IBM Netcool OMNIbus WebGUI 8.1

Load Balancing Configuration

A step by step example

Author: Gheorghe Mihaela, IBM NSA Software Engineer | IBM Clouds Lab
Mihaela.Gheorghe1@ibm.com

Description

This guide has the purpose to illustrate a complete step by step example for a load balancing configuration for IBM Netcool OMNIbus WebGUI.

The steps described within this document are applicable for environments with DASH version 3.1.2 and higher. For creating this document, the tests were performed within an environment with WebGUI 8.1 Fix Pack 15 , DASH 3.1.3.2. and DB2 11.1.

They can be tested against any WebGUI 8.1.x environments as long as the DASH version is at least 3.1.2. and the installed DB2 is supported.

All the servers that will be part of the cluster MUST have the exact same versions and components installed.

Additional references:

https://www.ibm.com/support/knowledgecenter/en/SSSHTQ_8.1.0/com.ibm.netcool_OMNIbus.doc_8.1.0/webtop/wip/concept/web_ovr_loadbalancingcluster.html

https://www-01.ibm.com/support/docview.wss?uid=swg21983344

Configuration needed on the DB2 server

Login to DB2 with the DB2 instance owner user, in this example the default **db2inst1** user has been used.

Start DB2 database by running the following command: **db2start**

Create an empty database, you can name it for example **DASHDB**

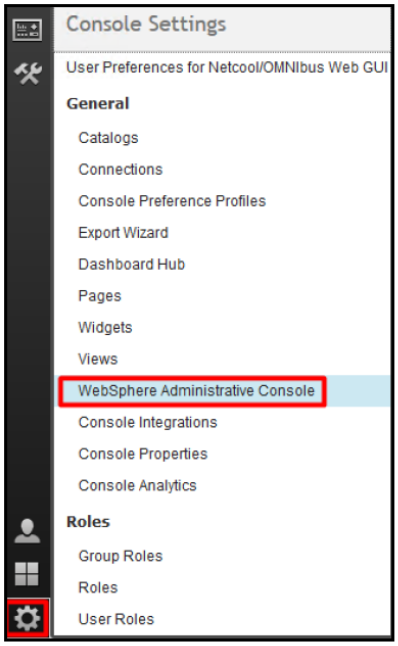
db2 create database DASHDB
connect to DASHDB: **db2 connect to DASHDB**

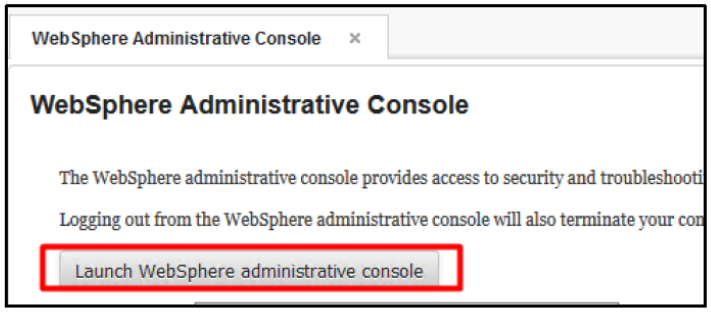
```
[db2inst1@thriver1 ~]$ db2 create database DASDB
DB20000I  The CREATE DATABASE command completed successfully.
[db2inst1@thriver1 ~]$ db2 connect to DASHDB

   Database Connection Information

 Database server        = DB2/LINUXX8664 10.5.0
 SQL authorization ID   = DB2INST1
 Local database alias   = DASHDB
```

Configuration needed on each WebGUI server

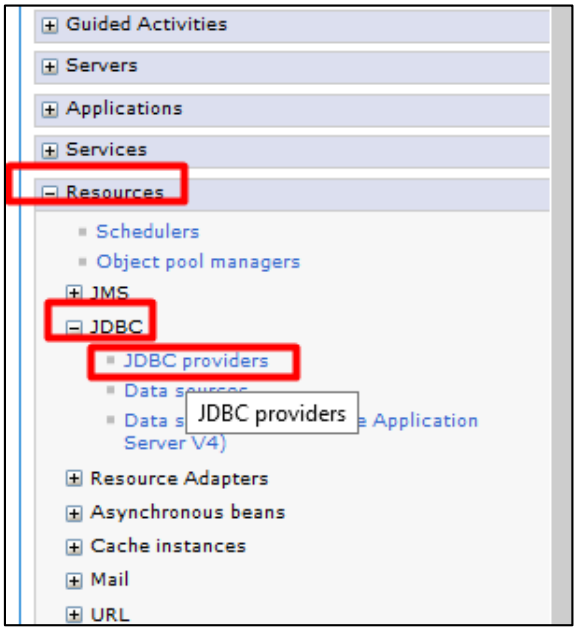
On the first WebGUI server:

