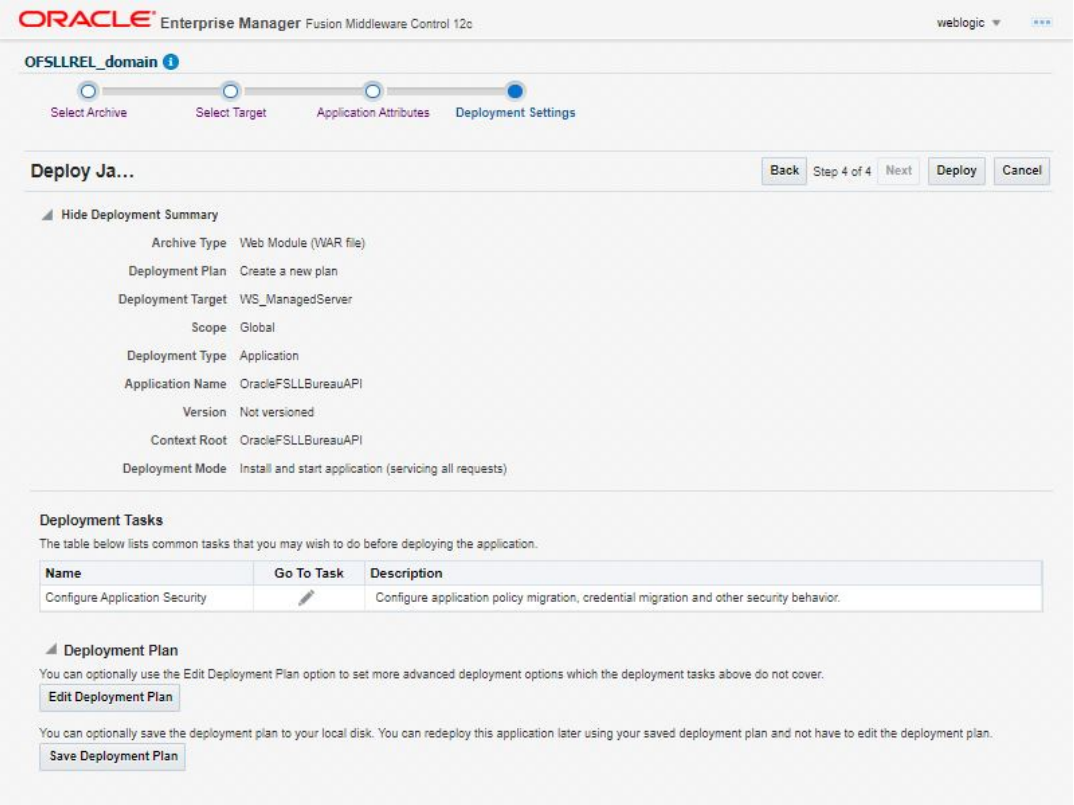
Figure 6-31 Deploy Bureau WebService 7



11. Select the option **Install and start application (servicing all requests)**.
12. Check the context root and click **Next**.

