WildFly Scripts

Scripts for WildFly Standalone Application Server

Standalone WildFly deployment includes a ServiceCatalogServer and ServiceLinkServer that can be installed on the same application server or on different servers.

Following are the start and stop scripts available in the installer for Standalone WildFly deployments.

Table B-1 Scripts for WildFly Standalone Application Server

Scripts	Execute On	Description
startServiceCatalog	VM1	Start Service Catalog on VM1
startServiceLink	VM1	Start Service Link on VM1
stopServiceCatalog	VM1	Stop Service Catalog on VM1
stopServiceLink	VM1	Stop Service Link on VM1

Scripts for WildFly Cluster Application Server

Cluster setup for WildFly can be configured in two VM or four VM topology.

Two VM topology:

- VM 1 is Domain Controller, Host Controller, and Service Link for Host 1
- VM 2 is Host Controller for Host 2

Four VM topology:

- VM 1 is Domain Controller
- VM 2 is standalone Service Link
- VM 3 is Host Controller for Host 1
- VM 4 is Host Controller for Host 2

Table B-2 Scripts for WildFly Cluster Application Server

Scripts	Execute On	Description
2 VM Cluster Topology: **VM1 is Domain Controller, HC1, and Service Link and VM2 is HC2.**		
startServiceCatalogCluster	VM1	Start the following servers on VM1: a) Domain Controller, b) Process Controller, c) Managed server for Service Link, d) Managed server for Service Catalog. **Note** startServiceCatalogCluster script is used to check whether the exploded $JBOSS_HOME/content/RequestCenter.war and $JBOSS_HOME/content/ISEE.war are already available. If yes, then the script will not create the content folder. If not, then it will explode the WAR files from /dist folder and create the content folder under these directories. This is applicable for VM1 and VM2 in two VM topology and for VM 3 and VM 4 in four VM topology.
startServiceCatalogOnHC1	VM1	Start the Managed server for Service Catalog on VM1
startServiceLinkOnHC1	VM1	Start the Managed server for Service Link on VM1.
startServiceCatalogOnHC2	VM1	Start the Managed server for Service Catalog on VM2.
startServiceCatalogOnHC3	VM1	Start the Managed server for Service Catalog on VM3. (Reserved for when user wants to add Host Controller 3.)
startServiceCatalogOnHC4	VM1	Start the Managed server for Service Catalog on VM3. (Reserved for when user wants to add Host Controller 4.)
deployServiceCatalogCluster all	VM1	Deploy the followings: a) ISEE.war, b) RequestCenter.war
deployServiceCatalogCluster RC	VM1	Deploy RequestCenter.war
deployServiceCatalogCluster SL	VM1	Deploy ISEE.war
shutdownAllOnHC1	VM1	Stop the following servers on VM1: a) Domain Controller, b) Process Controller, c) Managed server for Service Link, d) Managed server for Service Catalog.
forceStopAllOnHC	VM1	If for some reasons, the shutDownAllOnHC1 script doesn't work, then execute this script to force stop the following servers on VM1: a) Domain Controller, b) Process Controller, c) Managed server for Service Link, d) Managed server for Service Catalog
stopServiceCatalogOnHC1	VM1	Stop the Managed server for Service Catalog on VM1.
stopServiceLinkOnHC1	VM1	Stop the Managed server for Service Link on VM1.
stopServiceCatalogOnHC2	VM1	Stop the Managed server for Service Catalog on VM2.
shutdownAllOnHC2	VM1	Stop both the Process Controller and the Managed server on VM2.

Table B-2 Scripts (continued) for WildFly Cluster Application Server

Scripts	Execute On	Description
stopServiceCatalogOnHC3	VM1	Stop the Managed server for Service Catalog on VM3. (Reserved for when user wants to add HC3.)
shutdownAllOnHC3	VM1	Stop both the Process Controller and the Managed server on VM3. (Reserved for when user wants to add HC3.)
stopServiceCatalogOnHC4	VM1	Stop the Managed server for Service Catalog on VM3. (Reserved for when user wants to add HC4.)
shutdownAllOnHC4	VM1	Stop both the Process Controller and the Managed server on VM4. (Reserved for when user wants to add HC4.)
undeployServiceCatalogCluster all	VM1	Undeploy the followings: a) ISEE.war, b) RequestCenter.war
undeployServiceCatalogCluster RC	VM1	Undeploy RequestCenter.war
undeployServiceCatalogCluster SL	VM1	Undeploy ISEE.war
startServiceCatalogCluster	VM2	Start the following servers: a) Process Controller, b) Managed server for Service Catalog. The script is used to check whether the exploded $JBOSS_HOME/content/RequestCenter.war and $JBOSS_HOME/content/ISEE.war are already available. If yes, then the script will not create the content folder. If not, then it will explode the WAR files from /dist folder and create the content folder under these directories.
forceStopAllOnHC	VM2	Force stop: a) Process Controller, b) Managed server for Service Catalog.
If customer adds another VM as HC3		
startServiceCatalogCluster	VM3	Start the following servers: a) Process Controller, b) Managed server for Service Catalog.
forceStopAllOnHC	VM3	Force stop: a) Process Controller, b) Managed server for Service Catalog.
If customer adds another VM as HC4		
startServiceCatalogCluster	VM4	Start the following servers: a) Process Controller, b) Managed server for Service Catalog.
forceStopAllOnHC	VM4	Force stop: a) Process Controller, b) Managed server for Service Catalog.
4 VM Cluster Topology: **VM1 is Domain Controller, VM2 is Standalone for Service Link, VM3 is HC1, VM4 is HC2**		
startServiceCatalogCluster	VM1	Start the Domain Controller on VM1.
startServiceCatalogOnHC1	VM1	Start the Managed server for Service Catalog on VM3.
startServiceCatalogOnHC2	VM1	Start the Managed server for Service Catalog on VM4.

