Guided Exercise: Configuring Java Virtual Machines

In this lab assignment, you will tune the Java virtual machine settings for a stand-alone server as well as a managed domain.

Resources	
Files:	/home/student/JB248/labs/standalone /home/student/JB248/labs/domain/machine1/domain/ /home/student/JB248/labs/host/machine2/domain/ /home/student/JB248/labs/host/machine3/domain/
App URL:	ND
Resources	ND

Results

You should be able to configure the JVM settings for a standalone EAP server as well as a managed domain.

before you start

Before beginning the guided exercise, run the following command to verify that EAP was installed to /opt/jboss-eap-7.0, that no EAP instances are running, and that you have completed the previous guided exercise:

```
[student@workstation ~]$ lab jvm-settings setup
```

1. Configure JVM settings for a standalone server:

For a standalone server, the JVM settings are configured by manually editing a configuration file called JBOSS_HOME/bin/standalone.conf before starting the EAP server.

1.1. Edit the /opt/jboss-eap-7.0/bin/standalone.conf file using a text editor of your choice. Notice that you edited this file as the jboss user:

```
[student@workstation ~]$ sudo -u jboss vi /opt/jboss-eap-7.0/bin/standalone.conf
```

1.2. Notice the JVM values that are configured in the $JAVA_OPTS variable (approximately at line number 50). The minimum size (-Xms) and maximum size (-Xmx) of the JVM heap are set to 1303m (1.3 GB) by default. These default values may not be sufficient for larger applications and may cause Out of Memory (OOM) errors in the JVM, due to excessive memory consumption by the application.

It is recommended that you configure the maximum and minimum heap sizes based on the object allocation characteristics of your application. You should also tune the GC values after application profiling under production workloads.

1.3. Edit the minimum and maximum JVM sizes and increase them to 1500m (1.5 GB), as shown below:

```
JAVA_OPTS="-Xms1500m -Xmx1500m -Djava.net.preferIPv4Stack=true"
```

1.4. Start the standalone EAP server.

Use the following command to start an EAP instance using the /home/student/JB248/labs/standalone folder as the base directory:

```
[student@workstation ~]$ cd /opt/jboss-eap-7.0/bin
[student@workstation bin]$ ./standalone.sh \
-Djboss.server.base.dir=/home/student/JB248/labs/standalone/
```

1.5. Observe the console window in which EAP is starting and verify that EAP use the new JVM values:

```
    JBoss Bootstrap Environment

    JBOSS_HOME: /opt/jboss-eap-7.0

    JAVA: java

    JAVA_OPTS: -server -verbose:gc -Xloggc:"/home/student/JB248/labs/ standalone/log/gc.log"
-XX:+PrintGCDetails -XX:+PrintGCDateStamps -XX: +UseGCLogFileRotation
-XX:NumberOfGCLogFiles=5 -XX:GCLogFileSize=3M -XX:- TraceClassUnloading -Xms1500m
-Xmx1500m -Djava.net.preferIPv4Stack=true - Djboss.modules.system.pkgs=org.jboss.byteman
-Djava.awt.headless=true
```

1.6. You can also verify that the new values have been applied by running the following command to view the full arguments to the java process that represent the standalone EAP server:

```
    [student@workstation bin]$ ps -aef | grep java /usr/bin/java
    -D[Standalone] -server -verbose:gc -XX:+PrintGCDetails -XX:
+PrintGCDateStamps ...
    -XX:-TraceClassUnloading -Xms1500m -Xmx1500m -Djava.net.preferIPv4Stack=true
    ...
```

1.7. Stop the EAP instance by pressing Ctrl+C in the terminal window that is running EAP.

This concludes the standalone server's JVM configuration. In the next few steps, you will configure the JVM settings for a managed domain.

2. Perform JVM configuration for a managed domain:

The JVM settings for each server running in a managed domain can be configured using the EAP CLI or the management console at three levels:

• host controller level file (<jvm> section in host.xml)

• server group level file (<server-group> section in the domain.xml)

• server level file (<server> section in host.xml)

Remember from the class that components at lower levels can override values inherited from parent levels. It will now start the managed domain and override the JVM defaults at both the Server Group level and the Server level.

