Guided Exercise: Configuring Host Controllers

In this lab work, you will create two new host controllers to be slaves to the previously created domain controller.

Resources	
Files:	/home/student/JB248/labs/host
App URL:	http://172.25.250.254:8230/ http://172.25.250.254:9080/ http://172.25.250.254:8080/

Results

Must be able to boot from host controllers slaves of a domain controller.

Before you begin

Before you begin the guided exercise, run the following command to verify that EAP was installed in /opt/jboss-eap-7.0, that no EAP instances are running, and that the previous guided exercise established a domain controller in /home/student/JB248/labs/domain:

```
[student@workstation ~]$ lab configure-host setup
```

1. Create additional base directories for two new host controllers.

1.1. In the /home/student/JB248/labs/host folder, create two new folders: machine2 and machine3.

```
[student@workstation ~]$ cd /home/student/JB248/labs/host
[student@workstation host]$ mkdir machine2
[student@workstation host]$ mkdir machine3
```

1.2. Copy the opt/jboss-eap-7.0/domain folder to the machine2 and machine3. These two folders represent two separate computers that will connect to the domain controller.

```
[student@workstation host]$ cp -R /opt/jboss-eap-7.0/domain machine2/domain [student@workstation host]$ cp -R /opt/jboss-eap-7.0/domain machine3/domain
```

2. Set up different hostnames.

2.1. Use a text editor to open the host-slave.xml file in the /home/student/JB248/labs/host/machine2/domain/configuration folder.

2.2. Notice that this host does not have a name. Each host in a domain must have a different name, so you must assign the value "host2" to the name attribute as follows:

```
<host name="host2" xmlns="urn:jboss:domain:4.1">
```

3. Specify the IP address of the domain controller.

3.1. Notice that host-slave.xml is configured to be a slave based on the following domain-controller value that includes the <remote> element:

```
<domain-controller>
    <remote security-realm="ManagementRealm">
        <discovery-options> <static-
            discovery name="primary" protocol="$
                {jboss.domain.master.protocol:remote}" host="$
                {jboss.domain.master.address}" port="$
                {jboss.domain.master.port:9999}"/>
        </discovery-options> </
    remote>
</domain-controller>
```

The jboss.domain.master.address system property is sent in the IP address of the domain controller when you start host2 and host3 later in this guided exercise.

4. Avoid port conflicts from multiple host controllers.

4.1. To avoid port conflicts, different ports must be set for each host controller running on the workstation virtual machine.

Since the domain controller created in the previous lab work is already bound to port 9999, you must change the native interface port number for host2. In the <native-interface> section, change the default port to 29999:

```
<native-interface security-realm="ManagementRealm"> <socket
    interface="management" port="$
        {jboss.management.native.port:29999}"/>
</native-interface>
```

4.2. Notice in all the <interfaces> elements that the IP address is assigned to 127.0.0.1, the default loopback interface.

This will not work for jboss.bind.address.management in a multi-machine environment, because the master host will not be able to connect to this host. It also won't work for jboss.bind.address if you want the servers to be accessible from the outside.

Replace the loopback IP address 127.0.0.1 in both defaults with the IP address of the workstation virtual machine, 172.25.250.254:

```
<interface name="management"> <inet-
    address value="${jboss.bind.address.management:172.25.250.254}"/>
</interface>
<interface name="public"> <inet-
    address value="${jboss.bind.address:172.25.250.254}"/>
</interface>
```

5. Observe the host servers.

5.1. In the host-slave.xml file for host2, notice that there are already two servers defined, server-one and server-two:

```
<server name="server-one" group="main-server-group"/>
        <server name="server-two" group="other-server-group"> <socket-bindings
        port-offset="150"/> </server>
```

server-one belongs to the server group named main-server-group and server-two belongs to the server group other-server-group.

Also note that server-two uses a port offset of 150 to avoid conflicts between the two servers.

Save the changes to host-slave.xml and close the text editor.

6. Start the domain controller.

6.1. Before running the host controller, start the domain controller created in the previous exercise on machine1 using a new terminal window:

```
[student@workstation ~]$ cd /opt/jboss-eap-7.0/bin
[student@workstation bin]$ ./domain.sh \
-Djboss.domain.base.dir=/home/student/JB248/labs/domain/machine1/domain/ \ --host-
config=host-master.xml
```

6.2. Start host2 using the host-slave.xml configuration file that has its management interface bound to 172.25.250.254 on port 29999.

