Use Oracle JDeveloper to Deploy Applications

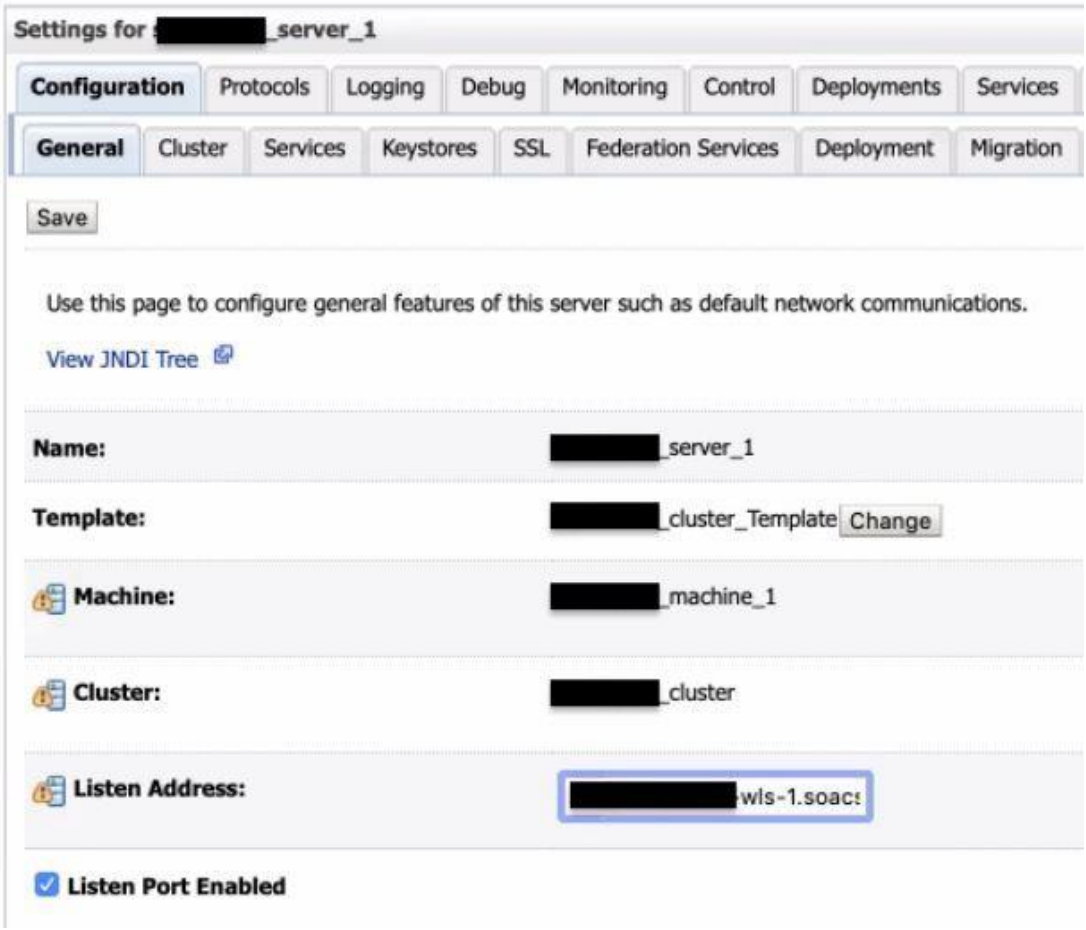
You can use Oracle JDeveloper to deploy SOA composite applications and Oracle Service Bus applications to an Oracle SOA Suite on Marketplace instance.

Add an Ingress Rule to Allow the JDeveloper Connection

After provisioning the Oracle SOA Suite on Marketplace instance, you must set up your JDeveloper environment before you can use it to deploy applications.