2.1. Start the domain controller (machine1) and the two host controllers (machine2 and machine3)

Start the domain controller 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
```

2.2. Start the host controller on machine2.

Run the following command from your /opt/jboss-eap-7.0/bin folder in a new terminal window on workstation to start machine2:

```
[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
```

23. Start machine3 to join the managed domain.

Run the following command from your /opt/jboss-eap-7.0/bin folder in a new terminal window on workstation to start machine3:

```
[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
```

2.4. Notice that all four servers in the managed domain have started with a minimum dynamic memory size of 64 MB and a maximum of 256 MB. These are the default JVM values for all server groups and servers in the managed domain, unless you expressly refer to the default JVM configuration in the host-slave.xml file. (Tip: use the ps -aef command and grep for the appropriate server name.)

```
[student@workstation bin]$ ps -aef | grep server-one java
-D[Server:server-one] -Xms64m -Xmx256m -Djava.awt.headless=true
...
```

2.5. Override the JVM values for server group Group1 and set the JVM heap min and max values to 1024m (1 GB).

To configure JVM settings at the Server Group level in a managed domain, you can use the EAP administration console:

Go to http://172.25.250.254:9990/, which is the location of the management console. You will be prompted to sign in. Log in as the user jbossadm with the password JBoss@RedHat123.

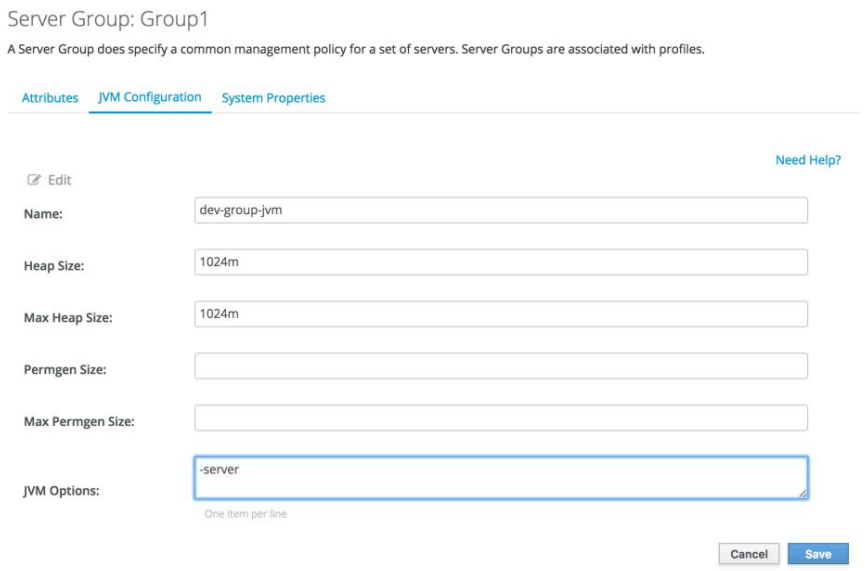
2.6. Navigate to Runtime > Server Groups > Group1 in the console administration. Click the View dropdown menu and select the Stop option to stop all servers in Group1.

2.7. When all servers in Group1 have stopped, click View next to Group1 to view the Group1 configuration page. Click the JVM Configuration tab and click the Edit link to edit the JVM settings for Group1.

2.8. Configure the JVM settings according to the following details:

- **Number: dev-group-jvm**
- **Dynamic memory size: 1024m**
- **Maximum dynamic memory size: 1024m**
- **Permgen Size: leave blank (not relevant for JDK 8)**
- **Permgen max size: leave blank (not relevant for JDK 8)**
- **JVM Options: -server**

Click Save to save your values.



2.9. Open the domain configuration file at /home/student/JB248/labs/ domain/machine1/domain/domain.xml with an editor of your choice and verify that the server group Group1 now references the dev group-jvm VM you added earlier.

```
<server-group name="Group1" profile="full-ha">
    <jvm name="dev-group-jvm"> <heap
        size="1024m" max-size="1024m"/> <jvm-options>
        <option value="-
            server"/> </jvm-options> </jvm>


      ...
  </server-group>
```

2.10. Start the servers in Group1 using the management console to verify that the changes have been applied to the JVM. Navigate to Runtime > Server Groups > Group1 in the management console. Click the View dropdown menu and select the Start option to start all servers in Group1.

Notice that the two servers in Group1 (server-one and server-three) have started with a minimum heap size of 1024 MB and a maximum of 1024 MB and the other JVM options that you provided in the previous step:

```
    [student@workstation bin]$ ps -aef | grep server-one java
    -D[Server:server-one] -Xms1024m -Xmx1024m -server -
Djava.awt.headless=true
    ...
```

The servers in Group2 (server-two and server-four) are still running with the default JVM options, since you did not override the values for Group2.

2.11 You can also configure JVM settings at the individual server level, which will override JVM settings inherited from the Host level or the Server Group level.

You should now set the server-three server JVM heap minimum and maximum sizes to 1200m (1.2 GB) and set the AggressiveOpts flag to allow the JVM to perform additional code optimizations at runtime (this flag is not set). set by default).
To do this, you can use the EAP CLI.

Run the EAP CLI in a new terminal window:

```
[student@workstation ~]$ cd /opt/jboss-eap-7.0/bin
[student@workstation bin]$ ./jboss-cli.sh --connect --
controller=172.25.250.254:9990
```

Change the server-three server JVM settings using the CLI command:

```
[[domain@172.25.250.254:9990 /] /host=host3/server-config=server-three\ /jvm=server-three-
jvm:add\ (heap-size=1200m,max-
heap-size=1200m,jvm-options=["-XX:+AggressiveOpts"])) {

        "outcome" => "success",
        ...
                "response-headers" =>
                        { "operation-requires-restart" => true, "process-
                        state" => "restart-required"
                }

}
```

Because you changed the JVM values for server-three, the server must be restarted:

```
[[domain@172.25.250.254:9990 /] /host=host3\ /server-
config=server-three:restart(blocking=true) {

        "outcome" => "success", "result"
        => "STARTED"
}
```

2.12.Open the host configuration file for host3 at /home/student/JB248/labs/host/machine3/ domain/host-slave.xml with an editor of your choice and verify that server-three now defines a section of jvm named server-three-jvm according to the following.

```
<jvm name="server-three-jvm"> <heap
        size="1200m" max-size="1200m"/> <jvm-options>
```

```
            <option value="-XX:+AggressiveOpts"/> </jvm-
        options> </jvm>
```

Also notice that only server-three has started with the new JVM settings, while server-one is still running with the JVM settings inherited from the Server Group level settings for Group1:

```
[student@workstation bin]$ ps -aef | grep server-three java
-D[Server:server-three] -Xms1200m -Xmx1200m -XX:+AggressiveOpts -
Djava.awt.headless=true
...
```

```
[student@workstation bin]$ ps -aef | grep server-one java
-D[Server:server-one] -Xms1024m -Xmx1024m -server -
Djava.awt.headless=true
...
```

3. Perform cleaning.

3.1. Exit the EAP CLI:

```
[standalone@localhost:9990 /] exit
```

3.2. Stop the managed domain by pressing Ctrl+C in all three windows of terminal on which you started EAP (machine1, machine2, and machine3), or use the EAP CLI to shut down the drivers, as demonstrated in previous lab work.

This concludes the guided exercise.

Lab Work: Configuring the Java Virtual Machine

In this lab assignment, you will configure the Java Virtual Machine (JVM) for an EAP managed domain.

Resources	
Files	/opt/domain
app url	ND

Result

You must be able to configure the Java virtual machine used by servers running in a managed domain.

before you start

Use the following command to download the relevant lab files and ensure that the managed domain is set correctly:

```
[student@workstation ~]$ lab jvm-lab-final setup
```

You can use the EAP 7 management console or the JBoss EAP CLI to achieve your goals, keeping in mind that the EAP CLI is the preferred option in production environments.

An EAP administrator has configured a managed domain with two host controllers running the servera and serverb virtual machines, respectively, and the domain controller on the workstation. The domain and host configuration files are stored in the /opt/domain folder on all three machines. You must start the managed domain and configure the Java virtual machine settings on it.

1. Start the domain controller in the workstation virtual machine. Because the domain controller configuration files are kept in the /opt/domain folder on workstation, use /opt/domain as the value of the jboss.domain.base.dir argument that you pass to the domain.sh startup script. Also note that the host file for the domain controller is named host-master.xml and is located in the /opt/domain/configuration folder. (Tip: Pass the --host-config=host-master.xml argument to domain.sh.)

Note that the /opt/domain directory is owned by the jboss user, so you must start the domain controller using sudo -u jboss /opt/jboss-eap-7.0/ bin/domain.sh ...

2. The two host controllers on servera and serverb connect to the host controller domain in the previous step and get the latest configuration of the domain. Start the two host controllers on servera and serverb.

2.1. Start the host controller on servera. Because the host controller configuration files are kept in the /opt/domain folder on servera, use /opt/domain as the value of the jboss.domain.base.dir argument that you pass to the domain.sh startup script. Also notice that the host file

for the host controller it is named host-slave.xml and is located in the /opt/domain/configuration folder. (Hint: Pass the --host config=host-slave.xml argument to domain.sh.)

Note that the /opt/domain directory is owned by the jboss user, so start the host controller using sudo -u jboss /opt/jboss eap-7.0/bin/domain.sh ...

2.2. Start the host controller on serverb. Because the host controller configuration files are kept in the /opt/domain folder on serverb, use /opt/domain as the value of the jboss.domain.base.dir argument that you pass to the domain.sh startup script. Also notice that the host file for the host controller is named host-slave.xml and is located in the /opt/domain/configuration folder. (Hint: Pass the --host config=host-slave.xml argument to domain.sh.)

Note that the /opt/domain directory is owned by the jboss user, so start the host controller using sudo -u jboss /opt/jboss eap-7.0/bin/domain.sh ...

23. Verify that both host controllers connect to the domain controller and form a managed domain. Look at the console window in which you started the domain controller and verify that both servera and serverb are registered as slaves to the domain controller.

3. You have been instructed by the development team to configure the JVM for the domain managed with the following specifications:

- The minimum and maximum heap size of the default JVM in the servera host controller should be set to 512 MB.
- For all servers in the Group1 server group, create a new JVM configuration named group1-jvm. Set the minimum JVM heap size to 512m (512MB) and the maximum to 1024m (1GB). Enable the -server flag for this server group.
- For server serverb.2, create a new JVM configuration named serverb.2-jvm, which overrides the legacy configuration. Set the minimum and maximum JVM heap size to 1024m (1 GB). Enable the -XX:+AggressiveOpts and -server flags for this server.
- Leave all other JVM settings at their default values.

Make sure that all servers in the managed domain start with these settings and without errors. Check the JVM flags with which the servers are running, using the ps -aef command on the respective hosts.

use

When you start the managed domain at the beginning of the lab, all servers are configured to start automatically. Be sure to STOP all servers before making changes to the JVM configuration in the managed domain.

3.1. Start the EAP CLI or the management console to configure the JVM.

3.2. Stop the servers in the Group1 server group.

3.3. Stop the servers in the Group2 server group.

3.4. Set the default JVM for the host controller to servera use the minimum amount of dynamic memory at 512 MB. The expected result is: Set the maximum amount of dynamic memory to 512 MB The expected result is:

3.5. Configure the minimum amount of JVM memory for the pool Group1 at 512 MB and the maximum amount of memory at 1024 MB. Also send the -server parameter to the JVM expected result is:

3.6. Configure the JVM for the server serverb.2. The minimum amount of memory for the server must be 1024 MB and the maximum amount of memory must be 1024 MB. Also send -XX:+AggressiveOpts and -server as parameters to the JVM .

3.7. Start the servers in Group1.

3.8. Start the servers in Group2.

3.9. Verify the startup of the servers with the JVM configured, using the ps command in each host.

4. Shut down servers, server groups, host controllers, and the entire domain managed.

4.1. Stop all servers in Group1.

4.2. Stop the servers in Group2.

4.3. Close the host controller on servera. Observe the servera console window and verify that the host controller has been shut down.

4.4. Close the host controller in serverb. Observe the serverb console window and verify that the host controller has been shut down.

5. Perform cleaning and grading.

5.1. Press Ctrl+C to stop the domain controller. (Alternatively, you can shut down the domain controller using the JBoss EAP CLI command / host=master:shutdown()).

5.2. Press Ctrl+C to exit the EAP CLI if you used the CLI in your lab work. (Alternatively, you can leave the CLI by typing exit.)

5.3. Run the following command on workstation to grade the assignment:

```
[student@workstation bin]$ lab jvm-lab-final grade
```

This concludes the lab work.

Solution

In this lab assignment, you will configure the Java Virtual Machine (JVM) for an EAP managed domain.

Resources	
Files	/opt/domain
app url	ND

Result

You must be able to configure the Java virtual machine used by servers running in a managed domain.

before you start

Use the following command to download the relevant lab files and ensure that the managed domain is set correctly:

```
[student@workstation ~]$ lab jvm-lab-final setup
```

You can use the EAP 7 management console or the JBoss EAP CLI to achieve your goals, keeping in mind that the EAP CLI is the preferred option in production environments.

An EAP administrator has configured a managed domain with two host controllers running the servera and serverb virtual machines, respectively, and the domain controller on the workstation. The domain and host configuration files are stored in the /opt/domain folder on all three machines. You must start the managed domain and configure the Java virtual machine settings on it.

1. Start the domain controller in the workstation virtual machine. Because domain controller configuration files are kept in the /opt/domain folder on workstation, use /opt/domain as the value of the jboss.domain.base.dir argument that you pass to the domain.sh startup script. Also notice that the host file for the domain controller is named host-master.xml and is located in the /opt/domain/ configuration folder. (Tip: Pass the --host-config=host-master.xml argument to domain.sh.)

Note that the /opt/domain directory is owned by the jboss user, so you must start the domain controller using sudo -u jboss /opt/jboss-eap-7.0/ bin/domain.sh ...