Table B-2 Scripts (continued) for WildFly Cluster Application Server

Scripts	Execute On	Description
startServiceCatalogOnHC3	VM1	Start the Managed server for Service Catalog on VM5. (Reserved for when user wants to add HC3.)
startServiceCatalogOnHC4	VM1	Start the Managed server for Service Catalog on VM6. (Reserved for when user wants to add HC4.)
deployServiceCatalogCluster RC	VM1	Deploy RequestCenter.war
stopDomainController	VM1	Stop the Domain Controller on VM1.
stopServiceCatalogOnHC1	VM1	Stop the Managed server for Service Catalog on VM3.
shutdownAllOnHC1	VM1	Stop both the Process Controller and the Managed server on VM3.
stopServiceCatalogOnHC2	VM1	Stop the Managed server for Service Catalog on VM4.
shutdownAllOnHC2	VM1	Stop both the Process Controller and the Managed server on VM4
stopServiceCatalogOnHC3	VM1	Stop the Managed server for Service Catalog on VM5. (Reserved for when user wants to add HC3.)
shutdownAllOnHC3	VM1	Stop both the Process Controller and the Managed server on VM5. (Reserved for when user wants to add HC3.)
stopServiceCatalogOnHC4	VM1	Stop the Managed server for Service Catalog on VM6. (Reserved for when user wants to add HC4.)
shutdownAllOnHC4	VM1	Stop both the Process Controller and the Managed server on VM6. (Reserved for when user wants to add HC4.)
undeployServiceCatalogCluster RC	VM1	Undeploy RequestCenter.war
VM1 is Domain Controller, VM2 is Standalone for Service Link, VM3 is HC1, and VM4 is HC2		
StartServiceLink	VM2	Start Service Link on VM2.
StopServiceLink	VM2	Stop Service Link on VM2.
VM1 is Domain Controller, VM2 is Standalone for Service Link, VM3 is HC1, and VM4 is HC2		
startServiceCatalogCluster	VM4	Start the following servers: a) Process Controller, b) Managed server for Service Catalog. The script is used to check whether the exploded $JBOSS_HOME/content/RequestCenter.war and $JBOSS_HOME/content/ISEE.war are already available. If yes, then the script will not create the content folder. If not, then it will explode the WAR files from /dist folder and create the content folder under these directories.
forceStopAllOnHC	VM4	Force stop: a) Process Controller, b) Managed server for Service Catalog.
If customer adds another VM as HC3		
startServiceCatalogCluster	VM5	Start the following servers: a) Process Controller, b) Managed server for Service Catalog.
forceStopAllOnHC	VM5	Force stop: a) Process Controller, b) Managed server for Service Catalog.
If customer adds another VM as HC4		

Table B-2 Scripts (continued) for WildFly Cluster Application Server

Scripts	Execute On	Description
startServiceCatalogCluster	VM6	Start the following servers: a) Process Controller, b) Managed server for Service Catalog.
forceStopAllOnHC	VM6	Force stop: a) Process Controller, b) Managed server for Service Catalog.

Adding Subsequent Host Nodes Manually in WildFly Cluster

Before You Begin

1. Run the GUI Installer for the host controller setup on the subsequent node, vm<N>, where N is the number of the node.
2. Select the Host Controller in the Node Type Selection panel followed by Host1 as the cluster node.
3. Do not run any of the startup-scripts.

Step 1 Rename host1.xml to host<N>.xml. The file, host1.xml is present under the *InstallationDirectory/wildfly-8.2.0.Final/domain/configuration* directory.