1. Login to the WebGUI server and open WebSphere Administrative Console

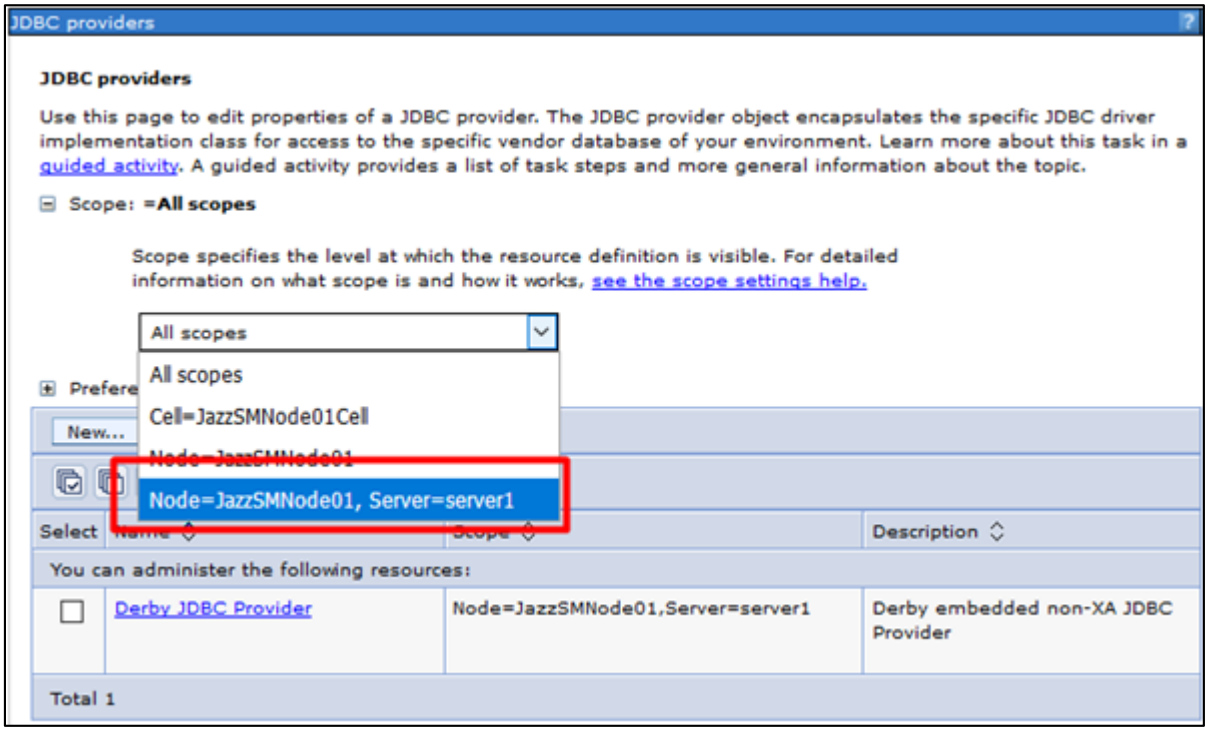




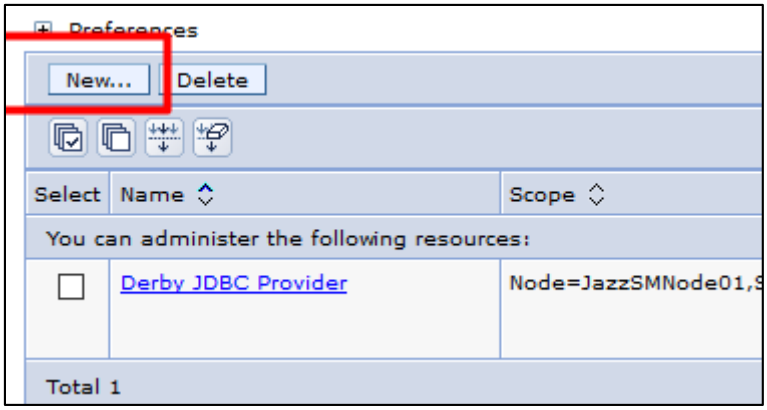
2. From WAS go to Resources -> JDBC -> JDBC providers



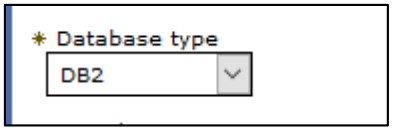
3. Select instead of "All scopes" the option Node=JazzSMNode01, Server=server1 :



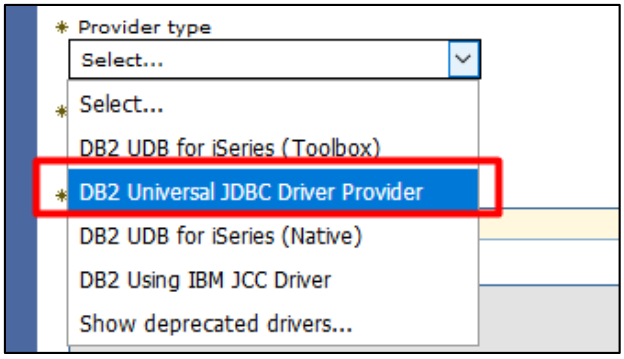
4. Create a new JDBC provider by clicking on the New option:



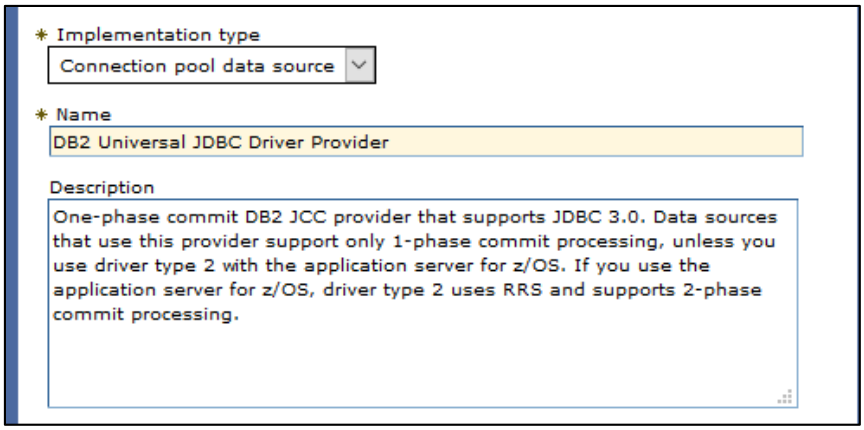
Select DB2 for database type:



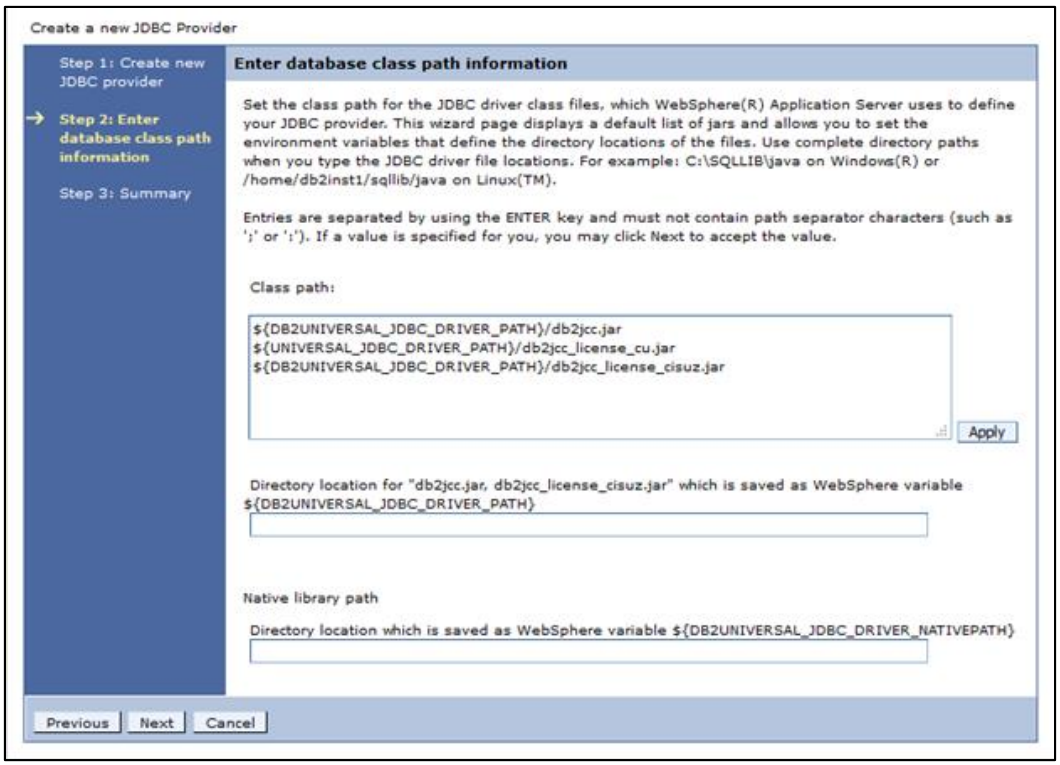
For provider type select DB2 universal JDBC driver provider:



For implementation type select connection pool data source:



Click **next.**



On the server search for **db2jcc.jar** file paths. There should be one under JazzSM directory which is required for native library path and one under WebSphere directory which is required for the first field.

```
[root@thriver1 linux_x86_64]# locate db2jcc.jar
/Miha/opt/IBM/JazzSM/lib/db2/db2jcc.jar
/Miha/opt/IBM/WebSphere/AppServer/deploytool/itp/plugins/com.ibm.datatools.db2_2.1.110.v20121008_1514/driver/db2jcc.jar
[root@thriver1 linux_x86_64]#
```

Enter the following path to the directory location for the mentioned jar files:

/Miha/opt/IBM/WebSphere/AppServer/deploytool/itp/plugins/com.ibm.datatools.db2_2.1.110.v20121008_1514/driver

Directory location for "db2jcc.jar, db2jcc_license_cisuz.jar" which is saved as WebSphere variable ${DB2UNIVERSAL_JDBC_DRIVER_PATH}

AppServer/deploytool/itp/plugins/com.ibm.datatools.db2_2.1.110.v20121008_1514/driver

And the following path for the native directory:

/Miha/opt/IBM/JazzSM/lib/db2

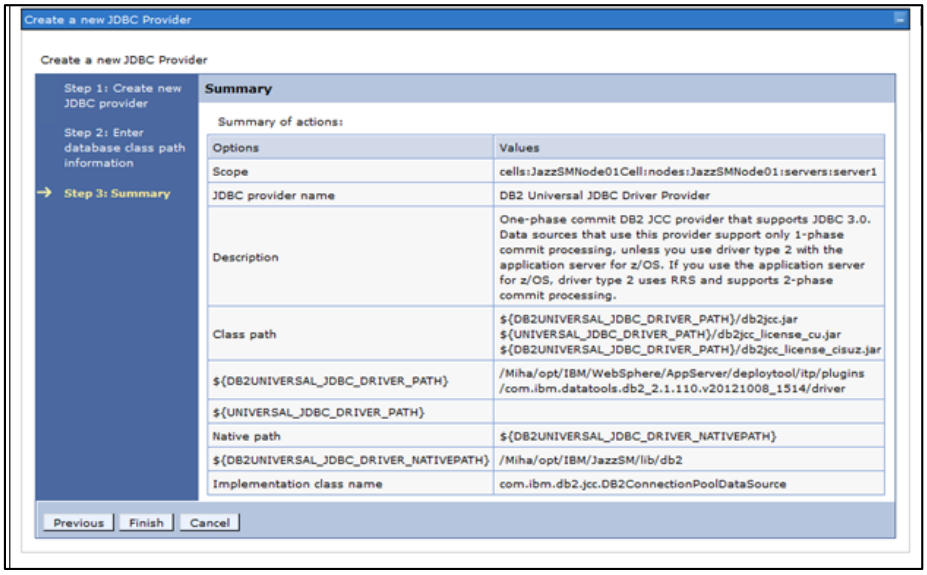
Native library path

Directory location which is saved as WebSphere variable ${DB2UNIVERSAL_JDBC_DRIVER_NATIVEPATH}

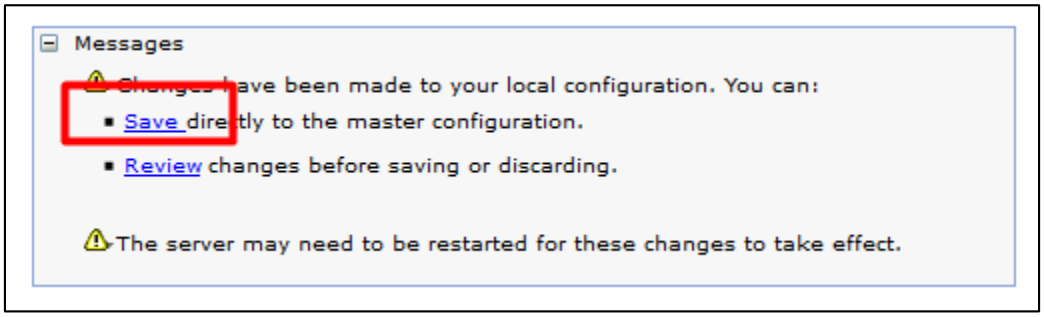
/Miha/opt/IBM/JazzSM/lib/db2

Click **next.**

Click **finish**.

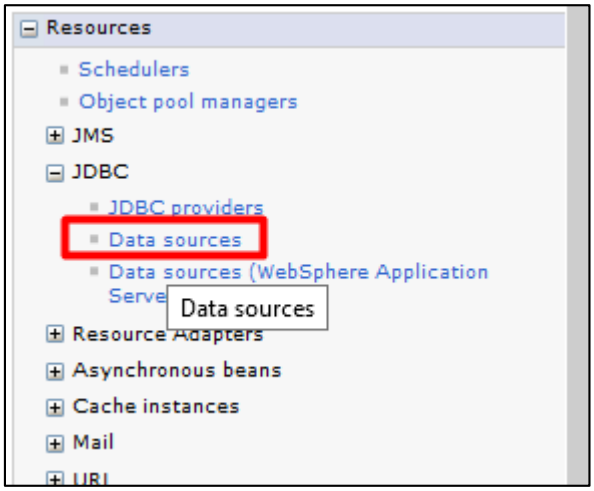


Click ***Save*** to save the configuration (you will need to do this each time you get this screen):