```
[student@workstation ~]$ sudo -u jboss /opt/jboss-eap-7.0/bin/domain.sh \ -Djboss.domain.base.dir=/
opt/domain/ --host-config=host-master.xml
```

2. The two host controllers on servera and serverb connect to the host controller domain in the previous step and get the latest configuration of the domain. Start the two host controllers on servera and serverb.

2.1. Start the host controller on servera. Because the host controller configuration files are kept in the /opt/domain folder on servera, use /opt/domain as the value of the jboss.domain.base.dir argument that you pass to the domain.sh startup script. Also notice that the host file

for the host controller it is named host-slave.xml and is located in the /opt/domain/configuration folder. (Hint: Pass the --host config=host-slave.xml argument to domain.sh.)

Note that the /opt/domain directory is owned by the jboss user, so start the host controller using sudo -u jboss /opt/jboss eap-7.0/bin/domain.sh ...

Open a new terminal window on the server virtual machine and run the following command:

```
[student@servera ~]$ sudo -u jboss /opt/jboss-eap-7.0/bin/domain.sh \
-Djboss.domain.base.dir=/opt/domain/ \
-Djboss.domain.master.address=172.25.250.254 \ --host-
config=host-slave.xml
```

2.2. Start the host controller on serverb. Because the host controller configuration files are kept in the /opt/domain folder on serverb, use /opt/domain as the value of the jboss.domain.base.dir argument that you pass to the domain.sh startup script. Also notice that the host file for the host controller is named host-slave.xml and is located in the /opt/domain/configuration folder. (Hint: Pass the --host config=host-slave.xml argument to domain.sh.)

Note that the /opt/domain directory is owned by the jboss user, so start the host controller using sudo -u jboss /opt/jboss eap-7.0/bin/domain.sh ...

Open a new terminal window on the serverb virtual machine and run the following command:

```
[student@serverb ~]$ sudo -u jboss /opt/jboss-eap-7.0/bin/domain.sh \
-Djboss.domain.base.dir=/opt/domain/ \
-Djboss.domain.master.address=172.25.250.254 \ --host-
config=host-slave.xml
```

23. Verify that both host controllers connect to the domain controller and form a managed domain. Look at the console window in which you started the domain controller and verify that both servera and serverb are registered as slaves to the domain controller.

3. You have been instructed by the development team to configure the JVM for the domain managed with the following specifications:

- **The minimum and maximum heap size of the default JVM in the servera host controller should be set to 512 MB.**
- **For all servers in the Group1 server group, create a new JVM configuration named group1-jvm. Set the minimum JVM heap size to 512m (512MB) and the maximum to 1024m (1GB). Enable the -server flag for this server group.**

- For server serverb.2, create a new JVM configuration named serverb.2-jvm, which overrides the legacy configuration. Set the minimum and maximum JVM heap size to 1024m (1 GB). Enable the -XX: +AggressiveOpts and -server flags for this server.
- Leave all other JVM settings at their default values.

Make sure that all servers in the managed domain start with these settings and without errors. Check the JVM flags with which the servers are running, using the ps -aef command on the respective hosts.

use

When you start the managed domain at the beginning of the lab, all servers are configured to start automatically. Be sure to STOP all servers before making changes to the JVM configuration in the managed domain.

3.1. Start the EAP CLI or the management console to configure the JVM.

In a new terminal window on workstation, start the EAP CLI and connect to the domain controller as the jboss user:

```
[student@workstation ~]$ sudo -u jboss\ /opt/jboss-
  eap-7.0/bin/jboss-cli.sh \ --connect --
controller=172.25.250.254:9990
```

3.2. Stop the servers in the Group1 server group.

```
[domain@workstation:9990 /] /server-group=Group1:\ stop-
servers(blocking=true)
```

The expected result is:

```
{
        "outcome" => "success",
        ...
}
```

3.3. Stop the servers in the Group2 server group.