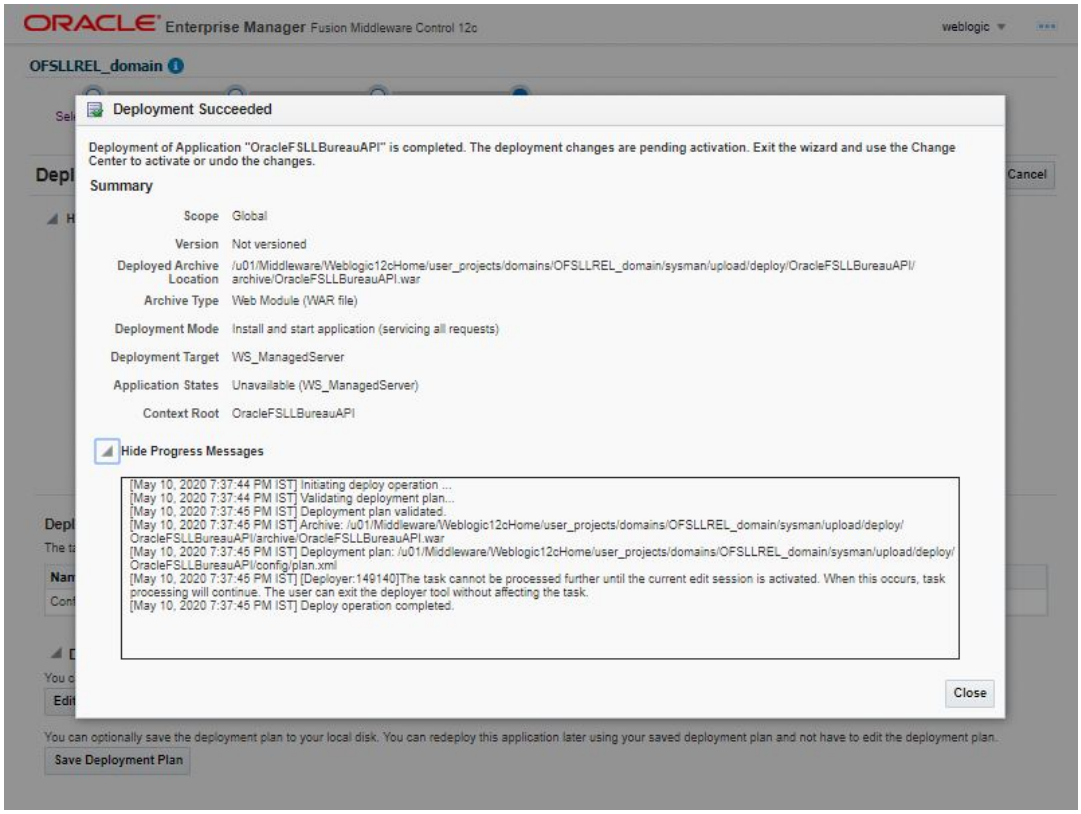
    The following window is displayed.

Figure 6-32 Deploy Bureau WebService 8



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

Figure 6-33 Deploy Bureau WebService 9



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.

The next step is to **Creating Credentials and System Policies for Credit Bureau Interface**.

- Creating Credentials and System Policies for Credit Bureau Interface

6.5.1 Creating Credentials and System Policies for Credit Bureau Interface

In order Configure Credit Bureau interface, 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.

Create the following Maps and corresponding keys as indicated in following table.

Table 6-1 Maps and corresponding keys

Maps	Keys	Description
ofsll.int.bureau	creditbureau_auth_mode_adapter	There are two modes: BASIC - On selecting this option, you need to define User Name and Password to authenticate. OAUTH2.0 - On selecting this option, you need to define additional enabled fields such as Grant Type, Client Id, Client Secret, Identity Domain, Token and Header Key.

Table 6-1 (Cont.) Maps and corresponding keys

Maps	Keys	Description
	creditbureau_adapter	If Authentication mode is selected as BASIC, specify the Basic Authentication User Name and Password.
	creditbureau_serviceurl_adapter	BureauApi or Third party RestAPI end point url.
	ProxyAuthenticationEnabled	Indicator used to validate proxy.
	ProxyEnabled	Indicator is for whether Proxy server info need to be set or not.
	ProxyPort	Port to which ProxyServer is running.
	ProxyServer	Name of the proxyServer to be configured
	<Bureau_name>_cert_path	The location of certificate file which contains the valid certificate for Credit Bureau.
	<Bureau_name>_cert_password	The password that requires to read the valid certificate for the Credit Bureau.
	<Bureau Name>_consumer_code	Consumer credentials to be configured for request creation of third party.
	<Bureau Name>_token_indicator	Indicator used for whether third party token request needs to be create or not.
	<Bureau Name>_cert_-check_indicator	Indicator used for whether certificate validation is required or not.

Note:

For certificate creation, please refer to **Interface_Certificate_Configuration.pdf** document available in the release bundle.

7

Appendix A :Configuration parameters

Refer to the following section for details on configuration parameters.