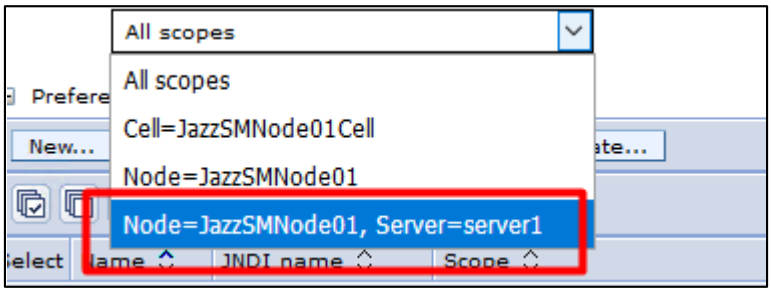


5. Create a new Data Source for JDBC.

Go to "Resources" -> JDBC -> Data Sources

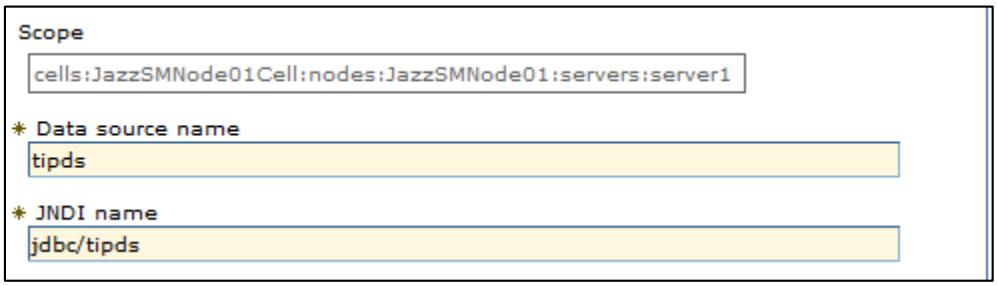


Select instead of "All scopes" the option Node=JazzSMNode01, Server=server1 :



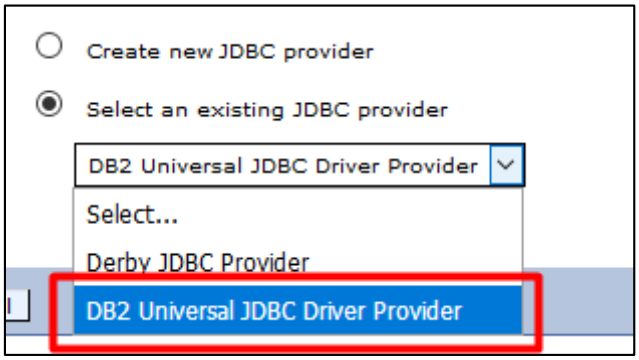
Click on "**New**"

Enter **tipds (**this should be always named as this) and **jdbc/tipds** for JNDI name **(**this should be always named as this):

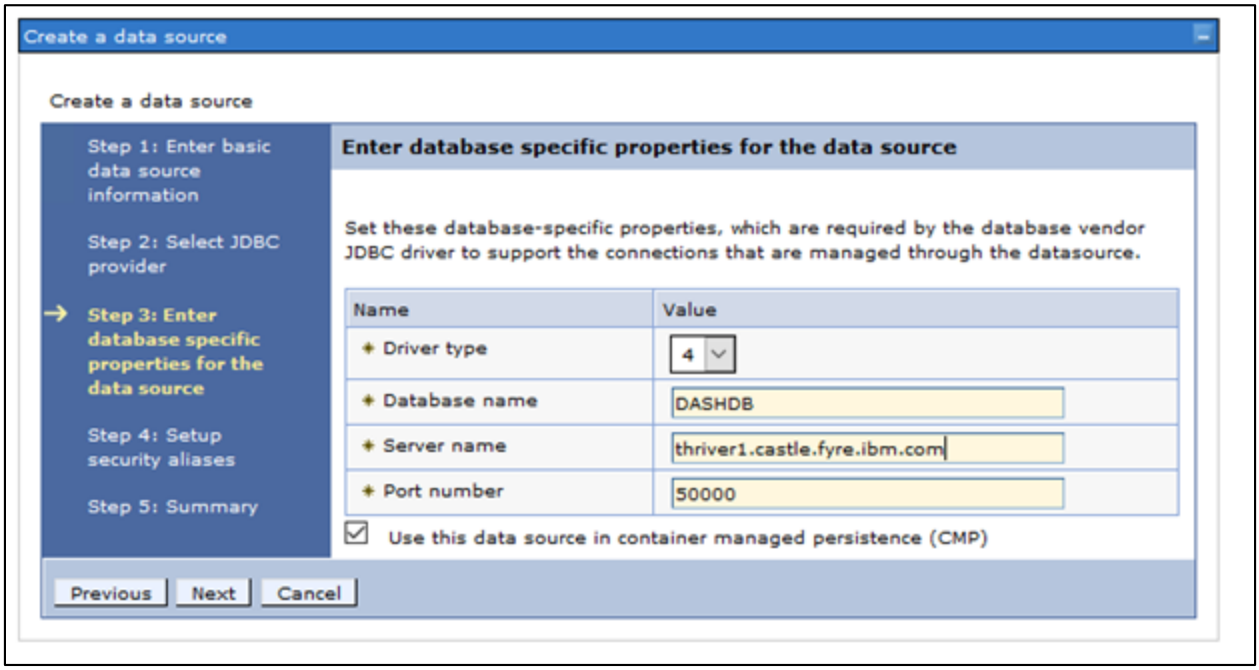


Click on **Next**.

Select the option "**Select an existing JDBC provider**" and select the "**DB2 universal JDBC driver provider"**:

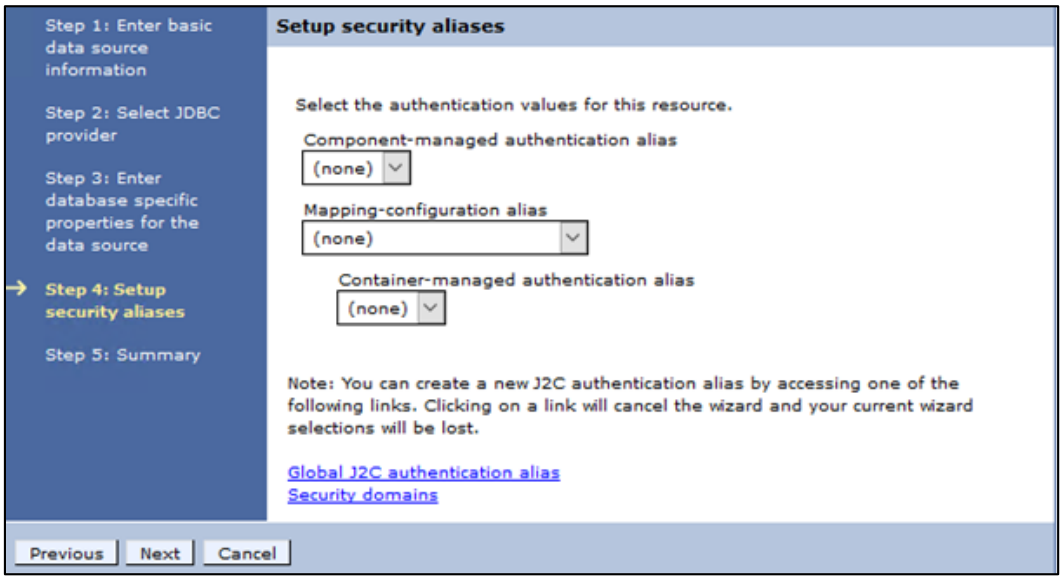


Click on **Next**.