```
[domain@workstation:9990 /] /server-group=Group2:\ stop-
servers(blocking=true)
```

The expected result is:

```
{
        "outcome" => "success",
        ...
}
```

3.4. Set the default JVM so that the servera host controller uses the minimum amount of dynamic memory at 512 MB.

```
[domain@workstation:9990 /] /host=servera\ /
jvm=default:write-attribute(name=heap-size,value=512m)
```

The expected result is:

```
{
        "outcome" => "success",
        ...
}
```

Set the maximum amount of dynamic memory to 512 MB

```
[domain@workstation:9990 /] /host=servera\ /
jvm=default:write-attribute(name=max-heap-size,value=512m)
```

The expected result is:

```
{
        "outcome" => "success",
        ...
}
```

3.5. Configure the minimum amount of JVM memory for the pool Group1 at 512 MB and the maximum amount of memory at 1024 MB. Also send the -server parameter to the JVM .

```
[domain@workstation:9990 /] /server-group=Group1\ /jvm=group1-
jvm:add\ (heap-
 size=512m,max-heap-size=1024m,jvm-options=["-server"])
```

The expected result is:

```
{
        "outcome" => "success",
        ...
}
```

3.6. Configure the JVM for the server serverb.2. The minimum amount of memory for the server must be 1024 MB and the maximum amount of memory must be 1024 MB. Also send -XX:+AggressiveOpts and -server as parameters to the JVM .

```
[[domain@172.25.250.254:9990 /] /host=serverb\ /server-
config=serverb.2\ /jvm=serverb.2-
jvm:add\ (heap-size=1024m,max-
 heap-size=1024m,\ jvm-options=["-XX:
+AggressiveOpts","-server"]) {

        "outcome" => "success",
        ...
}
```

3.7. Start the servers in Group1.

```
[domain@workstation:9990 /] /server-group=Group1\ :start-
servers(blocking=true)
```

The expected result is:

```
{
    "outcome" => "success",
    ...
}
```

3.8. Start the servers in Group2.

```
[domain@workstation:9990 /] /server-group=Group2\ :start-
servers(blocking=true)
```

The expected result is:

```
{
    "outcome" => "success",
    ...
}
```

3.9. Verify the startup of the servers with the JVM configured, using the ps command in each host.

```
[domain@workstation:9990 /] ps -aef | grep <name of server>
```

4. Shut down servers, server groups, host controllers, and the entire domain managed.

4.1. Stop all servers in Group1.

```
[domain@172.25.250.254:9990 /] /server-group=Group1:stop-servers(blocking=true)
```

The expected result is:

```
{
    "outcome" => "success",
    ...
}
```

4.2. Stop the servers in Group2.

```
[domain@172.25.250.254:9990 /] /server-group=Group2:stop-servers(blocking=true)
```

The expected result is:

```
{
```

```
        "outcome" => "success",
        ...
}
```

4.3. Close the host controller on servera. Observe the servera console window and verify that the host controller has been shut down.

```
[domain@172.25.250.254:9990 /] /host=servera:shutdown()
```

The expected result is:

```
{
        "outcome" => "success",
        ...
}
```

4.4. Close the host controller in serverb. Observe the serverb console window and verify that the host controller has been shut down.

```
[domain@172.25.250.254:9990 /] /host=serverb:shutdown() {

        "outcome" => "success",
        ...
}
```

5. Perform cleaning and grading.

5.1. Press Ctrl+C to stop the domain controller. (Alternatively, you can shut down the domain controller using the JBoss EAP CLI command / host=master:shutdown()).

5.2. Press Ctrl+C to exit the EAP CLI if you used the CLI in your lab work. (Alternatively, you can leave the CLI by typing exit.)

5.3. Run the following command on workstation to grade the assignment:

```
[student@workstation bin]$ lab jvm-lab-final grade
```

This concludes the lab work.

Summary

In this chapter, you learned the following:

- Both the EAP server and the applications deployed on EAP run on a Java Virtual Machine (JVM). The JVM can be configured for optimal performance by setting a series of command line parameters used to start the JVM process.

- The JVM settings for a standalone server and a managed domain are configured in different ways. The JVM settings for a standalone EAP server are configured in the $JBOSS_HOME/bin/standalone.conf file.

- JVM settings for an EAP managed domain can be configured at three levels many different:

  ÿ Host controller level

  ÿ Server group level

  ÿ Server level

- JVM settings at the lower levels override settings inherited from the lower levels. higher levels: servers can override values inherited from server groups and host controllers, and server groups can override values inherited from host controllers.