



CHAPTER 4

JBOSS CONFIGURATION EAP AS A DOMAIN MANAGED

General description	
Meta	Configuring JBoss EAP as a managed domain •
Goals	<p>Describe the managed domain architecture.</p> <ul style="list-style-type: none"> • Map the domain controller and start the managed domain. • Describe the configuration options for a host controller and make configuration changes to a host controller. • Describe configuration options for a domain controller and make configuration changes to the domain controller.
sections	<ul style="list-style-type: none"> • Running JBoss EAP as a managed domain (and quiz) • Assignment of a domain controller (and exercise guided) • Configuring a host controller (and exercise guided) • Configuration of a domain controller (and questionnaire)
Laboratory work	• Configuring JBoss EAP as a managed domain

Running JBoss EAP as a managed domain

Goals

After completing this section, students should be able to do the following:

- Describe the managed domain architecture.

Managed domain architecture

To understand EAP managed domains, you need to understand the following terms and how they are related to each other:

Domain

A group of EAP server instances.

domain controller

A single process that functions as the central management control point for a domain. The domain controller is also known as the host controller. master.

Server

An EAP server instance that runs in its own JVM process. Run the application code; that is, it acts as an application server.

host controller

A process running on a host machine that relays configuration information, runtime status, and administration commands to the EAP server instances on that particular machine. The host controller is also known as the slave.

process controller

A process running on a host machine that starts host controllers and server instances on that particular machine.

Host

The set of processes started by the same process controller; a host controller and zero or more application server instances.

Server group A group

of servers that are managed and configured as one.

Profile A named set of EAP subsystem configurations.

The following figure helps to visualize the relationships between the terms described above:

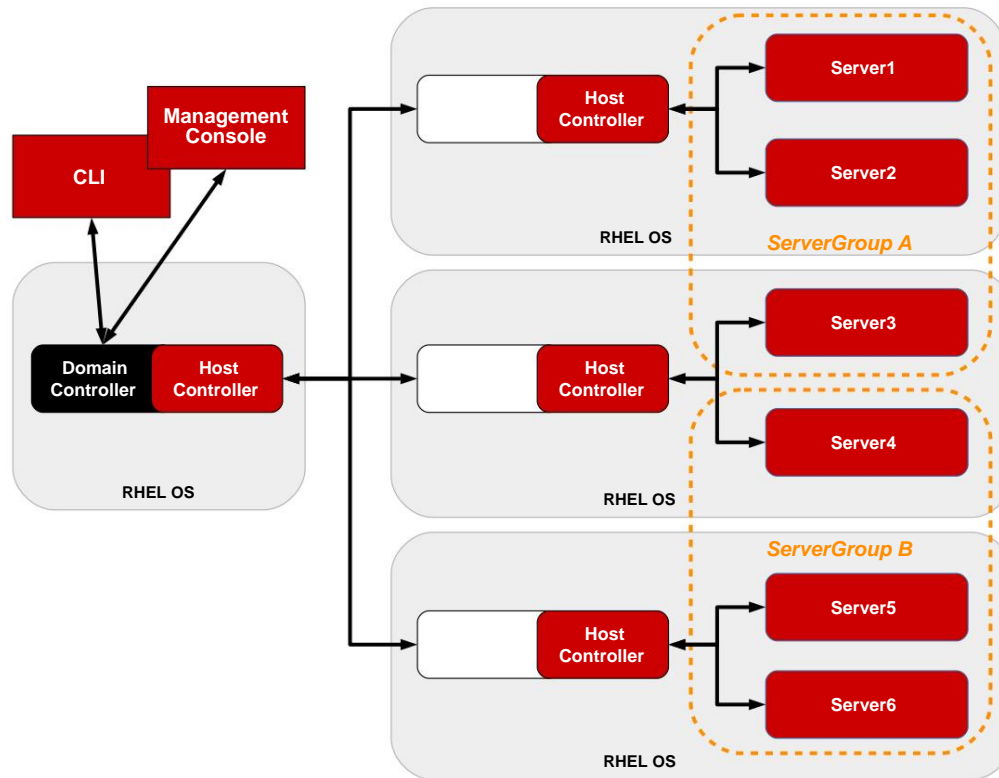


Figure
4.1: EAP Managed Domain

In the figure above, the light gray boxes represent teams. They can be physical or virtual machines, and they can run OSes other than Red Hat Enterprise Linux (RHEL). Each machine corresponds to a host and runs a process controller (not shown in the figure) and a host controller. Each machine also runs two server instances, but there can be more or less.