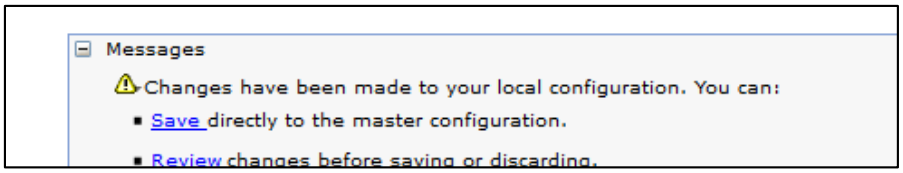
Within this screen you will have to enter the name of the database that you have created e.g. **DASHDB** and also the server hostname and the port number where DB2 is installed.

Click **next.**

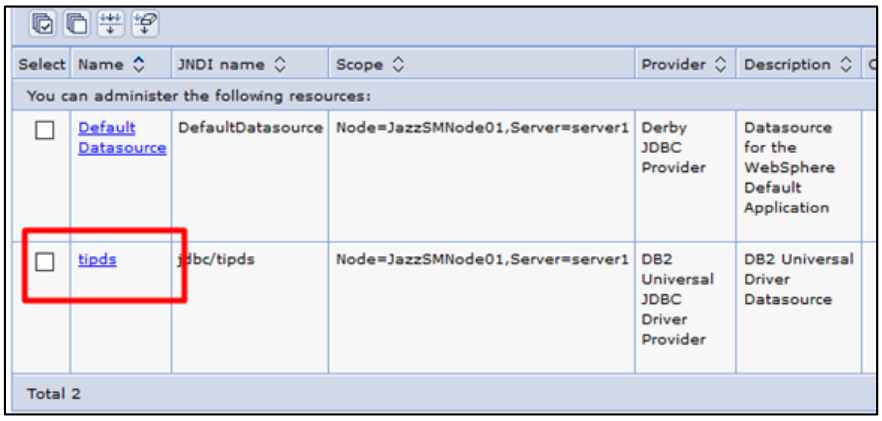


Within this screen you don't have to select anything, we'll complete this later.

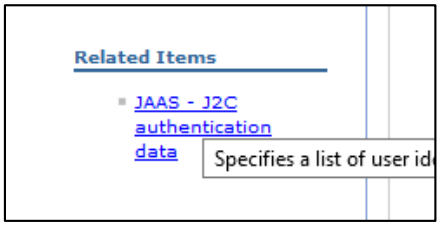
Click **next.**
Click **finish.**
Click ***Save*** to store the configuration



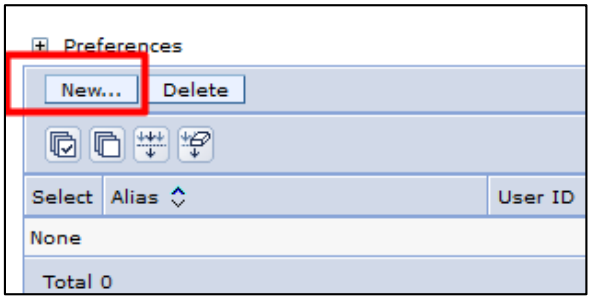
6. Click on the data source that was created e.g. "**tipds**":



Select "**JAAS - J2C authentication data**" under the **Related Items** section.

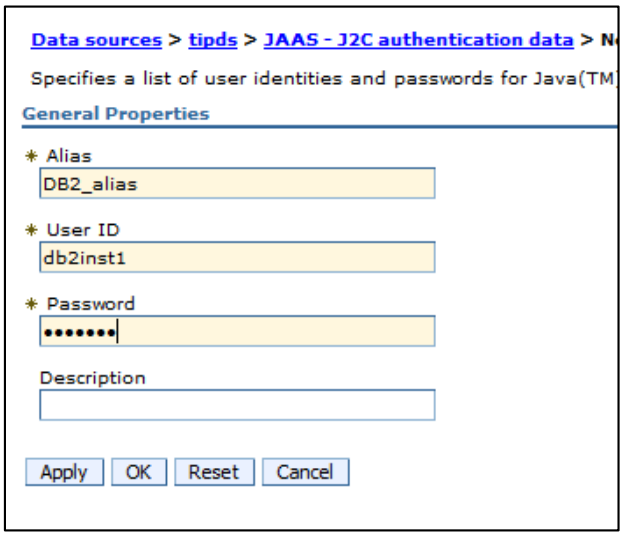


Click on **new:**

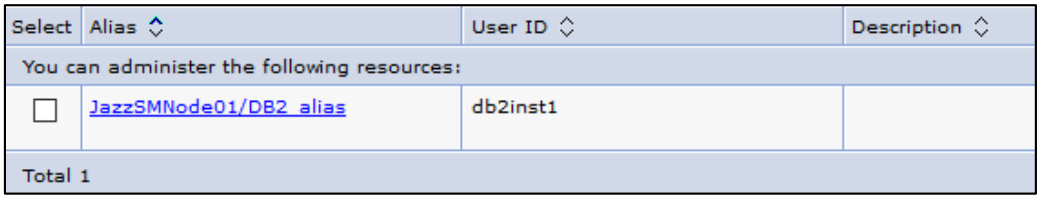


Enter a name as alias – in this example the following name was used: **DB2_alias**

Enter the **db2inst1** user (the instance owner user from DB2) and its password.

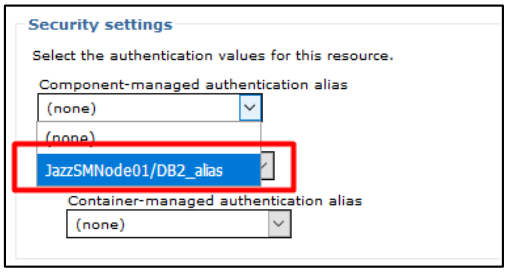


Click **ok.**
Save the configuration.