To set up JDeveloper for deploying to Oracle SOA Suite on Marketplace:

1. Make a note of the public IP address (or addresses in the case of a multinode cluster) associated with each SOA server.
2. Log in to the WebLogic Server Administration Console.
3. On the Summary of Servers page, click each Managed Server name and make a note of the **Listen Address** value:

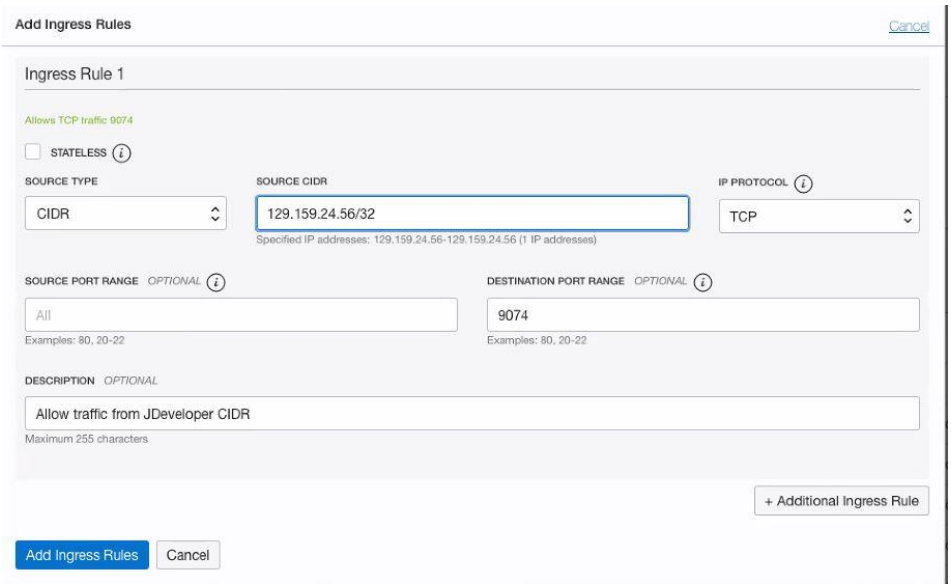


Be sure to capture the listen addresses for all Managed Servers.

4. On the host on which JDeveloper is running, map the listen address of each Managed Server to the associated SOA server public IP address in the hosts file. For Windows, the hosts file is typically located at C:\Windows\System32\Drivers\etc\hosts. For example:

```
129.146.136.141 sobdemo-wls-1.soacsp2pubsubne.soacsp2vcn.oraclevcn.com
158.101.23.141  sobdemo-wls-2.soacsp2pubsubne.soacsp2vcn.oraclevcn.com
129.146.136.141 sobdemo-wls-1
158.101.23.141  sobdemo-wls-2
```

5. Add the ingress rule to permit traffic from JDeveloper to the SSL listener port of the Managed Server:
    a. Sign in to the Oracle Cloud Infrastructure Console.
    b. Open the navigation menu, click **Networking**, and then click **Virtual Cloud Networks**.
    c. Select the compartment where you created the new instance.
    d. In the list of VCNs, select your VCN.
    e. On the Virtual Cloud Network Details page, click **Security Lists** in the left pane.
    f. Select a security list, and click **Add Ingress Rules** to open the Add Ingress Rules dialog.
    g. In the Add Ingress Rules dialog, create an ingress rule for port 9074 to access JDeveloper as shown in the following screenshot:



Note:

The source CIDR is the CIDR of the machine where JDeveloper is running.

h. Add another ingress rule for port 9072, with the same source CIDR as port 9074.

   Important:

   By adding this ingress rule, be aware that you are allowing traffic from the internet (known CIDRs) into WebLogic Server. You must be extra cautious and open traffic to known CIDRs only.

Create an Application Server Connection in JDeveloper

To create a new application server connection in JDeveloper:

1. Before you test the connection, clear your JDeveloper cache:
   a. In JDeveloper, click the **Help** menu and select **About**.
   b. In the About dialog, on the Properties tab, find ide.user.dir and note its value, which is the name of the cache directory.
   c. Back up the cache directory, then delete it.

      Note:

      All JDeveloper database connections and integrated WebLogic Server settings are lost when you delete the cache.
2. Add a new certificate to your JDeveloper for the secure WebLogic connection.