In a managed domain, one of the host controller instances is configured to function as the central management point, that is, to function as the domain controller. In the figure above, the host managed by the domain controller does not have any server instances. This approach is recommended but NOT required, since a domain controller is also a host controller and can directly manage its own server instances.

Each host controller interacts with the domain controller to ensure that each server instance is configured according to the domain's policies. All management interfaces, such as the graphical management console and the CLI, communicate only with the domain controller. The management console runs as part of the JVM and domain controller OS process.

Any host controller can be configured as a substitute domain controller in case the original is not available, but this promotion is not automated.

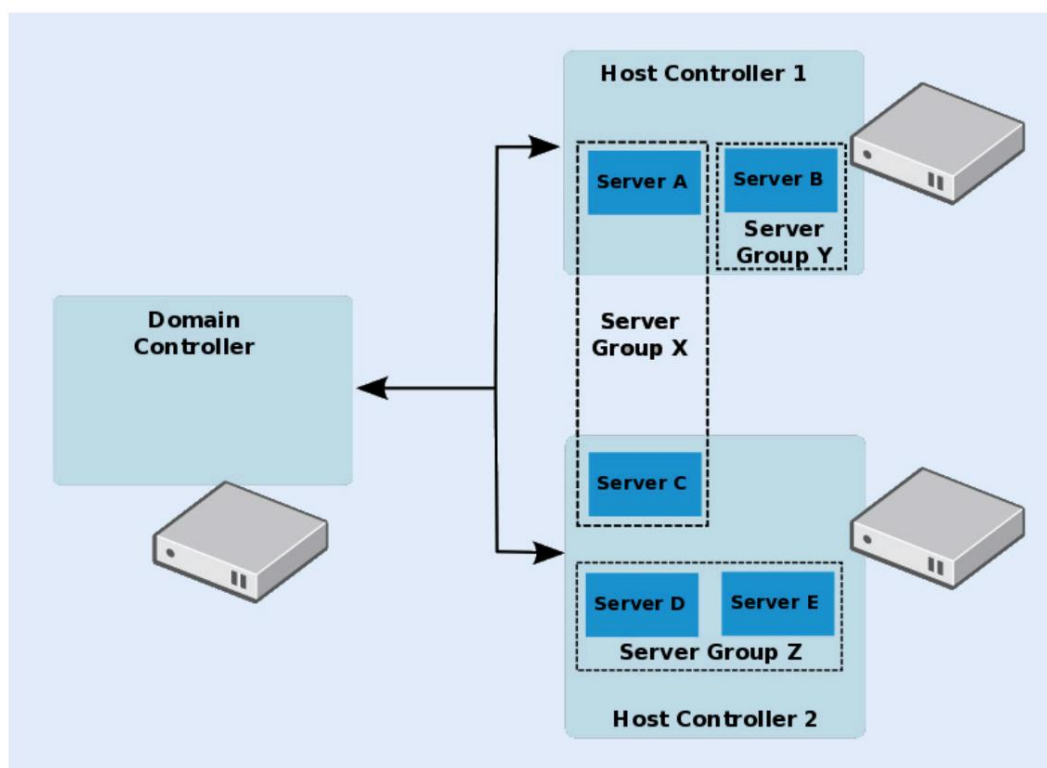
A host controller does not perform application server tasks, such as serving Java EE applications and responding to client requests. This is done exclusively by the server instances. The only responsibilities of the host controller are to interact with the domain controller to help manage the host servers.

Chapter 4. Configuring JBoss EAP as a managed domain

Hosts provide a physical pool that affects performance metrics such as processor cores and available RAM (dynamic memory). Server groups provide a logical grouping that affects configuration settings; EAP profiles are related to groups of servers, not individual server instances. Applications are also deployed and activated only for groups of servers and never for individual server instances in a managed domain.