Step 2 Follow the below steps to create a new user:

a. Execute **add-user.sh** or **add-user.bat** script from *InstallationDirectory/wildfly-8.2.0.Final/bin* location.

b. Enter a to select the Management User.

c. Provide **HOST<N>** as the Username.

d. Provide **HOST<N>** as the Password.

e. Enter yes to use the entered password.

f. Re-Enter **HOST<N>** for Password confirmation.

g. Press **Enter** to pass the management group information

h. Enter **yes** to add the user 'HOST<N>' for realm 'ManagementRealm'.

i. Enter **no** for the interconnection of AS process.

Note

- The new user needs to be created in the machine having host Controller(host<N>) and domain controller (host1) as well.
- Verify that the user is added in the *InstallationDirectory/wildfly-8.2.0.Final/domain/configuration/mgmt-users.properties*

Step 3 Perform the following changes in host<N>.xml:

a. Change the name of Host from HOST1 to HOST<N>.location for host<N>.xml. This file is available in the C:\Installation_directory\wildfly-8.2.0.Final\domain\configuration

```
<host name="HOST5" xmlns="urn:jboss:domain:2.2”>
```

b. Change the secret value in the <server-identities> section. The secret value can be obtained from the website *www.motobit.com*:

```
<secret value="SE9TVDU=“/>
```

Enter the URL http://www.motobit.com/util/base64-decoder-encoder.asp

•Enter HOST<N> in the text box (without any extra space or newline).

•Click on Convert the Source Data button.

c. Change the name of the server instance from server-host1-RC to server-host<N>-RC to avoid name conflicts:

```
<server name="server-host<N>-RC" group="main-server-group" auto-start="true”>
```

d. Change the CONTROLLER_TYPE from host1 to host<N> in setEnv.cmd script in *InstallationDirectory/bin* directory.

e. Delete the logs and servers directories from *InstallationDirectory/wildfly-8.2.0.Final/domain*, if exists.

Step 4 The following changes need to be done in startServiceCatalogCluster.cmd script in *InstallationDirectory/bin* directory.

a. Add the following code snippet in BOLD in the script file:

```
#Below if condition is applicable for the windows OS platform

if "%CONTROLLER_TYPE%"=="host<N>" (
    if exist "%WILDFLY_BASE_DIR%\configuration\domain.xml" rename
"%WILDFLY_BASE_DIR%\configuration\domain.xml" "domain_backup.xml"
)

#Below if condition is applicable for the Linux OS platform

#pause 'Press [Enter] key to continue6...'
FILE=${WILDFLY_BASE_DIR}/configuration/host<N>.xml
if [ "${CONTROLLER_TYPE}" == "host<N>" ]
then
    if [  -f "$FILE" ]
    then
        /bin/mv -i ${WILDFLY_BASE_DIR}/configuration/host<N>.xml
${WILDFLY_BASE_DIR}/configuration/host.xml
    fi
fi
```

b. Add the following code snippet in BOLD in the script file:

```
if "%CONTROLLER_TYPE%" =="host<N>" if exist "%WILDFLY_BASE_DIR%\configuration\host<N>.xml"
rename "%WILDFLY_BASE_DIR%\configuration\host<N>.xml" "host.xml"

If the Platform is Linux OS , skip the step b and proceed. above if condition only
applicable for the windows OS platform.
```

Step 5 Start the server with startServiceCatalogCluster.cmd or sh and once it is up, verify from the Wildfly admin server console under the domain whether the host<N> is registered.

Step 6 Addition of scripts for both 4-VM Cluster and 2-VM Cluster in the VM-1 machine (machine containing the domain controller)

a. Copy *shutdownAllOnHC1.cmd* and rename the copied script as *shutdownAllOnHC<N>.cmd*

b. Copy *startServiceCatalogOnHC1.cmd* and rename the copied script as *startServiceCatalogOnHC<N>.cmd*

c. Copy *stopServiceCatalogOnHC1.cmd* and rename the copied script as *stopServiceCatalogOnHC<N>.cmd*

Note In the Linux platform if new scripts does not have the read/write permission, you should manually assign the read/write permission before executing.

d. Perform the below operations in the *stopServiceCatalogOnHC<N>.cmd* script file cautiously:

- Rename HOST1 to HOST<N>.
- Rename host1 to host<N>.

Configuring Plugin for IIS Web Server

This section provides information on how to configure the plugin for IIS web server (version 8.x) on a Windows Server 2012 R2 machine to redirect to clustered Wildfly application servers (version 8.2.0.Final).

To configure plugin got IIS Web Server:

Step 1 Add Web Server Role for IIS.

Step 2 Install Tomcat Plugin.

Step 3 Copy WAR Directories.

Step 4 Create Virtual Directories for IIS.

Step 5 Modify Plugin Properties.

Step 6 Configure Instance-ID for WildFly.

Step 7 Test IIS.

Add Web Server Role for IIS

Step 1 Navigate to **Service Manager Dashboard > Manage Tab > Add Roles and Features**.

