Guided Exercise: Explore the CLI Tool

In this lab assignment, you will explore the CLI using offline mode.

Resources	
Files:	**/opt/jboss-eap-7.0/standalone/configuration/ exploring-cli.xml**
App URL:	**N/D**

Results

You should be able to configure a standalone EAP server using the CLI.

before you start

Before beginning 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, as well as to copy the file for the exercise:

```
[student@workstation ~]$ lab exploring-cli setup
```

1. Start the CLI.

1.1. Open a terminal window from the workstation virtual machine (Applications > Favorites > Terminal) and run the following commands to start the CLI:

```
[student@workstation ~]$ sudo
-u jboss /opt/jboss-eap-7.0/bin/jboss-cli.sh
```

1.2. During the installation of this lab, a file named exploring cli.xml was copied to the /opt/jboss-eap-7.0/standalone/configuration folder. Open this file and compare it with the original standalone.xml file to see that they have the same content.

1.3. During this lab work, you must use offline mode to explore the CLI tool. Start a new embedded server using exploring cli.xml as the configuration file:

```
[disconnected /] embed-server --server-config=exploring-cli.xml
```

The above command started an embedded instance of the CLI.

2. Learn about configuration items.

2.1. Check the values at the top level using the following command:

```
[standalone@embedded /] /:read-resource
```

Remember that CLI input is ordered in a hierarchical structure starting at /. The above commands execute the read-resource operation at the top level. The / level refers to the XML structure of exploring-cli.xml.

When you run :read-resource, it will request information about the first level of the exploring-cli.xml file. You are getting information from the server.

2.2. The ejb3 subsystem is responsible for configuring the Enterprise Java Beans specification on the EAP server. This subsystem has the resource called thread set. A thread pool contains a number of EJBs already available for instances in the application, when required. Using this approach, the latency will decrease during the requirements.

Using the absolute path, check the default threadset configuration in the ejb3 subsystem:

```
[standalone@embedded /] /subsystem=ejb3/thread-pool=default:read-resource
```

The above command should show that the default thread set can create up to ten threads.

23. Navigate to the logging subsystem and check the configuration recursively:

```
[standalone@embedded /] cd /subsystem=logging
[standalone@embedded subsystem=logging] :read-resource(recursive=true)
```

Unlike the previous step, this step uses the cd command to navigate to the subsystem tree. This helps minimize the commands required to get and change information.

Notice that the CONSOLE handler has the INFO level defined. Startup handlers and logging levels will be discussed later in this chapter.

use

Remember to use the Tab key to auto-complete the command to avoid typing errors.

2.4. The data source is a resource on the server that creates connections, which is responsible for accessing the database of a Java application. The data source specifies the number of connections to create to minimize the amount of time spent waiting for the connection to become available.

Navigate to the data source named ExampleDS in the datasources subsystem and view the description of the test-connection-in-pool operation:

```
[standalone@embedded subsystem=logging] cd \ /
subsystem=datasources/data-source=ExampleDS
[standalone@embedded data-source=ExampleDS] :read-operation-description\ (name=test-
connection-in-pool)
```

This operation is very useful for testing if the data source is working. Based on the description, you can pass a username and password to test the connection.

2.5. See the description of all the attributes of the ExampleDS data source:

```
[standalone@embedded data-source=ExampleDS] :read-resource-description
```

Notice that the command displays whether an attribute is editable or read-only. Check what the min-pool-size attribute is responsible for.

read-resource-description is an alternative to the product documentation.

3. Modify the resources.

3.1. Modify the data source to be at least five connections and up to ten connections within the set:

```
[standalone@embedded data-source=ExampleDS] :write-attribute\ (name=min-pool-size,value=5)
[standalone@embedded data-source=ExampleDS] :write-attribute\ (name=max-pool-size,value=10)
```

3.2. Verify that the values have been changed:

```
[standalone@embedded data-source=ExampleDS] :read-attribute(name=min-pool-size)
[standalone@embedded data-source=ExampleDS] :read-attribute(name=max-pool-size)
```

3.3. Open /opt/jboss-eap-7.0/standalone/configuration/exploring-cli.xml and verify that ExampleDS has an array defined with the values specified in the previous step. You should see the following content:

```
<pool>
    <min-pool-size>5</min-pool-size> <max-
    pool-size>10</max-pool-size> </pool>
```

3.4. Create a new system property named env whose value is production:

```
[standalone@embedded data-source=ExampleDS] /system-property=\
env:add(value=production)
```

Remember that when you want to create a new resource, you need to specify a name after the = and call the :add operation, sending the required attributes. You can discover the available attributes by using the Tab key for auto-complete.

3.5. Try removing the path named user.home. It is available in the structure / path:

```
[standalone@embedded data-source=ExampleDS] /path=user.home:remove
```

The user.home path could not be removed because it is read-only.

3.6. If necessary, it is possible to remove a subsystem. For example, if your company does not have projects that use JSF, you can remove this subsystem. Remove the jsf subsystem:

```
[standalone@embedded data-source=ExampleDS] /subsystem=jsf:remove
```

4. Perform cleaning.

4.1. Exit the CLI tool.

```
[standalone@embedded data-source=ExampleDS] exit
```

4.2. Remove the exploring-cli.xml file.

```
[student@workstation ~]$ sudo \ rm /opt/
jboss-eap-7.0/standalone/configuration/exploring-cli.xml
```

This concludes the guided exercise.