The EAP managed domain startup script `domain.sh` starts a host process controller, and the process controller starts the host controller. The host controller then requests that the process controller start the server instances according to the host configuration, and subsequently when requested by the domain controller.

Quiz: Managed Domains



Based on the graph above, choose the correct answer to the following questions:

1. How many host controllers are in the domain? (Choose one option).

- a. 5
- b. 3
- c. 2
- d. 1

2. How many servers are in the domain? (Choose one option).

- a. 5
- b. 3
- c. 2
- d. 1

3. Which servers are managed by host controller 1? (Choose two options).

- a. Server A
- b. Server B
- c. Server C
- d. Server D

It is. **Server And**

4. Which servers are managed by host controller 2? (Choose three options).

- a. **Server A**
 - b. **Server B**
 - c. **Server C**
 - d. **Server D**
- It is. **Server And**

5. What servers are managed by the domain controller? (Choose one option).

- a. **Server A**
 - b. **Server B**
 - c. **Server C**
 - d. **Server D**
- It is. **Server And**
- f. **None of them**

6. Which servers belong to server group Z? (Choose two options.)

- a. **Server A**
 - b. **Server B**
 - c. **Server C**
 - d. **Server D**
- It is. **Server And**

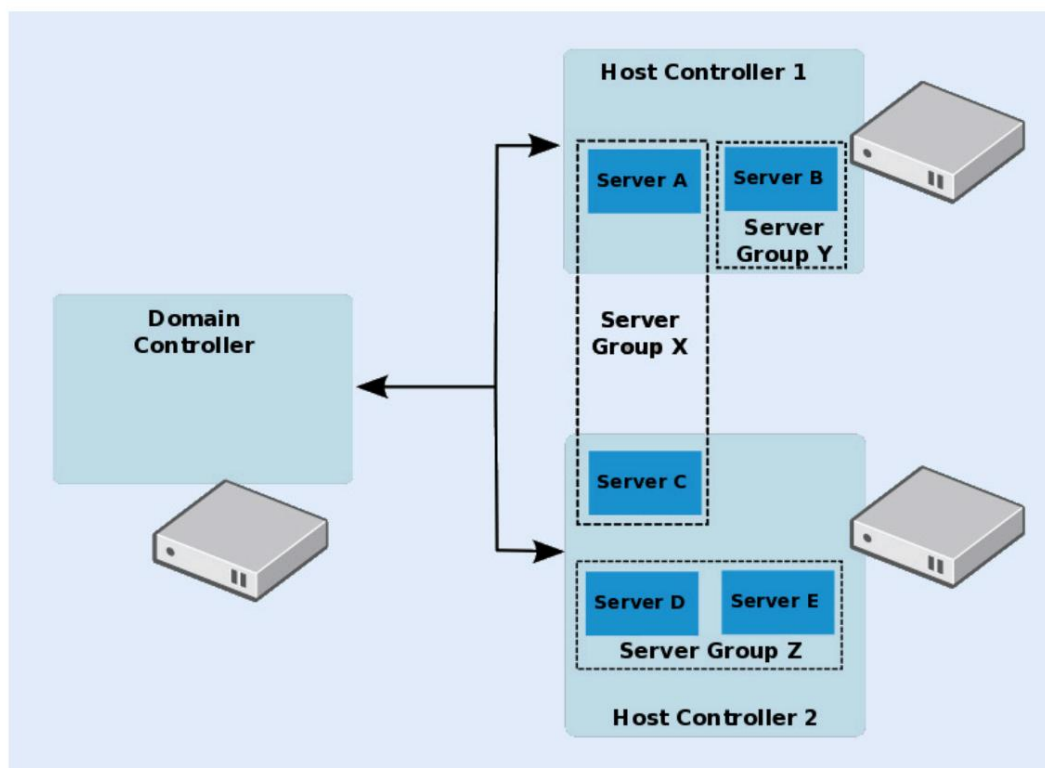
7. Which servers belong to the X server group? (Choose two options.)