   Note:

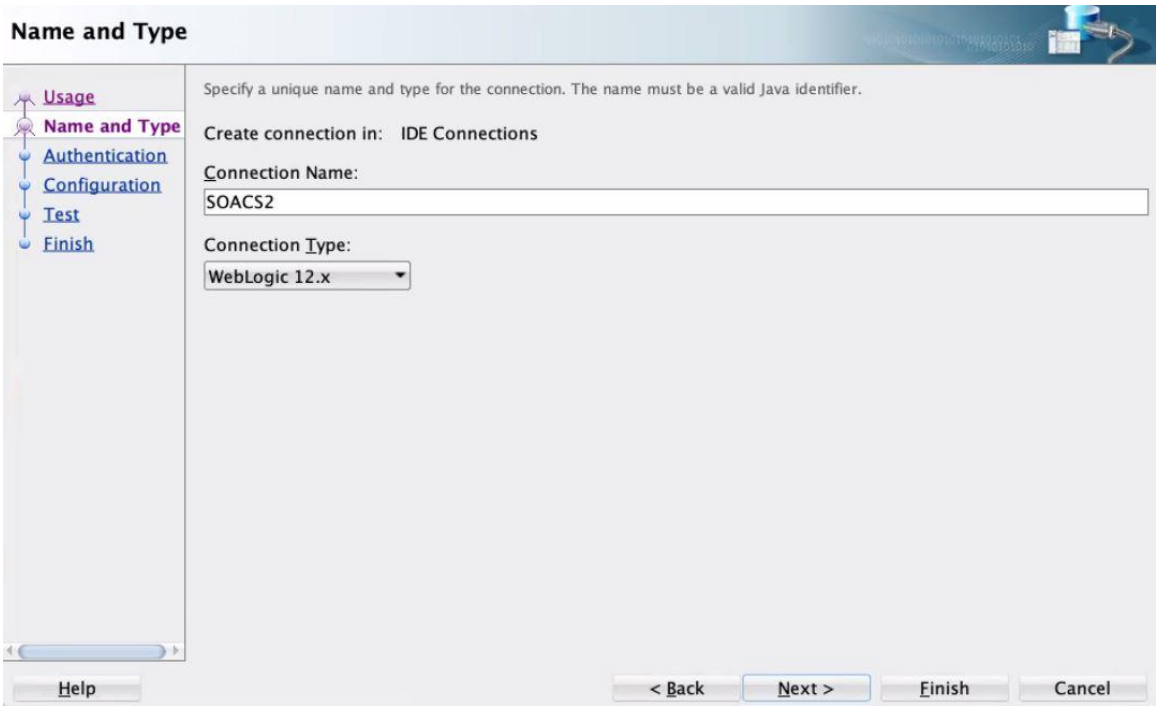
   Perform this step if you have not used the default certificate for secure WebLogic connection.
   a. In JDeveloper, navigate to **Tools**, and click **Preferences**.
   b. Under **Credentials**, select **Keystores**.
   c. In the Keystores page, specify the following values:
      - **Keystore Password**: Specify the demo trust keystore passphrase. The default password is DemoTrustKeyStorePassPhrase.
      - **Certificate Location**: Click **Browse** to find and select the location of the keystore certificate.
      - **Key Alias**: Click **Import** to import the key alias.

      Note:

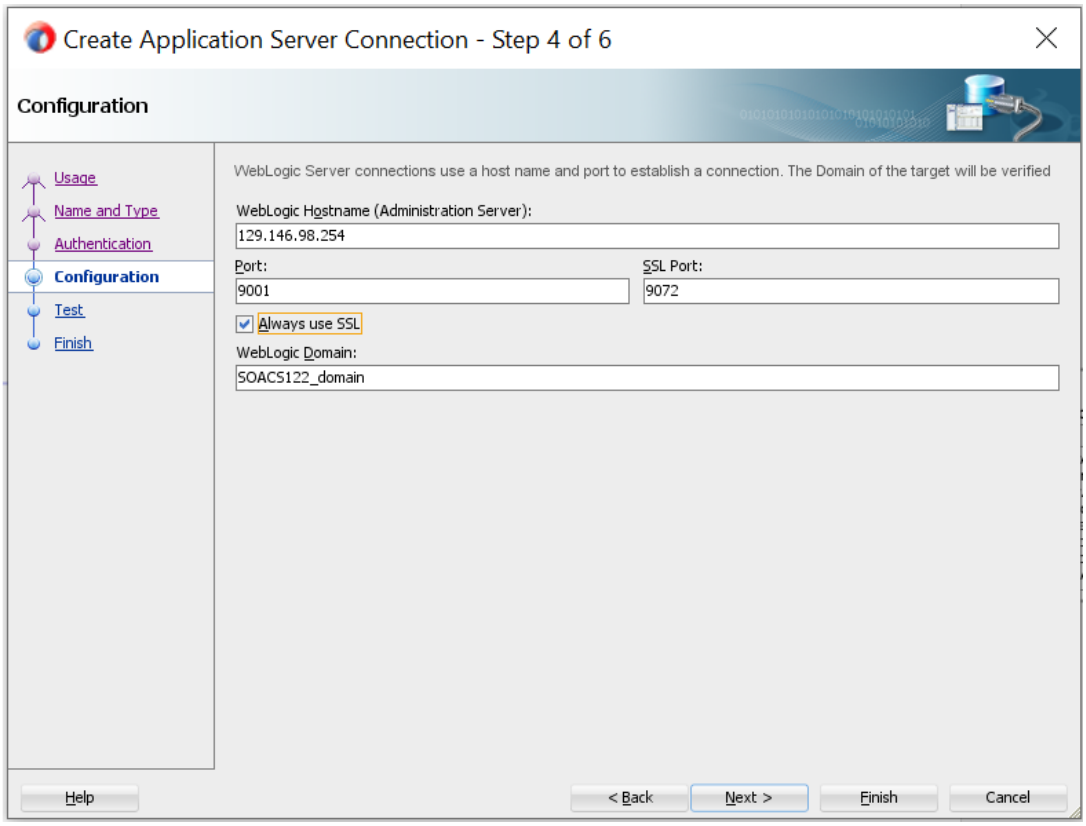
      Repeat the step for cacerts. The cacerts file is a collection of trusted Certificate Authority (CA) certificates. Specify the following details:
      - **Keystore Location**: Click **Browse** to find and select the JDK cacert [JAVA_HOME\jre\lib\security\cacerts].