Modifying Configuration Files

Table 7-1 Route One Configuration

Property Name	Property Value	Description	Remarks
Configuration File: ro_servlet_init.conf			
LOSApplicationRequest-ServiceURL	http://<HOST NAME>:<PORT>/ <CONTEXT ROOT>/ LOSApplicationRequest Service	URL for OFSLL New application WebService	Refer the recommended context root table for CONTEXT ROOT. Sample URL: http://host-name:port/dbkls-xws-app-ro/LOSApplication-RequestService
LOSApplicationUpdate-ServiceURL	http://<HOST NAME>:<PORT>/ <CONTEXT ROOT>/ LOSAppli-cationUpdateService	URL for OFSLL application update WebService	Refer the recommended context root table for CONTEXT ROOT. Sample URL: http://host-name:port/dbkls-xws-app-ro/LOSApplica-tionUpdateService
LOSeContractWebSer-viceServiceURL	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://host-name:port/dbkls-xws-app-ro/LOSEContract-Service
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://messag-ing.itl.routeone.net/ IF1_ITL/ CASSBMessagingRecei ver	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.

Table 7-1 (Cont.) Route One Configuration

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	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	ROUTEONE	Parameter to set application source	Don't modify this 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	Needed only for RouteOne
jndiLookupDataSource	jdbc/ IN1HukWznG0b4esj	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 Sig-nature	The keystore location should not be changed and keystore should be available in this location
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.	

Table 7-1 (Cont.) Route One Configuration

Property Name	Property Value	Description	Remarks
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	
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	setme	The keystore password	The deafult password is "setme". 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		

Table 7-1 (Cont.) Route One Configuration

Property Name	Property Value	Description	Remarks
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

Table 7-2 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://host-name: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://host-name: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"	

Table 7-2 (Cont.) Dealer Track

Property Name	Property Value	Description	Remarks
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
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
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	setme	The keystore password	The default password is "setme". 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.	

Table 7-2 (Cont.) Dealer Track

Property Name	Property Value	Description	Remarks
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	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	setme	The keystore password	The deafult password is "setme". 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		

Table 7-2 (Cont.) Dealer Track

Property Name	Property Value	Description	Remarks
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

Table 7-3 EDOCS

Property Name	Property Value	Description	Remarks
Configuration File: ds_servlet_init.conf			
LOSeApplicationRequestServiceURL	http://<HOST NAME>:<PORT>/<CONTEXT ROOT>/LOSeApplicationRequestService	URL for OFSLL eDocs Create/update application WebService	Refer the recommended context root table for CONTEXT ROOT. Sample URL: http://host-name:port/dbkls-xws-app-ds/LOSeApplicationRequestService
LOSApplicationCommentUpdateServiceURL	http://<HOST NAME>:<PORT>/<CONTEXT ROOT>/LOSApplicationCommentUpdateService	URL for OFSLL eDocs comment update WebService	Refer the recommended context root table for CONTEXT ROOT. Sample URL: http://host-name:port/dbkls-xws-app-ds/LOSApplicationCommentUpdateService
LOSApplicationLocationUpdateServiceURL	http://<HOST NAME>:<PORT>/<CONTEXT ROOT>/LOSApplicationLocationUpdateService	URL for OFSLL eDocs location update WebService	Refer the recommended context root table for CONTEXT ROOT. Sample URL: http://host-name:port/dbkls-xws-app-ds/LOSApplicationLocationUpdateService
postTimeout	10	Parameter to set timeout value to post XML message to third party.	Timeout value should be specified in number of seconds.

Table 7-3 (Cont.) EDOCS

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	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	EDOCS	Parameter to set application source	Don't modify this 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
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	setme	The keystore password	The deafult password is "setme". 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"	

Table 7-3 (Cont.) EDOCS

Property Name	Property Value	Description	Remarks
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
fileLocation	/tmp		The directory for temporary files. Make sure that such directory exists on the deployment server
fileDelimiter	,	Delimiter used in dealer header file	
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	setme	The keystore password	The deafult password is "setme". 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		

Table 7-3 (Cont.) EDOCS

Property Name	Property Value	Description	Remarks
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

Table 7-4 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 DEALERTRACK
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

Table 7-4 (Cont.) System Parameters to be configured

System Parameter Name	System Parameter Desc
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 PASSWORD
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
XWS_XPR_INCLUDE_TEMP	INCLUDE TEMP PRODUCERS
XWS_XPR_ROUTEONE_RESP_PWD	ROUTE ONE DEALER LOAD PASSWORD
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_DLR_TRACK_LENDER_NAME	DEALERTRACK LENDER NAME
XWS_XAE_ROUTEONE_LENDER_NAME	ROUTEONE LENDER NAME
XWS_XAE_ECON_ROUTEEONE_RESP_URL	Route One E-contract response URL