Step 2 Add the role **Web Server (IIS)** on your Windows Server 2012 R2 operating system.
Make sure you select the **ISAPI Extensions** and **ISAPI Filters** role services when adding IIS.

Step 3 After adding the **Web Server (IIS) Role**, start the **World Wide Web Publishing Service**.

Step 4 Launch a browser and connect to **URL = http://localhost**.
You should see the **Welcome IIS 8.x** page.

Install Tomcat Plugin

Step 1 Download the **PSC 11.x** software package from cisco.com and extract it.

Step 2 Cd to **<PSC_Software_Extract_Dir>\isapi** directory.

Step 3 Copy file **tomcat-isapi-redirector-v1.2.37.zip** to your IIS web server machine and extract it under the **C:\inetpub\isapi directory**.

Note Create the *isapi* sub-directory under C:\inetpub first if necessary.

Step 4 Cd to **C:\inetpub\isapi** and rename the file i**sapi_redirect-1.2.37-win64.dll** to **isapi_redirect.dll**.

Copy WAR Directories

If you have WildFly installation and your IIS is on a separate machine from WildFly, then perform the following steps to copy WAR directories:

Standalone WildFly Installation

Step 1 On the WildFly machine where the **RequestCenter Server** is running, **cd** to the **<PSC_Install_Dir>\wildfly-8.2.0.Final\ServiceCatalogServer\deployments** directory.

Step 2 Copy the entire **RequestCenter.war** sub-directory from this machine to the IIS machine and place it under the **C:\inetpub\WAR** directory.

Step 3 On the WildFly machine where the **ServiceLink server** is running, **cd** to the **<PSC_Install_Dir>\wildfly-8.2.0.Final\ServiceLinkServer\deployments** directory.

Step 4 Copy the entire **ServiceLink.war** sub-directory from this machine to the IIS machine and place it under the **C:\inetpub\WAR** directory.

Clustered WildFly installation with Four VM Topology

Step 1 On the WildFly machine where one of the cluster nodes for **RequestCenter** is running, **cd** to the **<PSC_Install_Dir>\wildfly-8.2.0.Final\content** directory.

Step 2 Copy the entire **RequestCenter.war** sub-directory from this machine to the IIS machine and place it under the **C:\inetpub\WAR** directory.

Step 3 On the WildFly machine where the **ServiceLink server** is running, **cd** to the **<PSC_Install_Dir>\wildfly-8.2.0.Final\ServiceLinkServer\deployments** directory.

Step 4 Copy the entire **ServiceLink.war** sub-directory from this machine to the IIS machine and place it under the **C:\inetpub\WAR** directory.

Clustered Wildfly installation with Two VM Topology

Step 1 On the WildFly machine that is the primary cluster node for **RequestCenter, cd** to the **<PSC_Install_Dir>\wildfly-8.2.0.Final\content** directory.

Step 2 Copy the entire **RequestCenter.war** sub-directory from this machine to the IIS machine, and place it under the **C:\inetpub\WAR** directory.

Step 3 On the same WildFly machine, **cd** to the **<PSC_Install_Dir>\wildfly-8.2.0.Final\content** directory.

Step 4 Copy the entire **ISEE.war** sub-directory from this machine to the IIS machine and place it under the **C:\inetpub\WAR** directory.

Step 5 Rename the folder to **C:\inetpub\WAR\ServiceLink.war**.

Create Virtual Directories for IIS

Step 1 On the IIS machine, launch **Internet Information Services** (IIS) **Manager**.

Step 2 Choose **Hostname > Sites > Default Web Site**.

Step 3 Right click on the **Default Web Site** and select **Add Virtual Directory**.

Step 4 On the pop up window, enter the following values, and then click **OK**:

```
Alias = RequestCenter
Physical path = <Click the browse button, and select the
"C:\inetpub\WAR\RequestCenter.war" directory.>
```

Step 5 Right click on **Default Web Site** and select **Add Virtual Directory** to add another directory.

Step 6 On the pop up window, enter the following values, and then click **OK**:

```
Alias = IntegrationServer
Physical path = <Click the browse button, and select the
"C:\inetpub\WAR\ServiceLink.war" directory.>
```

Step 7 Right click on **Default Web Site** and select **Add Virtual Directory** to add another directory.

Step 8 On the display window, enter the following values, and then click **OK**:

```
Alias = tomcat
Physical path = C:\inetpub\isapi
```

Step 9 Click on **Default Web Site** node. And on the right pane, double click **ISAPI Filters**.

Step 10 Click **Add** link under the **Actions** column on the right pane.

Step 11 On the pop up window, enter the following values, and then click **OK**:

```
Filter name = tomcat
Executable = C:\inetpub\isapi\isapi_redirect.dll
```

Step 12 Click on **Default Web Site** node. And on the right panel, double click **Handler Mappings**.

Step 13 Click the **Edit Feature Permissions** link under the **Actions** column on the right pane.