- a. **Server A**
 - b. **Server B**
 - c. **Server C**
 - d. **Server D**
- It is. **Server And**

8. Is server group Y necessary or can server B function outside of a group because there are no other members in the group? (Choose one option).

- a. **Yes, because the server group determines the host where the server instance will start.**
- b. **No, standalone server instances CANNOT be members of a managed domain server group.**
- c. **Yes, a server instance can NOT be part of a managed domain without being a member of a server group.**
- d. **No, because the only function of server groups is to ensure that server instances are related to the same profile, and if Server B is the only one related to the same profile, a server group is not required.**

Solution



Based on the graph above, choose the correct answer to the following questions:

1. How many host controllers are in the domain? (Choose one option).

- a. 5
- b. 3
- c. 2
- d. 1

2. How many servers are in the domain? (Choose one option).

- a. 5
- b. 3
- c. 2
- d. 1

3. Which servers are managed by host controller 1? (Choose two options).

- a. **Server A**
- b. **Server B**
- c. Server C
- d. Server D
- It is. Server And

4. Which servers are managed by host controller 2? (Choose three options).

- a. ☐ Server A
- b. ☐ Server B
- c. ☒ Server C
- d. ☒ Server D
- It is. ☒ Server And

5. What servers are managed by the domain controller? (Choose one option).

- a. ☐ Server A
- b. ☐ Server B
- c. ☐ Server C
- d. ☐ Server D
- It is. ☐ Server And
- f. ☒ None of them

6. Which servers belong to server group Z? (Choose two options.)

- a. ☐ Server A
- b. ☐ Server B
- c. ☐ Server C
- d. ☒ Server D
- It is. ☒ Server And

7. Which servers belong to the X server group? (Choose two options.)

- a. ☒ Server A
- b. ☐ Server B
- c. ☒ Server C
- d. ☐ Server D
- It is. ☐ Server And

8. Is server group Y necessary or can server B function outside of a group because there are no other members in the group? (Choose one option).

- a. ☐ Yes, because the server group determines the host where the server instance will start.
- b. ☐ No, standalone server instances CANNOT be members of a managed domain server group.
- c. ☒ Yes, a server instance can NOT be part of a managed domain without being a member of a server group.
- d. ☐ No, because the only function of server groups is to ensure that server instances are related to the same profile, and if Server B is the only one related to the same profile, a server group is not required.

Assigning a domain controller

Goals

After completing this section, students should be able to do the following:

- Map the domain controller and start the managed domain.
- Configure slave host controllers in an existing managed domain.

Configure a host controller as the master

Settings and settings for a managed domain are divided into two files:

- **host.xml** – The configuration file for a host controller. This file must be configured for domain controller discovery mode, or it must configure itself as a domain controller.

There are other settings in this file, such as server settings, but they will be introduced later in this book. In summary, the values in this file relate to the host hardware and OS specifications.

- **domain.xml** – The domain controller configuration file, which defines available profiles and other settings that are not directly influenced by the managed domain host hardware and OS, such as server groups and socket bindings.

This chapter and the next one focus primarily on the **host.xml** configuration file. The **domain.xml** configuration file is covered in later chapters.

A host controller instance is named based on the hostname of its computer, but this can be overridden by using the **name** attribute on the top **<host>** element at the beginning of the **host.xml** configuration file. For example:

```
<?xml version="1.0" ?> <host
xmlns="urn:jboss:domain:4.1" name="mydomainmaster">
...
```

To denote that a host controller is a domain controller, that is, the master host controller, add the following to your **host.xml** configuration file between the **<management>** and **<interfaces>** elements:

```
<?xml version="1.0" ?> <host
xmlns="urn:jboss:domain:4.1" name="mydomainmaster">
...
</management>

<domain-controller> 1
  <local/> </ 2
domain-controller>

<interfaces>
...
</host>
```

Chapter 4. Configuring JBoss EAP as a managed domain

- 1 The `<domain-controller>` element tells a host controller where to find the domain controller.
- 2 If a host controller must be master for your managed domain, use the `<local/>` element. In other words, host controller configuration implies that "the domain controller is the local host controller, that is, the one that reads the configuration file."