- **Keystore Password**: Specify the keystore password. The default password is changeit.
- **Certificate Location**: Click **Browse** and select the certificate used to create the WebLogic Server Connection.
- **Key Alias**: Click **Import** to import the key alias.

3. Restart JDeveloper.
4. On the Name and Type page, in the **Connection Name** field, enter a name for the connection, and select a **Connection Type** of **WebLogic 12.x**.



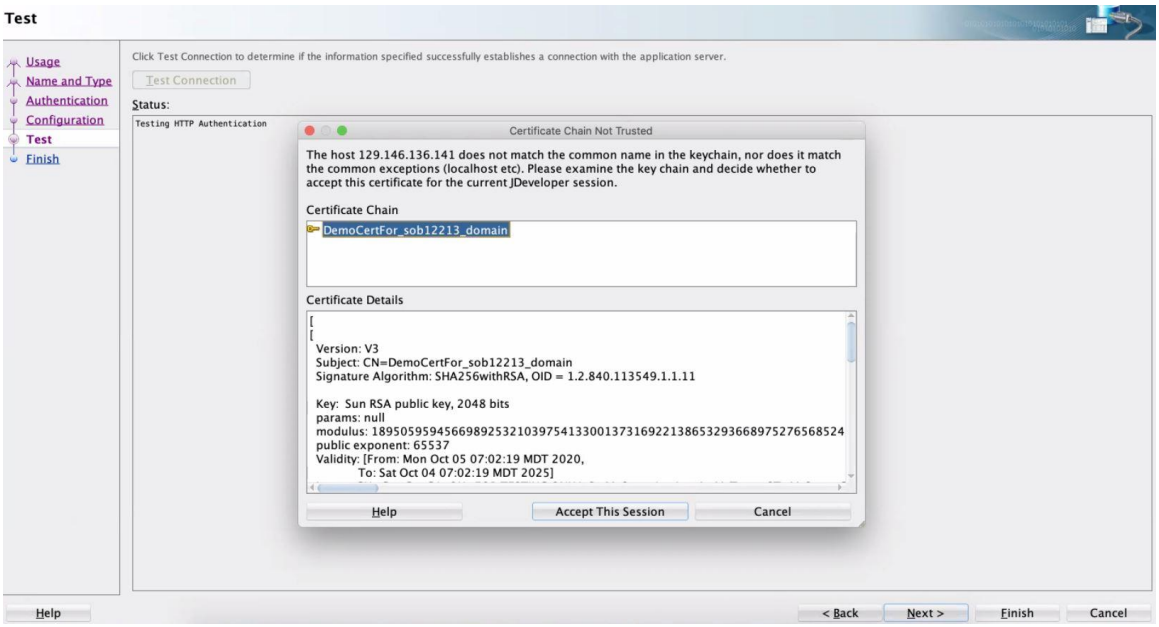
5. On the Authentication page, enter your WebLogic Server credentials.
6. On the Configuration page:
   - In the **WebLogic Hostname (Administration Server)** field, enter the public IP address of the Administration Server that you noted down for the provisioned Oracle SOA Suite on Marketplace instance.
   - Enter a **Port** value of 9001 and an **SSL port** value of 9072.
   - Select **Always use SSL** when the instance is using a public IP address. For instances with a private IP address only, leave this unchecked.
   - Enter the name of your **WebLogic Domain**.