Run the following command from your /opt/jboss-eap-7.0/bin folder in the original terminal window:

```
[student@workstation domain]$ cd /opt/jboss-eap-7.0/bin
[student@workstation bin]$ ./domain.sh \
-Djboss.domain.base.dir=/home/student/JB248/labs/host/machine2/domain/ \ --host-
config=host-slave.xml \
-Djboss.domain.master.address=172.25.250.254
```

use

Notice that the prefix of each log entry in the terminal window is either [HostController] or the name of the server that caused the log event, which is [Server:server-one] or [Server:server-two] in the implementation .

6.3. Observe the terminal window of the machine2 host controller. Check

Look closely at the log output and you should see the host controller connecting to the master host and also messages indicating that server-one and server-two are starting.

```
[Host Controller] 16:42:57,307 INFO [org.jboss.as.host.controller]
  (Controller Boot Thread) WFLYHC0148: Connected to master host controller at remote://172.25.250.254:9999

[Host Controller] 16:42:57,367 INFO [org.jboss.as.host.controller] (Controller Boot Thread) WFLYHC0023:
  Starting server server-one
```

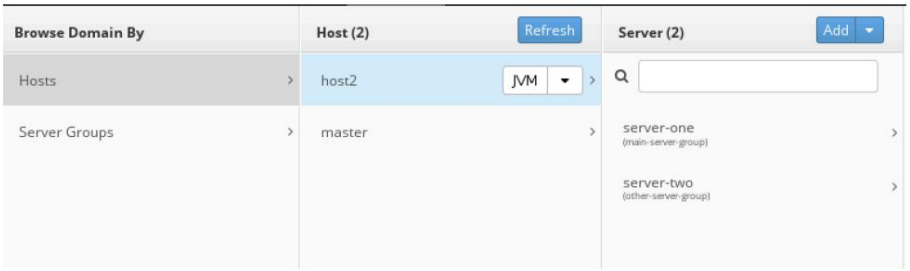
6.4. Observe the terminal window of the domain controller. must see entry log showing the connecting slave host:

```
[Host Controller] 11:42:16,348 INFO [org.jboss.as.domain.controller] (Host Controller Service Threads -
  36) WFLYHC0019: Registered remote slave host "host2", JBoss JBoss EAP 7.0.0.GA (WildFly
  2.1.2.Final-redhat-1)
```

7. Verify that host2 is in the domain.

7.1. In your browser, navigate to http://172.25.250.254:9990/, which is the URL for the domain controller administration tool. The administrator username is jbossadm and the password is JBoss@RedHat123.

7.2. Verify that server-one and server-two are listed on the Runtime page on host host2.



8. Configure the host controller host3.

8.1. Use a text editor to open the host-slave.xml file included in the /home/student/JB248/labs/host/machine3/domain/configuration folder. Make sure you leave the domain controller and the other host controller running.

8.2. Add a name attribute to the <host> element, assigning it the value host3:

```
<host name="host3" xmlns="urn:jboss:domain:4.1">
```

8.3. Change the native management interface port to 39999:

```
<native-interface security-realm="ManagementRealm"> <socket
        interface="management" port="$
                {jboss.management.native.port:39999}"/> </native-interface>
```

8.4. In the <interfaces> section, replace 127.0.0.1 in the default value of jboss.bind.address.management and jboss.bind.address with the IP address of the workstation virtual machine, 172.25.250.254:

```
<interface name="management"> <inet-
        address value="${jboss.bind.address.management:172.25.250.254}"/>
</interface>
<interface name="public"> <inet-
      address value="${jboss.bind.address:172.25.250.254}"/> </interface>
```

8.5. Remove server-one from the <servers> section.

8.6. Rename server-two to server-three and change its port-offset to 1000. Your <servers> section should appear as follows:

```
<servers>
        <server name="server-three" group="other-server-group"> <socket-bindings
              port-offset="1000"/> </server>

</servers>
```

8.7. Save the changes to host-slave.xml and exit the text editor.

9. Start host3.

9.1. Open a new terminal window and run the following commands to start host3, as a slave in your domain:

```
[student@workstation domain]$ cd /opt/jboss-eap-7.0/bin [student@workstation
bin]$ ./domain.sh \ -Djboss.domain.base.dir=/home/
student/JB248/labs/host/machine3/domain/ \ --host-config=host-slave.xml \
-Djboss.domain.master.address=172.25.250.254
```