In addition to declaring itself as the master, a domain controller's `host.xml` configuration file must declare a native management interface using the `<native-interface>` element. For example:

```
...
</extensions>

<management>
  ...
  <management-interfaces>
    <native-interface security-realm="ManagementRealm"> <socket
      interface="management" port="$
        {jboss.management.native.port:9999}"/> </native-interface>
    <http-interface security-
      realm="ManagementRealm" http-upgrade-enabled="true"> <socket
        interface="management" port="$
          {jboss.management.http.port:9990}"/> </
          http-interface> </management-interfaces> </management>

  <domain-controller>
  ...
```

The native management interface accepts management API requests using the native EAP binary protocol. Unlike previous EAP versions, this management interface is NOT required for administrative access, although host controller communication still requires it.

A domain controller is not required to define the HTTP administration interface, although it is commonly used, as the listing above shows. This enables the web administration console for the domain. If an administrator wishes, they can remove the HTTP admin interface and use only the admin CLI to manage the domain.

The HTTP management and native interfaces refer to the management (network) interface, so it must also be declared in the `host.xml` configuration file. For example:

```
...
</domain-controller>

<interfaces>
  <interface name="management">
    <inet-address value="{jboss.bind.address.management:127.0.0.1}"/>
  </interface>
  <interface name="public"> <inet-
    address value="{jboss.bind.address:127.0.0.1}"/>
  </interface> </
  interfaces>
```

Configuring host controllers as slaves

```
<jvm>
...
```

Notice that `host.xml` also defines the public network interface. This interface is used by regular users to access applications. Domain controllers are NOT required to define this interface, unless they have their own server instances.



use

Don't confuse management-interfaces with (network) interfaces. The first concerns remote administrative access. The second concerns the separation of different types of network traffic to different IP addresses so that they can be subject to different firewall rules. Those configuration items have the same meaning in the managed domain and standalone server operating modes.

The native management interface also references the `ManagementRealm` security domain. Security in managed domains is introduced later in this book.

Configuring host controllers as slaves

To indicate that a host controller is not the domain controller, that is, that it is supposed to act as a slave, specify in `host.xml` where the domain controller can be found:

```
...
<domain-controller>
  <remote security-realm="ManagementRealm"> 1
    <discovery-options> 2
      <static-discovery name="primary" protocol="$ 3
        {jboss.domain.master.protocol:remote}" host="$
        {jboss.domain.master.address}" port="$
        {jboss.domain.master.port:9999}"/>
      </discovery-options> </
    remote>
  </domain-controller>
...
```

- ❶ The `<remote>` element reports that this host controller is a slave, and the child elements specify how to find the master.
- ❷ EAP 7 supports multiple domain controller discovery mechanisms, configured as children of the `<discovery-options>` element. Different combinations of mechanisms allow you to implement failover policies when the domain controller is unavailable and to find the domain controller in highly dynamic cloud environments.
- ❸ The `<static-discovery>` mechanism points to the IP address and TCP port of the native management interface of the domain controller.

Chapter 4. Configuring JBoss EAP as a managed domain

This course uses only the <static-discovery> mechanism. For information about other mechanisms, see the EAP product documentation.

In the above configuration, the `jboss.domain.master.address` system property must be set when starting the host controller. This is demonstrated later in this section.

An alternative is to modify the `host.xml` configuration file to map it to the name and IP address of the computer on which the domain controller is running. For example:

```
<domain-controller>
  <remote security-realm="ManagementRealm">
    <discovery-options> <static-
      discovery name="primary" protocol="$
        {jboss.domain.master.protocol:remote}"
        host="{jboss.domain.master.address:172.25.14.9}" port="$
        {jboss.domain.master.port:9999}"/>
    </discovery-options> </
  </remote>
</domain-controller>
```

- ❶ Add a default value for the `jboss.domain.master.address` system property reference.

The example above illustrates a common practice with EAP 7 configuration files: provide attribute values as defaults for system properties. This allows you to override the value of the attribute by setting a system property on the command line.