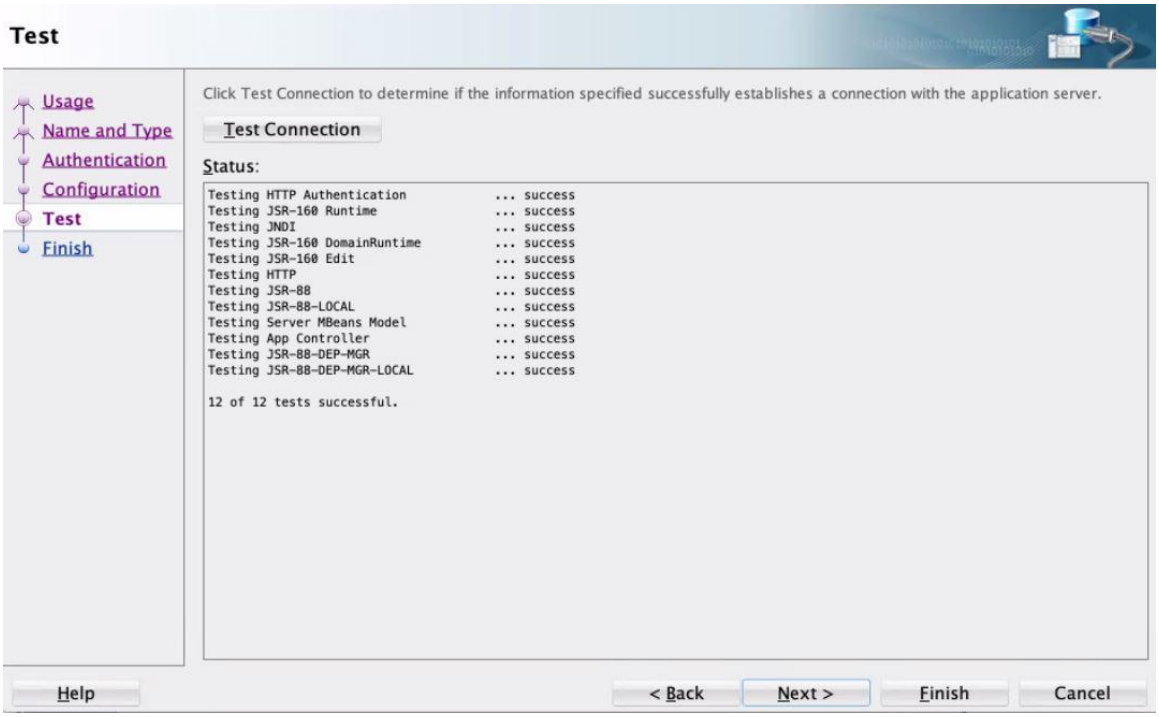


7. On the Test page, click **Test Connection**. If the instance is using a public IP address, then click **Accept This Session** to accept the certificates in the dialog that is displayed.

   Note:

   If the Certificate Chain Not Trusted dialog does not display, you must clear your JDeveloper cache as described in step 1 and try again.