Step 14 On the pop up window, select **all Read**, **Script** and **Execute** check boxes, and then click **OK**.

Step 15 Click on the **Hostname** node. On the right panel, double click on **ISAPI and CGI Restrictions**.

Note The *Hostname* node is the parent node of sites.

Step 16 Click the **Add** link under the **Actions** column on the right most pane.

Step 17 On the display window, enter the following values, and then click **OK**:

```
ISAPI or CGI path = C:\inetpub\isapi\isapi_redirect.dll
Description = Tomcat ISAPI Filter
Select the checkbox "Allow extension path to execute".
```

Modify Plugin Properties

Step 1 Modify the file **C:\inetpub\isapi\isapi_redirect.properties** as follows:

```
# Configuration file for the Jakarta ISAPI Redirector
# The path to the ISAPI Redirector Extension, relative to the website
# This must be in a virtual directory with execute privileges
extension_uri=/tomcat/isapi_redirect.dll
# Full path to the log file for the ISAPI Redirector
log_file=C:\inetpub\isapi\logs\isapi_redirect.log
# Log level (debug, info, warn, error or trace)
log_level=error
# Full path to the workers.properties file
worker_file=C:\inetpub\isapi\conf\workers.properties
# Full path to the uriworkermap.properties file
worker_mount_file=C:\inetpub\isapi\conf\uriworkermap.properties
```

Step 2 Modify the file **C:\inetpub\isapi\conf\uriworkermap.properties** as follows:

```
/RequestCenter=router1
/RequestCenter/*=router1
/RequestCenter/servlet/*=router1

/IntegrationServer=router2
/IntegrationServer/*=router2
/IntegrationServer/servlet/*=router2

/private/admin/jkstatus=jkstatus
```

Step 3 If you have a standalone WildFly installation, then modify the file **C:\inetpub\isapi\conf\workers.properties** as follows:

```
# Define list of workers that will be used for mapping requests
worker.list=router1,router2,jkstatus

# Define Node1 worker for RequestCenter
worker.node1.port=8009
worker.node1.host=<IP_Address_of_RC_host>
worker.node1.type=ajp13
worker.node1.lbfactor=1
worker.router1.type=lb
worker.router1.balance_workers=node1

# Define Node2 worker for ServiceLink
worker.node2.port=6009
worker.node2.host=<IP_Address_of_SL_host>
worker.node2.type=ajp13
worker.node2.lbfactor=1
worker.router2.type=lb
worker.router2.balance_workers=node2

# Define a jkstatus worker using status
worker.jkstatus.port=8009
worker.jkstatus.host=<IP_Address_of_IIS_host>
```

```
worker.jkstatus.type=status
worker.status.type=status
```

The port numbers *8009* and *6009* should be set to the actual *ajp* port numbers used by the *RequestCenter server* and *ServiceLink server.*

Step 4 If you have a clustered WildFly installation (regardless of 4-VM Topology or 2-VM Topology), then modify the file **C:\inetpub\isapi\conf\workers.properties** as follows:

Note The following is an example for a WildFly installation with 2 cluster nodes of *RequestCenter.*

- That is there are 2 *RequestCenter servers* running on two separate VM's.
- If you have 3 or 4 cluster nodes, you just need to follow this example and add a section for *rcnode3* and for *rcnode4* appropriately.

```
# Define list of workers that will be used for mapping requests
worker.list=router1,router2,jkstatus

# Define rcnode1 worker for RequestCenter 1
worker.rcnode1.port=8009
worker.rcnode1.host=<IP_Address_of_RC1_host>
worker.rcnode1.type=ajp13
worker.rcnode1.lbfactor=1

# Define rcnode2 worker for RequestCenter 2
worker.rcnode2.port=8009
worker.rcnode2.host=<IP_Address_of_RC2_host>
worker.rcnode2.type=ajp13
worker.rcnode2.lbfactor=1

# For clustering, set the line below to rcnode1, rcnode2, etc…
worker.router1.type=lb
worker.router1.balance_workers=rcnode1,rcnode2

# Define slnode worker for ServiceLink
worker.slnode.port=6009
worker.slnode.host=<IP_Address_of_SL_host>
worker.slnode.type=ajp13
worker.slnode.lbfactor=1

worker.router2.type=lb
worker.router2.balance_workers=slnode

# Define a 'jkstatus' worker using status
worker.jkstatus.port=8009
worker.jkstatus.host=<IP_Address_of_IIS_host>
worker.jkstatus.type=status
worker.status.type=status
```

Step 5 Restart **World Wide Web Publishing Service**.

Step 6 If you have a standalone WildFly installation, skip to the section Test IIS.

Note If you have clustered Wildfly installation, continue to the next section Configure instance-id for Wildfly.