The following figure shows an example managed domain consisting of three hosts, plus the critical parts of each host controller `host.xml` configuration file:

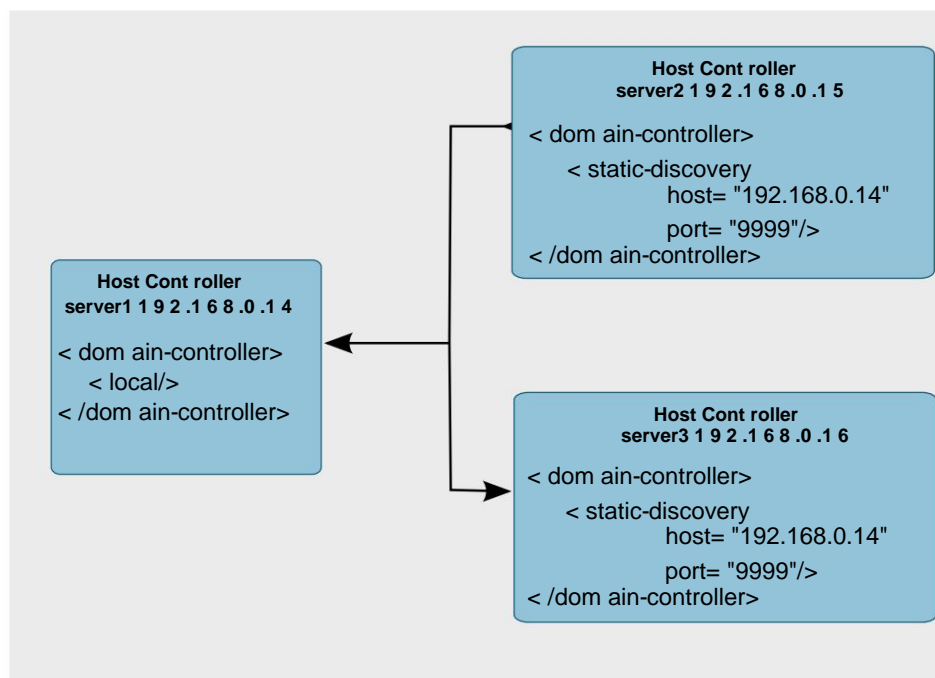


Figure 4.3:

Sample managed domain

In the preceding figure, the server1 computer is running the domain controller. The IP address assigned to all (network) interfaces of the domain controller is 192.168.0.14 and its native management interface uses the default port 9999. The server2 and server3 computers are running slave host controllers configured to point to the domain controller that it runs on the server1 machine.

Starting a managed domain

To start an EAP host controller, use the specific domain script for your platform. The script is located in the /bin folder of the EAP installation. On Linux and UNIX, the command is as follows:

```
$ ./domain.sh
```

When you start EAP in managed domain mode, EAP must determine whether the instance being started is the domain controller or not. The following is the sequence of events that occurs when the domain startup script is executed:

1. The process controller starts in a single JVM process.
2. The process controller starts a host controller in another JVM process.
3. The host configuration file (domain/configuration/host.xml if defaults are used) is processed first, and the <domain controller> element is checked to determine whether this host controller should act as the domain controller or not. .
4. If this particular host controller is configured as a domain controller, the values in the domain configuration file (domain/

Chapter 4. Configuring JBoss EAP as a managed domain

configuration/domain.xml if default values are used) are combined with the values in host.xml. A domain controller must expose a management interface binding that can be reached through an address in order for other host controllers to communicate with it.

5. If this particular host controller is not the domain controller, it should attempt to connect to the domain controller based on the values in the <domain-controller> element in host.xml.
6. Regardless of whether this host controller is the domain controller, it starts its servers, each in its own JVM process, if the host.xml configuration file has any servers defined.

EAP managed domains are secured by default, as they only allow access from the local computer. To allow remote access to an EAP domain controller for administrative actions, as well as for slave host controllers to join the domain, a different IP address must be assigned to the management interface.