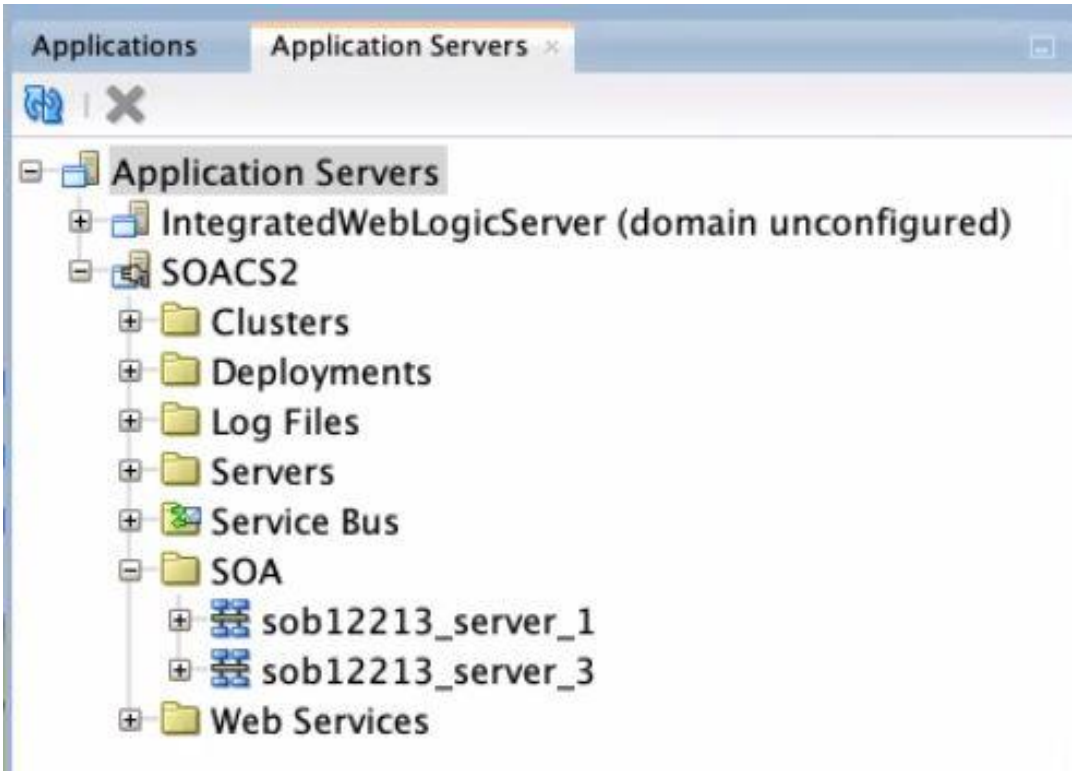




Notes:

- If **Test Connection** has failures, then ensure that /etc/hosts has the required entries and ports 9072/9074 allow inbound traffic from the JDeveloper host.
- Do not proceed without accepting the certificates when using instances with a public IP address.

8. In JDeveloper, on the Application Servers tab, expand the connection name, then **SOA** (or **Service Bus**), and confirm that the names of the Managed Servers are listed, indicating that the connection is established from JDeveloper to the servers. If servers are not displayed, then check the /etc/hosts file has both host name and fully qualified domain name entries.

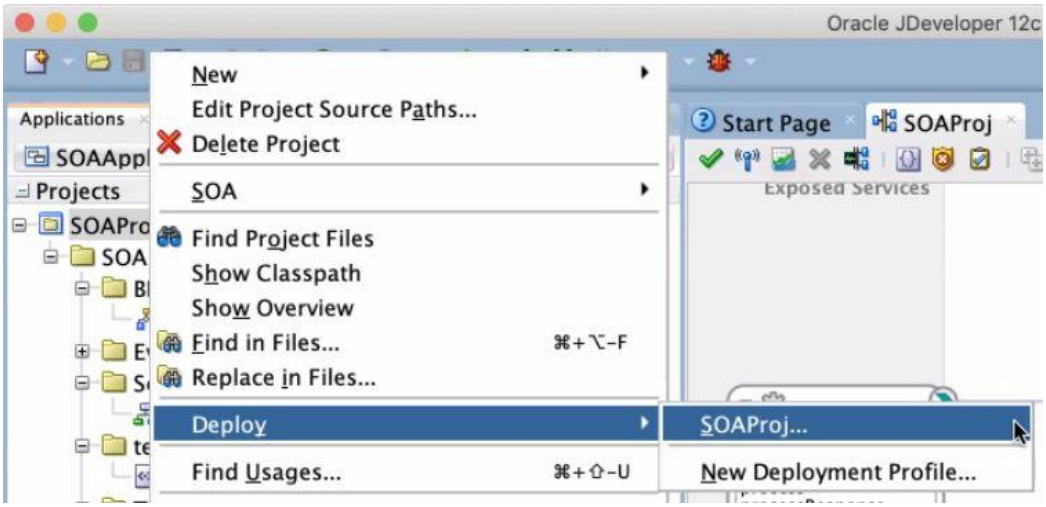


Deploy a SOA Composite Application to Oracle SOA Suite on Marketplace from JDeveloper

SOA composite applications are deployed to Managed Servers.