Configure Instance-ID for WildFly

For 4 VM Topology

You must perform the following steps for clustered WildFly installation with 4 VM topology:

Step 1 Log on to the **WildFly Domain Controller** machine, and stop all WildFly servers.

Step 2 Open file **<PSC_Install_Dir>\wildfly-8.2.0.Final\domain\configuration\domain.xml** and search for the following line:

```
<subsystem xmlns="urn:jboss:domain:undertow:1.2">
```

Replace it with the following value:

```
<subsystem xmlns="urn:jboss:domain:undertow:1.2" instance-id="${jboss.web.instanceId}">
```

Step 3 Log on to the **Host Controller 1** for **RequestCenter**, and stop all WildFly servers.

Step 4 Open file **<PSC_Install_Dir>\wildfly-8.2.0.Final\domain\configuration\host1_backup.xml**, and search for the following section:

```
<servers>
<server name="server-host1-RC" group="main-server-group" auto-start="true">
<jvm name="default">
<heap size="2048m" max-size="2048m"/>
<permgen size="512m" max-size="512m"/>
<jvm-options>
<!--<option value="-Xrunjdwp:transport=dt_socket,address=8787,server=y,suspend=n"/>-->
<option
value="-XX:CompileCommand=exclude,com/newscale/bfw/signon/filters,AuthenticationFilter"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xml/dtm/ref/sax2dtm/SAX2DTM,startElement"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xpath/compiler/XPathParser,UnionExpr"/>
</jvm-options>
</jvm>
<socket-bindings socket-binding-group="ha-sockets" port-offset="0"/>
</server>
</servers>
```

Replace it with the following:

```
<servers>
<server name="server-host1-RC" group="main-server-group" auto-start="true">
<jvm name="default">
<heap size="2048m" max-size="2048m"/>
<permgen size="512m" max-size="512m"/>
<jvm-options>
<!--<option value="-Xrunjdwp:transport=dt_socket,address=8787,server=y,suspend=n"/>-->
<option
value="-XX:CompileCommand=exclude,com/newscale/bfw/signon/filters,AuthenticationFilter"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xml/dtm/ref/sax2dtm/SAX2DTM,startElement"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xpath/compiler/XPathParser,UnionExpr"/>
</jvm-options>
```

```
</jvm>
<socket-bindings socket-binding-group="ha-sockets" port-offset="0"/>
<system-properties>
   <property name="jboss.web.instanceId" value="rcnode1"/>
</system-properties>
</server>
</servers>
```

Step 5 Log on to the **Host Controller 2** for **RequestCenter**, and stop all Wildfly servers.

Step 6 Open file **<PSC_Install_Dir>\wildfly-8.2.0.Final\domain\configuration\host2_backup.xml**, and search for the following section:

```
<servers>
<server name="server-host2-RC" group="main-server-group" auto-start="true">
<jvm name="default">
<heap size="2048m" max-size="2048m"/>
<permgen size="512m" max-size="512m"/>
<jvm-options>
<!--<option value="-Xrunjdwp:transport=dt_socket,address=8787,server=y,suspend=n"/>-->
<option
value="-XX:CompileCommand=exclude,com/newscale/bfw/signon/filters,AuthenticationFilter"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xml/dtm/ref/sax2dtm/SAX2DTM,startElement"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xpath/compiler/XPathParser,UnionExpr"/>
</jvm-options>
</jvm>
<socket-bindings socket-binding-group="ha-sockets" port-offset="0"/>
</server>
</servers>
```

Replace it with the following:

```
<servers>
<server name="server-host2-RC" group="main-server-group" auto-start="true">
<jvm name="default">
<heap size="2048m" max-size="2048m"/>
<permgen size="512m" max-size="512m"/>
<jvm-options>
<!--<option value="-Xrunjdwp:transport=dt_socket,address=8787,server=y,suspend=n"/>-->
<option
value="-XX:CompileCommand=exclude,com/newscale/bfw/signon/filters,AuthenticationFilter"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xml/dtm/ref/sax2dtm/SAX2DTM,startElement"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xpath/compiler/XPathParser,UnionExpr"/>
</jvm-options>
</jvm>
<socket-bindings socket-binding-group="ha-sockets" port-offset="0"/>
<system-properties>
   <property name="jboss.web.instanceId" value="rcnode2"/>
</system-properties>
</server>
</servers>
```

Note If you have more than 2 cluster nodes, then repeat Step 5 and Step 6 for each subsequent node. For example,

- On the 3rd node, you need to modify the file *host3_backup.xml* and add the system-properties for *rcnode3*.
- On the 4th node, you need to modify the file *host4_backup.xml* and add the system-properties for *rcnode4*.

Step 7 Start up Wildfly servers on the **Domain Controller** machine and on each **Host Controller** machine.

Step 8 Go to the section Test IIS.

For 2 VM Topology

Perform the steps in this section only if you have a clustered WildFly installation with 2 VM topology:

Step 1 Log on to the **WildFly Domain Controller** machine, and stop all WildFly servers.