9.2. Observe the terminal window of the domain controller. must see entry log showing the connecting slave host:

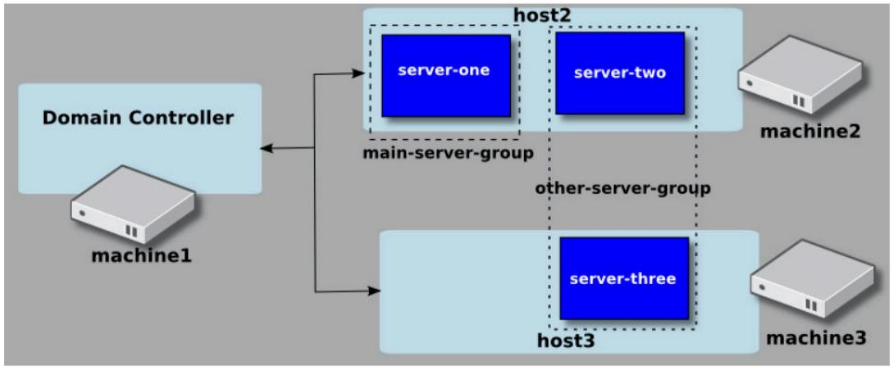
```
[Host Controller] 01:34:29,845 INFO [org.jboss.as.domain.controller] (Host Controller Service Threads -
 35) WFLYHC0019: Registered remote slave host "host3", JBoss JBoss EAP 7.0.0.GA (WildFly
 2.1.2.Final-redhat-1)
```

10. Verify that host3 is in the domain.

10.1. Go back to the management console and refresh the Runtime page.

10.2. Verify that server-three is running on host3.

10.3. The following graph shows how the domain now looks:



domain servers

server	Address	port offset
server-one	172.25.250.254:8080	0
server-two	172.25.250.254:8230	150
server-three	172.25.250.254:9080	1000

11. Verify that the servers are up and running.

11.1. In the workstation virtual machine, open a web browser and go to http://172.25.250.254:8080/. You should see the default EAP welcome page that is delivered by server-one.

11.2. Point your web browser to http://172.25.250.254:8230/. Again, you should see the default EAP welcome page, but this time, the page is served by server-two.

11.3. Point your web browser to http://172.25.250.254:9080/. Again, you should see the default EAP welcome page, but this time, the page is served by server-three.

12. Stop a host controller.

12.1. Press Ctrl+C in the host2 terminal window, which will start the host2 shutdown process

12.2. Look at the output in the terminal window and notice that the server-one and server-two processes stop, followed by the host controller process.

```
19:47:21,305 INFO [org.jboss.as.process.Host Controller.status] (reaper for Host Controller)
  WFLYPC0011: Process 'Host Controller' finished with an exit status of 130

19:47:21,305 INFO [org.jboss.as.process] (Shutdown thread) WFLYPC0016: All processes finished;
  exiting
```

12.3. Observe the terminal window of the domain controller. You should see a log event similar to the following, indicating that host2 has been removed from the managed domain:

```
[Host Controller] 01:42:49,331 INFO [org.jboss.as.domain.controller] (management task-7)
  WFLYHC0026: Unregistered remote slave host "host2"
```

12.4. Restart host2. Once started, you should see a registration event in machine1's terminal window showing that host2 registers again with the domain.

```
[student@workstation bin]$ ./domain.sh \
-Djboss.domain.base.dir=/home/student/JB248/labs/host/machine2/domain/ \ --host-
config=host-slave.xml \
-Djboss.domain.master.address=172.25.250.254
```

13. Stop the domain controller.

13.1. Press Ctrl+C in machine1's terminal window to shutdown the controller Of domain.

13.2. Refresh the web page at http://172.25.250.254:8080/. The page should look fine, even though the master host controller is no longer working.

13.3. Reboot the domain controller:

```
[student@workstation bin]$ ./domain.sh \
-Djboss.domain.base.dir=/home/student/JB248/labs/domain/machine1/domain/ \ --host-
config=host-master.xml
```

13.4. Observe the output in the terminal window and wait. You should see host2 and host3 reconnect to the master controller after a few seconds.

14. Perform cleaning.

14.1. Use Ctrl+C to stop the domain controller and the two host controllers.

This concludes the guided exercise.