Following the example in the previous figure, a domain controller is initialized to listen for slave connections on a private IP address by defining the `jboss.bind.address.management` system property. For example:

```
$ ./domain.sh -Djboss.bind.address.management=192.168.0.14
```

The example above shows a domain controller whose management network interface is assigned the IP address 192.168.0.14.

Continuing with the same example, a slave host controller would be started to connect to the master at that address, by setting the `jboss.domain.master.address` system property, as well as listen for management commands (from the master) on another IP address. private, by defining the `jboss.bind.address.management` system property.

The following example should be used to start the host controller on the server2 computer:

```
$ ./domain.sh -Djboss.bind.address.management=192.168.0.15 \
-Djboss.domain.master.address=192.168.0.14
```

In the example above, a slave host controller boots and attempts to connect to a domain controller running at 192.168.0.14 on port 9999. Notice that the master IP address was specified on the command line using a system property, while the master native management interface TCP port was left as the default provided by the host.xml configuration file.

The example above also shows that the slave accepts management requests from the master at the IP address 192.168.0.15, specified on the command line using a system property. The slave native management interface TCP port was also left as the default provided by the host.xml configuration file.

In closing, the following example should be used to start the host controller on machine server3:

```
$ ./domain.sh -Djboss.bind.address.management=192.168.0.16 \
-Djboss.domain.master.address=192.168.0.14
```

Host configuration file names and folders

Most EAP installations need custom configuration files and want to keep the default configuration files unchanged for future reference. This can be done using some command line options of the startup host script:

- `--domain-config` provides an alternate name for the `domain.xml` file.
- `--host-config` provides an alternate name for the `host.xml` file.

For example:

```
$ ./domain.sh --domain-config=mydomain.xml --host-config=myhost.xml
```

The files specified by `--host-config` and `--domain-config` must exist in the default location which is the `JBOSS_HOME/domain/configuration` folder. This path can be modified by assigning a new value to the `jboss.domain.config.dir` system property.

To create a custom configuration folder for an EAP host controller, you typically need to create custom folders for subsystem data files, logs, and other functional files. Because of this, EAP also recognizes the `jboss.domain.base.dir` system property. Provides an alternate location for the `JBOSS_HOME/domain/` folder, which should have the configuration, data, and log folders as children.

For example, to start an EAP host controller using domain configuration files from `/usr/local/eap/configuration`, use the following command:

```
$ ./domain.sh -Djboss.domain.base.dir=/usr/local/eap/configuration
```

You can combine the use of system properties and filename options in a single command:

```
$ ./domain.sh -Djboss.domain.base.dir=/usr/local/eap/configuration \
--domain-config=mydomain.xml --host-config=myhost.xml
```

It is recommended that you copy the `/domain` folder from a clean installation of EAP 7 before starting EAP on a managed domain. Separates configurations, deployments, log files, and temporary folders of specific host controller instances from the default installation folders. Two key benefits are gained from this practice:

1. Multiple EAP instances in a managed domain, that is, multiple host controllers and their associated server instances, can be run on the same machine using the same setup files.

Chapter 4. Configuring JBoss EAP as a managed domain

2. It is possible to update EAP without affecting or overwriting already customized configurations.

The `jboss.domain.base.dir` runtime property enables this best practice.

The default `host.xml` file configures the host controller as master. It also configures sample server instances and groups so that test applications can be deployed immediately. It can NOT be used unmodified to start additional host (slave) controllers.

To start the additional slave host controllers in the domain, the default `host.xml` configuration file must be modified to replace the `<local/>` element with a `<remote>` element, as noted above.

For convenience, the EAP installation also includes two additional sample host controller configuration files:

- `host-master.xml` – This file configures a minimal domain controller with no server instances. Because it defines only the management network interface, it can NOT be used as is, when the master host controller must also have server instances. Having a master host controller with no server instances is a recommended EAP 7 configuration.
- `host-slave.xml` – This file configures a slave host controller with the same sample server groups and server instances, as the default `host.xml` configuration file. This configuration file uses the `jboss.domain.master.address` and `jboss.domain.master.port` system properties to provide the master IP address and TCP port.