Step 2 Open file **<PSC_Install_Dir>\wildfly-8.2.0.Final\domain\configuration\domain.xml** and search for the following line:

```
<subsystem xmlns="urn:jboss:domain:undertow:1.2">
```

Replace it with the following value:

```
<subsystem xmlns="urn:jboss:domain:undertow:1.2" instance-id="${jboss.web.instanceId}">
```

Step 3 Open file **<PSC_Install_Dir>\wildfly-8.2.0.Final\domain\configuration \hostva_ backup.xml**, and search for the following section:

```
<servers>
<server name="server-host1-RC" group="main-server-group" auto-start="true">
<jvm name="default">
<heap size="2048m" max-size="2048m"/>
<permgen size="512m" max-size="512m"/>
<jvm-options>
<!--<option value="-Xrunjdwp:transport=dt_socket,address=8787,server=y,suspend=n"/>-->
<option
value="-XX:CompileCommand=exclude,com/newscale/bfw/signon/filters,AuthenticationFilter"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xml/dtm/ref/sax2dtm/SAX2DTM,startElement"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xpath/compiler/XPathParser,UnionExpr"/>
</jvm-options>
</jvm>
<socket-bindings socket-binding-group="ha-sockets" port-offset="0"/>
</server>
<server name="server-host1-SL" group="other-server-group" auto-start="true">
<jvm name="default">
<heap size="2048m" max-size="2048m"/>
<permgen size="512m" max-size="512m"/>
<jvm-options>
<option value="-server"/>
</jvm-options>
</jvm>
<socket-bindings socket-binding-group="standard-sockets" port-offset="0"/>
</server>
</servers>
```

Replace it with the following:

```
<servers>
<server name="server-host1-RC" group="main-server-group" auto-start="true">
<jvm name="default">
<heap size="2048m" max-size="2048m"/>
<permgen size="512m" max-size="512m"/>
<jvm-options>
<!--<option value="-Xrunjdwp:transport=dt_socket,address=8787,server=y,suspend=n"/>-->
<option
value="-XX:CompileCommand=exclude,com/newscale/bfw/signon/filters,AuthenticationFilter"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xml/dtm/ref/sax2dtm/SAX2DTM,startElement"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xpath/compiler/XPathParser,UnionExpr"/>
</jvm-options>
</jvm>
<socket-bindings socket-binding-group="ha-sockets" port-offset="0"/>
<system-properties>
   <property name="jboss.web.instanceId" value="rcnode1"/>
</system-properties>
</server>
<server name="server-host1-SL" group="other-server-group" auto-start="true">
<jvm name="default">
<heap size="2048m" max-size="2048m"/>
<permgen size="512m" max-size="512m"/>
<jvm-options>
<option value="-server"/>
</jvm-options>
</jvm>
<socket-bindings socket-binding-group="standard-sockets" port-offset="0"/>
<system-properties>
   <property name="jboss.web.instanceId" value="slnode"/>
</system-properties>
</server>
</servers>
```

Step 4 Log on to the **Host Controller 2** for **RequestCenter**, and stop all WildFly servers.

Step 5 Open file **<PSC_Install_Dir>\wildfly-8.2.0.Final\domain\configuration\host2_backup.xml**, and search for the following section:

```
<servers>
<server name="server-host2-RC" group="main-server-group" auto-start="true">
<jvm name="default">
<heap size="2048m" max-size="2048m"/>
<permgen size="512m" max-size="512m"/>
<jvm-options>
<!--<option value="-Xrunjdwp:transport=dt_socket,address=8787,server=y,suspend=n"/>-->
<option
value="-XX:CompileCommand=exclude,com/newscale/bfw/signon/filters,AuthenticationFilter"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xml/dtm/ref/sax2dtm/SAX2DTM,startElement"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xpath/compiler/XPathParser,UnionExpr"/>
</jvm-options>
</jvm>
<socket-bindings socket-binding-group="ha-sockets" port-offset="0"/>
</server>
```

```
</servers>
```

Replace it with the following:

```
<servers>
<server name="server-host2-RC" group="main-server-group" auto-start="true">
<jvm name="default">
<heap size="2048m" max-size="2048m"/>
<permgen size="512m" max-size="512m"/>
<jvm-options>
<!--<option value="-Xrunjdwp:transport=dt_socket,address=8787,server=y,suspend=n"/>-->
<option
value="-XX:CompileCommand=exclude,com/newscale/bfw/signon/filters,AuthenticationFilter"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xml/dtm/ref/sax2dtm/SAX2DTM,startElement"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option value="-XX:CompileCommand=exclude,org/exolab/castor/xml/Marshaller,marshal"/>
<option
value="-XX:CompileCommand=exclude,org/apache/xpath/compiler/XPathParser,UnionExpr"/>
</jvm-options>
</jvm>
<socket-bindings socket-binding-group="ha-sockets" port-offset="0"/>
<system-properties>
   <property name="jboss.web.instanceId" value="rcnode2"/>
</system-properties>
</server>
</servers>
```

Note If you have more than 2 cluster nodes, then repeat Step 4 and Step 5 for each subsequent node. For example,

- On the 3rd node, you need to modify the file *host3_backup.xml* and add the system-properties for *rcnode3*.
- On the 4th node, you need to modify file *host4_backup.xml* and add the system-properties for *rcnode4*.