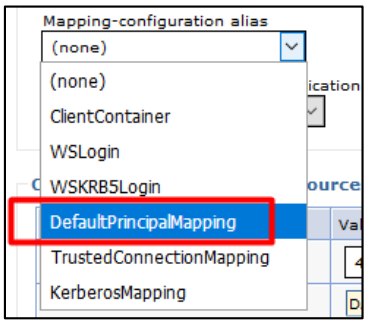


7. Return to the **tipds** data source and go to **Security Settings** section:

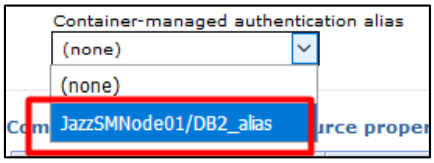
Select **JazzSMNode01/DB2_alias** for component-managed authentication alias:



Select **DefaultPrincipalMapping** for mapping-configuration alias:

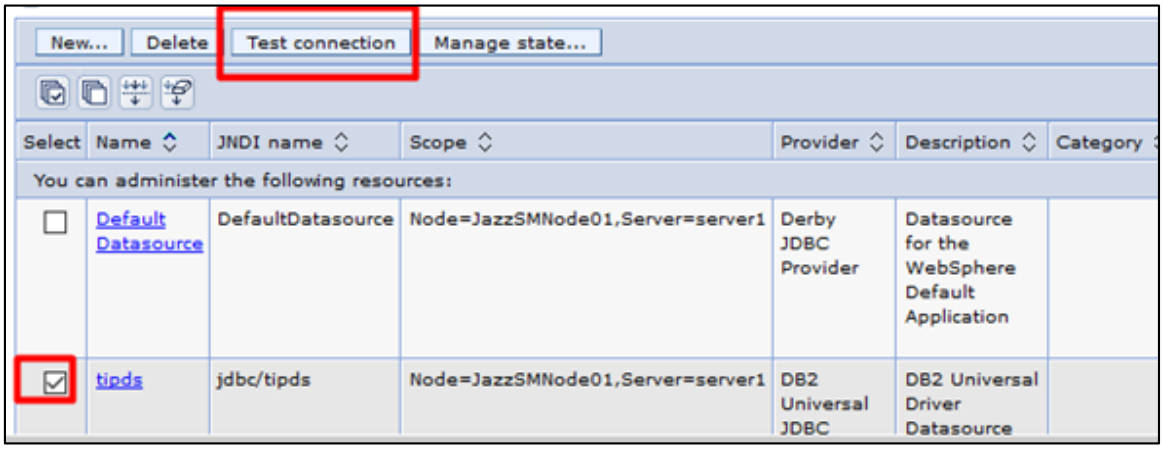


Select **JazzSMNode01/DB2_alias** for container-manager authentication alias:

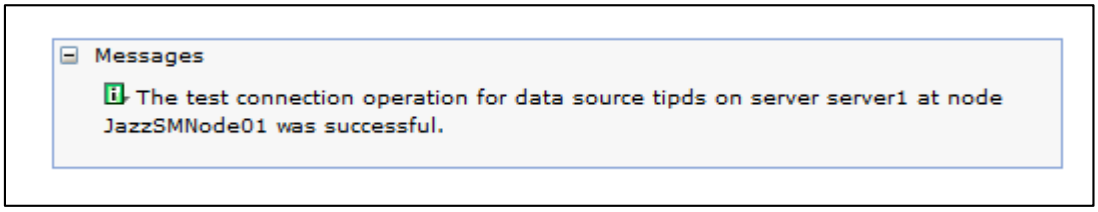


Click **ok.**
Click **Save** to store the configuration.

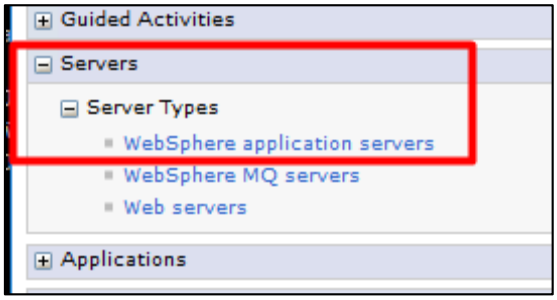
8. Check **tipds** data source connection:



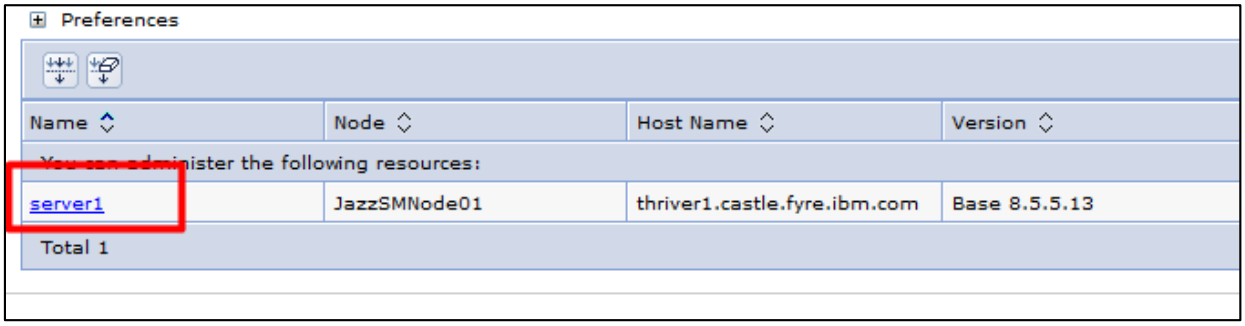
The output should be the below one:



9. From WAS menu -> Servers -> Server Types -> WebSphere application servers

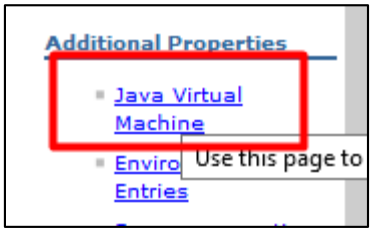


Click on **server1**:



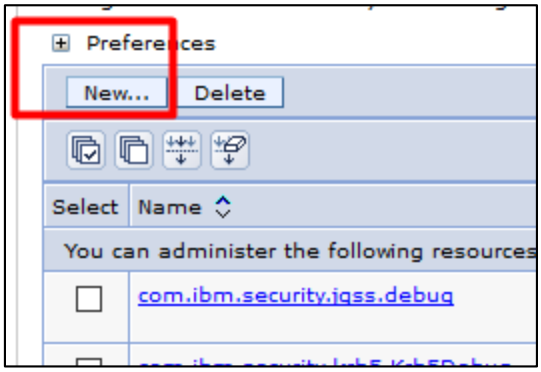
Under **Server Infrastructure** menu-> **Java and Process Management** => **Process Definition**

Click on **Java Virtual Machine** under the **Additional Properties** section:

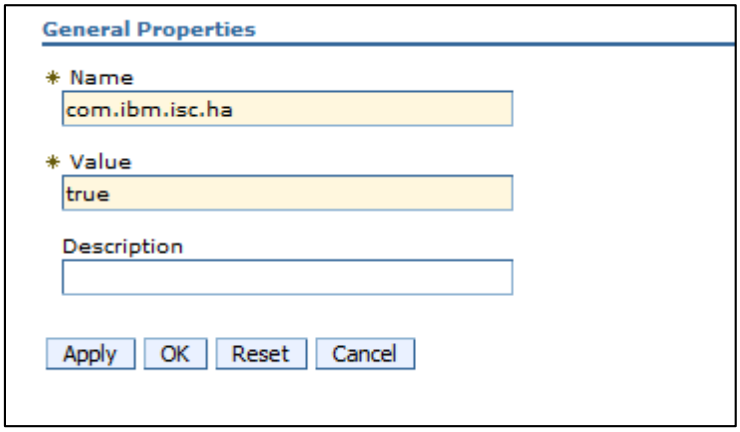


Click on **Custom Properties** under the **Additional Properties** section:

Click on **New**:



Enter **com.ibm.isc.ha** for the Name property and **true** for the Value property:



Click **apply** and **save**.

10. On the server, edit the **server.init** file from the webgui **etc** directory and set the following 3 properties as per above:

cluster.mode:on

timedtasks.enabled:true

cluster.hostname:<server_hostname>

Make sure to add the correct hostname of your WebGUI/DASH server for each server.

Afterwards, you will need to restart webgui.

Then on the webgui server run the following command:

./consolecli.sh ListHANodes --username smadmin --password netcool

You should get your webgui server on the list.

Repeat all the above steps from 1 to 10 on all the other WebGUI servers that you want to add to this cluster setup.

Afterwards, with both servers configured you will need to enable server to server trust by following the steps described within the following link:

https://www.ibm.com/support/knowledgecenter/en/SSEKCU_1.1.2.1/com.ibm.psc.doc/tip_original/ttip_config_loadbal_trust.html

e.g. repeat the below steps from 1 to 5 for each WebGUI server:

1. Edit ssl.client.props properties file

/Miha/opt/IBM/JazzSM/profile/properties/ssl.client.props

Uncomment the section that starts with **com.ibm.ssl.alias=AnotherSSLSettings** so that it looks like this:

```
#-------------------------------------------------------------------------
# Another SSL configuration (this is a template, uncomment and modify)
# You can configure the dynamicSelectionInfo OR reference this alias
# from another protocol (e.g., soap.client.props or sas.client.props)
#-------------------------------------------------------------------------
com.ibm.ssl.alias=AnotherSSLSettings
com.ibm.ssl.protocol=SSL_TLSv2
com.ibm.ssl.securityLevel=HIGH
com.ibm.ssl.trustManager=IbmX509
com.ibm.ssl.keyManager=IbmX509
com.ibm.ssl.contextProvider=IBMJSSE2
com.ibm.ssl.enableSignerExchangePrompt=true
#com.ibm.ssl.keyStoreClientAlias=default
#com.ibm.ssl.customTrustManagers=
#com.ibm.ssl.customKeyManager=
#com.ibm.ssl.dynamicSelectionInfo=
#com.ibm.ssl.enabledCipherSuites=
```

2. Uncomment the section that starts with **com.ibm.ssl.trustStoreName=AnotherTrustStore** so that it looks like this:

```
# TrustStore information
com.ibm.ssl.trustStoreName=AnotherTrustStore
com.ibm.ssl.trustStore=${user.root}/etc/trust.p12
com.ibm.ssl.trustStorePassword={xor}CDo9Hgw=
com.ibm.ssl.trustStoreType=PKCS12
com.ibm.ssl.trustStoreProvider=IBMJCE
com.ibm.ssl.trustStoreFileBased=true
com.ibm.ssl.trustStoreReadOnly=false
```

3. Update the location of the trust store that the signer should be added to in the **com.ibm.ssl.trustStore** property of **AnotherTrustStore** by replacing the default value **com.ibm.ssl.trustStore=${user.root}/etc/trust.p12** with the correct path for your trust store. Example:

```
# TrustStore information
com.ibm.ssl.trustStoreName=ClientDefaultTrustStore
com.ibm.ssl.trustStore=${user.root}/config/cells/JazzSMNode01Cell/nodes/JazzSMNode01/trust.p12
com.ibm.ssl.trustStorePassword={xor}CDo9Hgw=
com.ibm.ssl.trustStoreType=PKCS12
com.ibm.ssl.trustStoreProvider=IBMJCE
com.ibm.ssl.trustStoreFileBased=true
com.ibm.ssl.trustStoreReadOnly=false
```

com.ibm.ssl.trustStore=${user.root}/config/cells/JazzSMNode01Cell/nodes/JazzSMNode01/trust.p12

4. Save file.
5. Restart webgui.

Repeat the same steps from 1 to 5 on the other servers.

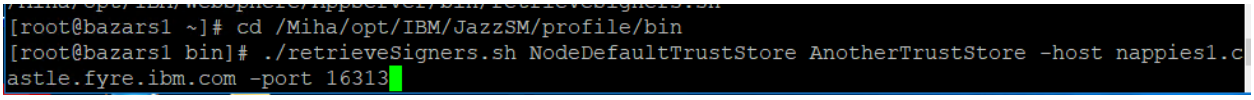
Afterwards, run the following command on each node for each *myremotehost* (that is, for every node that you want to enable trust with) in the cluster.