To deploy a SOA composite application to Oracle SOA Suite on Marketplace from JDeveloper:

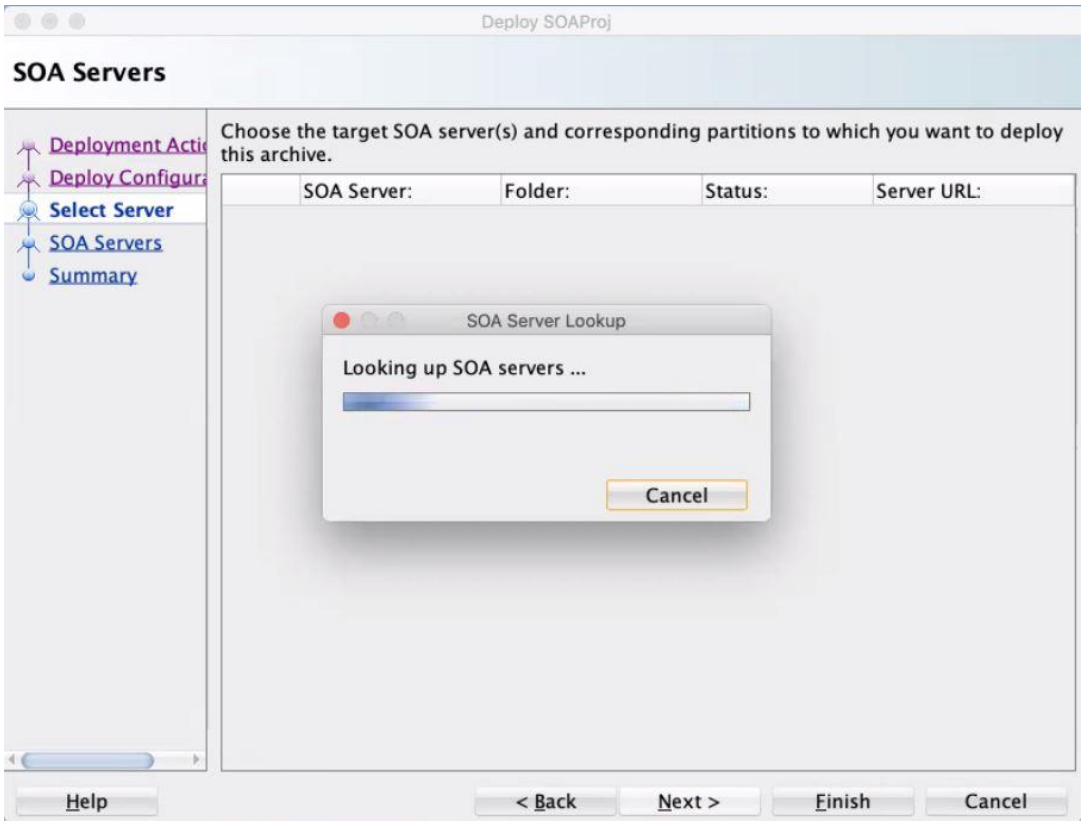
1. In JDeveloper, right-click the SOA project you want to deploy and select **Deploy**, then the name of the project.