Step 6 Start up WildFly servers on the **Domain Controller** machine and on the **Host Controller 2** machine.

Step 7 Go to the section Test IIS.

Test IIS

The following section provides information on verifying the connection to Prime Service Catalog.

Step 1 Verify by connecting to the URL **http://<IP_Address_of_IIS_Host>/RequestCenter**.
If you connected you can see the Login Page of Prime Service Catalog.

Step 2 For clustered WildFly installation with 2 nodes, stop one of the nodes.
And verify that you can still connect to the URL **http://<IP_Address_of_IIS_Host>/RequestCenter**.

Configuring Data Source for SQL in WildFly

Step 1 Log on to the WildFly Admin console (URL example below) with you credentials and click **OK**. This will take you to WildFly Application administrator console.

URL example:

http://<hostname>:<port>/RequestCenter

where,

<hostname> = The fully qualified domain hostname or the IP address of the computer where you installed the WildFly server for Service Catalog.

<port> =The HTTP Port number assigned to the Wildfly server for Service Catalog. The default value for HTTP Port number is 8080.

Step 2 Click on **Configuration** tab to be able to edit and make changes.

Step 3 Click **Add** to add a new data source.

Step 4 Enter **Name**: **SERVICECATALOGDS** and **JNDI Name: java:/<Name of the data source>**

Step 5 Click **Next**

Step 6 Select driver as "**Microsoft"** and click **Next**.

Step 7 Enter the connection URL: jdbc:sqlserver://<db_server>:1433;DatabaseName=VM236_RCDB_RC4

Step 8 Enter your credentials.

Step 9 Fill up the various fields under different tabs as mentioned in the table below:

Tab Name	Fieldname	Value
Security	Secure Domain	CiscoSecureDataSource
Connection	**Use JTA** and **Use CCMsECURITY**	Both these options should be checked.
Properties	1. SelectMethod: 2. sendStringParametersAsUnicode:	1. Direct 2. True
Pool	Minimum size is 20, maximum size is 80 and other values set to False.	
Validation		
	Background Validation	False
	Validation Millis	90000
	Validate on Match	False

Step 10 Click **Enable > Confirm**.

Note (Applicable only for WildFly cluster) If the datasource is still disabled, set the datasoure to <enabled = true> in domain.xml file and restart the domain.

Step 11 In the Connection Tab test the connection and you will see the confirmation message: "Successfully created the JDBC connection."

Step 12 Restart the WildFly server.

Configuring Data Source for Oracle in WildFly

Step 1 Log on to the WildFly Admin console (URL example below) with you credentials and click **OK**. Click on **Configuration** tab to be able to edit and make changes.

URL example:

http://<hostname>:<port>/RequestCenter

where,

<hostname> = The fully qualified domain hostname or the IP address of the computer where you installed the WildFly server for Service Catalog.

<port> =The HTTP Port number assigned to the Wildfly server for Service Catalog. The default value for HTTP Port number is 8080.

Step 2 Click **Add** to add a new data source.

Step 3 Enter **Name**: **SERVICECATALOGDS** and **JNDI Name: java:/<Name of the data source>**

Step 4 Click **Next**

Step 5 Select driver as "**oracle-thin"** and click **Next**.

Step 6 Enter the connection URL:
jdbc:oracle:thin:@//<db_server>:1433;DatabaseName=VM236_RCDB_RC4

Step 7 Enter your credentials.

Step 8 Fill up the various fields under different tabs as mentioned in the table below:

Tab Name	Fieldname	Value
Security	Secure Domain	CiscoSecureDataSource
Connection	**Use JTA** and **Use CCM**	Both these options should be checked.
Properties	1. SelectMethod: 2. sendStringParametersAsUnicode:	1. Direct 2. True

Pool	Minimum size is 20, maximum size is 80 and other values set to False.	
Validation		
	Background Validation	False
	Validation Millis	90000
	Validate on Match	False

Step 9 Click **Enable > Confirm**.

Note (Applicable only for WildFly cluster) If the datasource is still disabled, set the datasoure to <enabled = true> in domain.xml file and restart the domain.

Step 10 In the Connection Tab test the connection and you will see the confirmation message: "Successfully created the JDBC connection."

Step 11 Restart the WildFly server.