JazzSM_WAS_Profile/bin/retrieveSigners.sh NodeDefaultTrustStore AnotherTrustStore -host **myremotehost** -port **remote_SOAP_port**

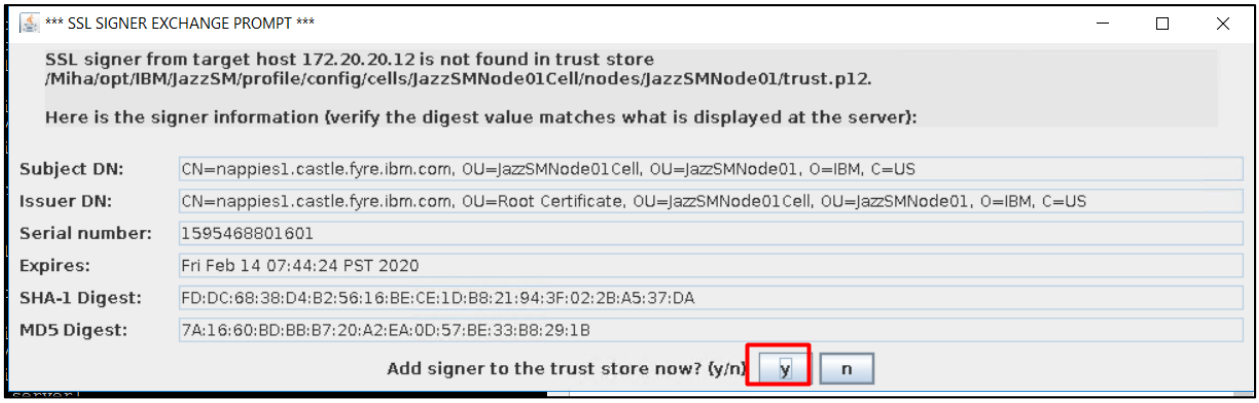
where:
myremotehost is the name of the server to enable trust with;
remote_SOAP_port is the SOAP connector port number (16313 is the default). If you have installed with non-default ports, check *JazzSM_WAS_Profile*/properties/portdef.props for the value of SOAP_CONNECTOR_ADDRESS and use that.

So, on server 1: run the following command – the host added in command line is the one from the second server:

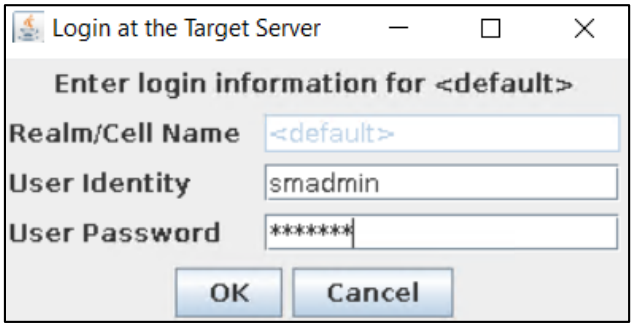
./retrieveSigners.sh NodeDefaultTrustStore AnotherTrustStore -host **loaf1.castle.fyre.ibm.com** -port 16313



Click **yes** to add the signer to the trust store:



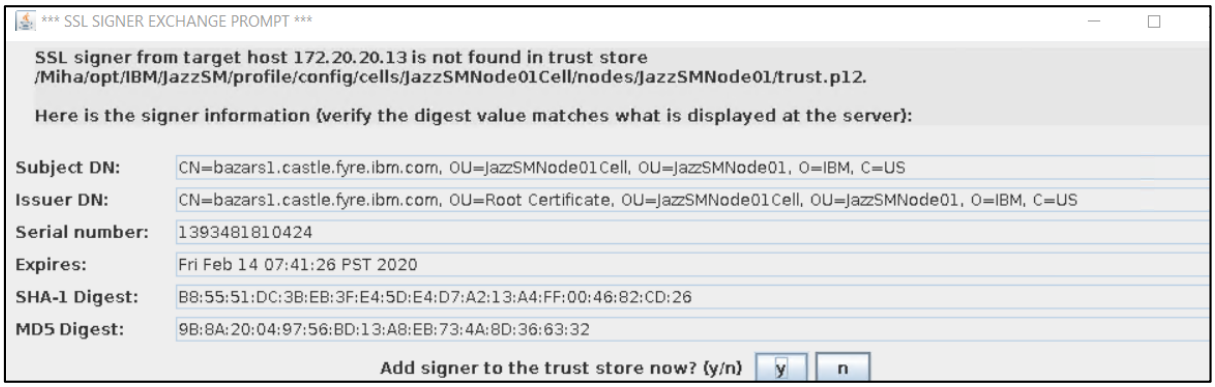
Enter **smadmin** credentials and click **ok**:

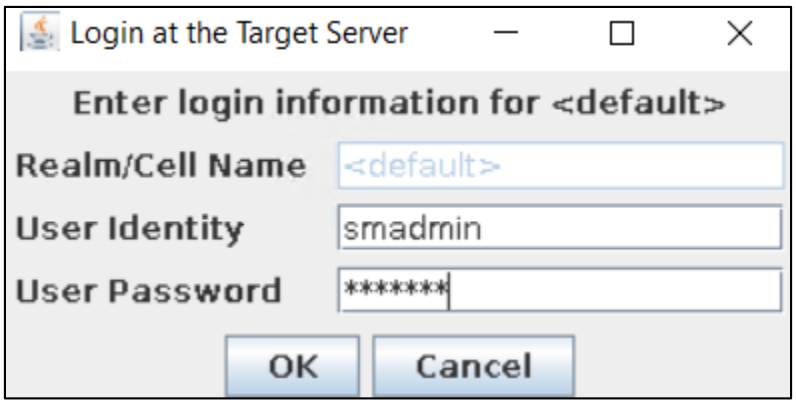


On server 2 – run the same but add the host of the webgui **server1**, example:

./retrieveSigners.sh NodeDefaultTrustStore AnotherTrustStore -host bazarz1.castle.fyre.ibm.com -port 16313

```
[root@nappies1 bin]# cd /Miha/opt/IBM/JazzSM/profile/bin
[root@nappies1 bin]# ./retrieveSigners.sh NodeDefaultTrustStore AnotherTrustStore -host bazars1.ca
stle.fyre.ibm.com -port 16313
```





Restart all webgui servers again.

At the end, you will have your HA environment configured.

Check status by running on each webgui server the following command:

./consolecli.sh ListHANodes –username smadmin –password netcool

```
[root@bazars1 bin]# /Miha/opt/IBM/JazzSM/ui/bin/consolecli.sh ListHANodes --username smadmin --pa
ssword netcool
NodeName                        NodeStatus      NodeSync        NodeVersion
bazars1.castle.fyre.ibm.com:16311               ACTIVE          InSync          3.1.3.0

nappies1.castle.fyre.ibm.com:16311              ACTIVE          InSync          3.1.3.0



CTGWA4017I The command completed successfully.
```

Hope you'll find this useful for your HA configuration!