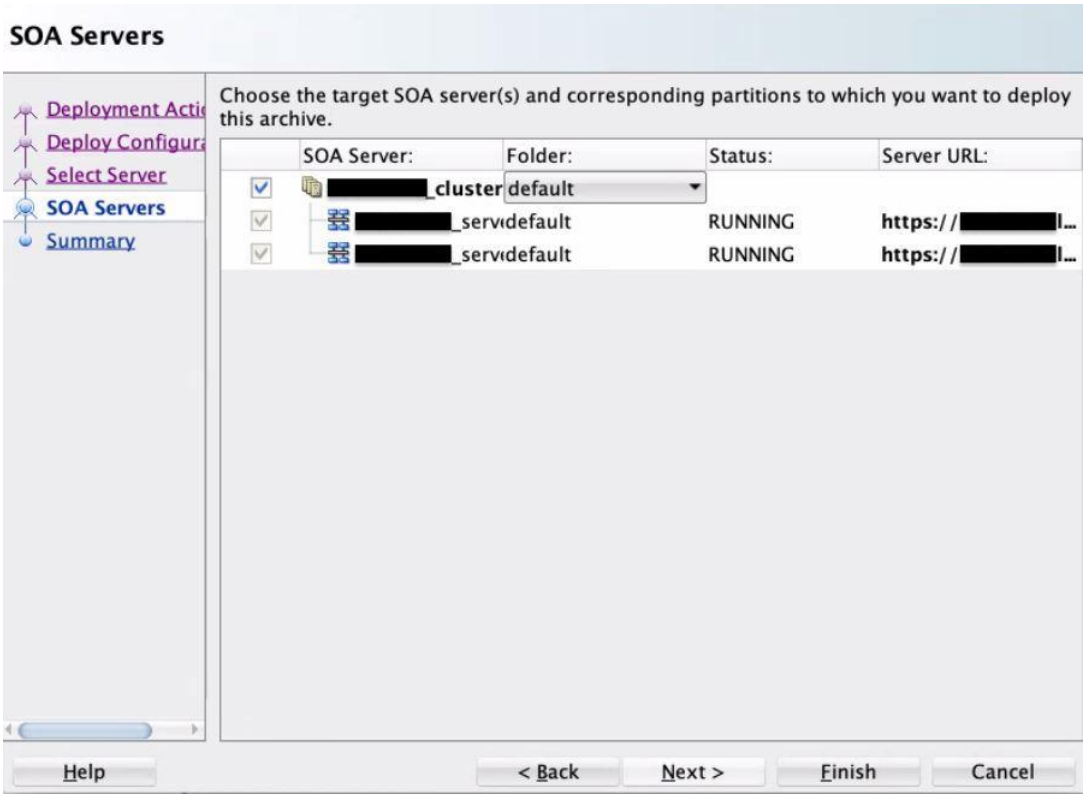


The deployment wizard is displayed.

2. On the Select Server page, select the application server connection that you created.

   If the server is configured correctly, the deployment wizard looks up the SOA servers and shows the SOA servers to which to deploy the SOA composite application.

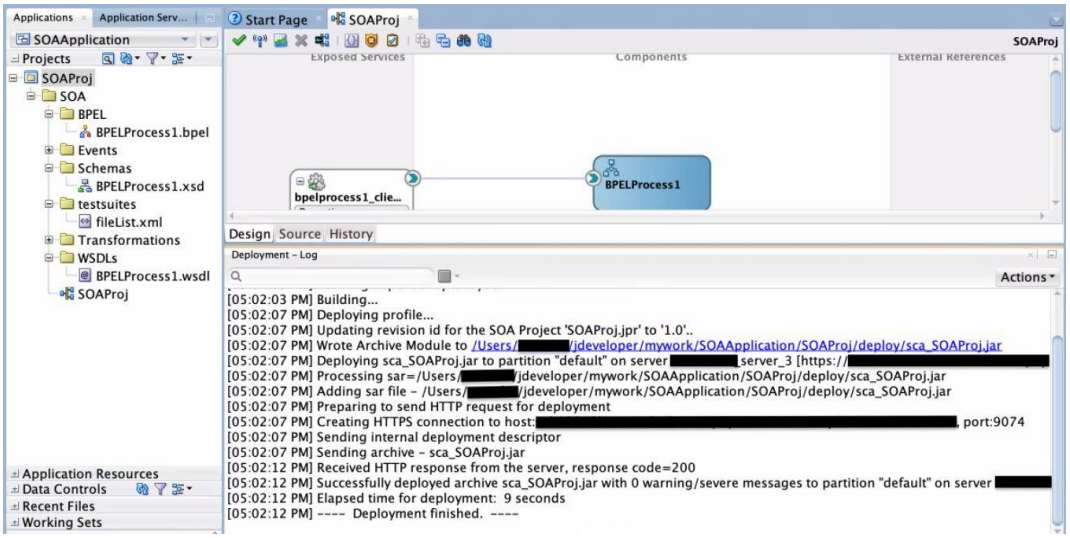




Note:

If the SOA Server lookup has failures, then ensure that /etc/hosts has the required entries and ports 9072/9074 allow inbound traffic from the JDeveloper host.

3. Click **Finish** and verify that the deployment completes successfully as shown in the following screenshot.



The JDeveloper Console logs indicate that the composite application was deployed successfully.