Guided Exercise: Create a Standalone Server

In this lab work, you will create a standalone EAP server.

Resources	
Files:	**/home/student/JB248/labs/standalone-instance**
App URL:	**http://localhost:18080 http://localhost:19990**

Results

You should be able to run a standalone EAP server using a custom location for the server's base directory.

Before you begin

Before you begin the guided exercise, run the following command to verify that EAP was installed to /opt/jboss-eap-7.0 and that no EAP instances are running, and to download the files for the exercise:

```
[student@workstation ~]$ lab standalone-instance setup
```

1. Copy the standalone directories to a new location.

In this guided exercise, you will create an alternate directory to customize the EAP server and leave the original installation intact. Copy the configuration, deployments, and lib directories to the new location.

```
[student@workstation ~]$ cd /opt/jboss-eap-7.0/standalone [student@workstation
standalone]$ cp -r configuration deployments lib \
~/JB248/labs/standalone-instance
```

2. Modify the port offset for the standalone server.

To verify that the EAP server is using the newly copied configuration files, you will need to update the port offset to match the ports that this EAP instance creates, allowing multiple EAP instances to run at the same time, without port conflicts. .

2.1. Update the new configuration file by editing the standalone.xml file, to verify that EAP is using the configuration files in the new folder. Open the /home/ student/JB248/labs/standalone-instance/ configuration/standalone.xml file with a text editor.

2.2. In the standard-sockets socket binding group, change the value

The port-offset attribute defaults from 0 to 10000. This action will add 10000 to each port number used in the standard-sockets bonding group.

```
...
<socket-binding-group name="standard-sockets" default-interface="public" port  offset="${jboss.socket.binding.port-
offset:10000}">
...
```

23. Save the changes to standalone.xml and close the text editor.

3. Run and test the EAP server.

3.1. Run the following command to start the EAP server using the script standalone.sh in the original EAP installation, while using the new EAP configuration files:

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

The server should start without issue with output similar to the following:

```
17:03:30,497 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: JBoss EAP 7.0.0.GA (WildFly Core
  2.1.2.Final-redhat-1) started in 3313ms -Started 261 of 509 services (332 services are lazy, passive or on-
  demand)
```

use

When starting the server for the first time, students may receive an error similar to the following:

```
java.lang.IllegalArgumentException: Failed to instantiate class
   "org.jboss.logmanager.handlers.PeriodicRotatingFileHandler" for handler "FILE"

 ...
Caused by: java.io.FileNotFoundException: /opt/jboss-eap-7.0/ standalone/log/
server.log (Permission Denied)
```

This error can be safely ignored because it is an EAP error. Subsequent server starts should not generate the same error message.
error.

3.2. On the workstation, point the browser to http://localhost:18080 to see the EAP welcome page with the new port offset.

3.3. On the workstation, point the browser to http://localhost:19990 to see the EAP management console running with the same port offset.

use

The administrator username is jbossadm and the password is JBoss@RedHat123.

3.4. Browse the contents of the /home/student/JB248/labs/standalone folder instance in a new terminal window:

```
[student@workstation ~]$ ls /home/student/JB248/labs/standalone-instance/ configuration data
deployments lib log tmp
```

Notice the three new folders: data, log, tmp. The folders are automatically created when the EAP server starts.

4. Perform cleaning.

4.1. Stop the running EAP instance that was started in the previous step. Press Ctrl+C in the terminal window in which the server is running.

This concludes the guided exercise.