Recover from a failed domain controller

An EAP managed domain is a dynamic entity, created by the existence of one or more host controllers running the same domain configuration. Typically, the master host controller (that is, the domain controller) boots first, and then the slave domain controllers boot. Additional host controls can join the managed domain at any time.

If the domain controller is unavailable for some reason, no new host controllers can join the managed domain, although the existing ones continue to run. Running applications are not affected by domain controller failures; only administration tasks are affected.

At the time this book was written, the domain controller did not have an automatic failover mechanism. If a domain controller fails, the host controllers simply keep trying to reconnect until the domain controller boots up again. Note that the server instances are NOT affected if a domain controller fails, but continue to function and respond to requests.

In this way, a domain controller can fail without affecting user application uptime.

A host controller can maintain a cache of the managed domain configuration file so that the host can start if the master controller is down. Start the slave controller with the `--backup` option so that the slave host controller creates and maintains a local copy of the domain configuration file. For example:

The administration console in a managed domain

```
$ ./domain.sh --backup -Djboss.domain.master.address=192.168.0.14
```

If the slave host controller cannot communicate with the master domain controller to get its configuration at startup, the `--cached-dc` option tells the slave controller to use a local copy created earlier using `--backup`. For example:

```
$ ./domain.sh --cached-dc -Djboss.domain.master.address=192.168.0.14
```

`--backup` and `--cached-dc` can NOT be used at the same time, so their use as a failover mechanism requires manual intervention by the system administrator. Also note that the `--cached-dc` option will NOT cause the slave host controller to act as the domain controller: cached settings are considered read-only.

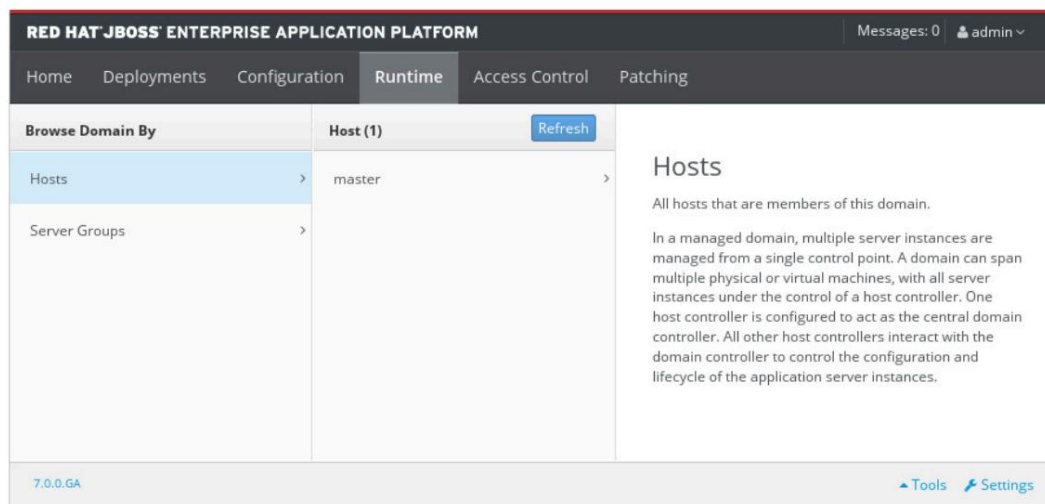
The administration console in a managed domain

The EAP 7 administration console is somewhat different compared to running from a separate server instance. Additional tabs and navigation steps are required to address the extra items provided by the managed domain: named profiles, hosts, server groups, and server instances.

The basic navigation is the same, and the specific configuration of the subsystem is also the same. The most visible changes are:

- **Deployment tab** – Deployments are organized into groups of servers, instead of having all applications running on the single server instance.
- **Configuration tab** – A profile must be selected before configuring a subsystem.
- **Runtime tab** – Adds operation to view and configure hosts, server groups, and server instances.

The following figure shows the Runtime tab as it appears after starting EAP 7 as a managed domain using the default configuration files:



Chapter 4. Configuring JBoss EAP as a managed domain

Figure

4.4: Runtime tab of the EAP administration console in managed domain mode

Similar changes occur in the management CLI as well, as there are objects that represent the additional configuration items of the managed domain.