CHAPTER 5

SERVERS CONFIGURATION ON A MANAGED DOMAIN

General description	
Meta	Configure servers and server groups in a managed domain.
Goals	• Describe the relationship between hosts, server groups, and servers in a managed domain. • Configure groups of servers in a managed domain. • Configure servers on a host controller. • Implement
sections	• Managed domain server architecture (and quiz) • Configuration of groups of servers (and exercise guided) • Server configuration (and guided exercise) • Deploying applications in a managed domain (and guided exercise)
Laboratory work	• Configuring servers in a managed domain

Managed Domain Server Architecture

Goals

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

• Describe the relationship between hosts, server groups, and servers.

Understanding hosts and servers

A managed domain consists of:

• *Domain Controller:* Responsible for all configuration management through *profiles.*

• *Host Controller* – Responsible for managing the server instances. In a production environment, a host controller is typically installed on a separate host (physical or virtual machine).

• *Server Groups* – A logical grouping of EAP server instances that are configured and managed together as a single unit. Server groups can span any number of servers on multiple host controllers.

• *Server:* responsible for executing JavaEE applications (JAR, EAR, WAR files).

To simplify the administration of server instances, EAP defines the concept of a *server pool,* which shares the same set of applications and reuses the same profile along with all built-in configurations.

use

A server cannot be part of multiple server groups.

Server groups are configured in the domain.xml configuration file on the domain controller, while servers are configured in the host.xml configuration file on each host.

The topology of an EAP managed domain can be looked at from two points of view:

• Host View – A managed domain consists of multiple hosts, where a host is an EAP instance operating in managed domain mode. An EAP managed domain is a set of hosts.

• Server View - An EAP managed domain can be viewed as a group of server instances, and each server instance belongs to a server group.

For demonstration purposes, the following graphic shows a domain from the point of view of hosts within the managed domain:

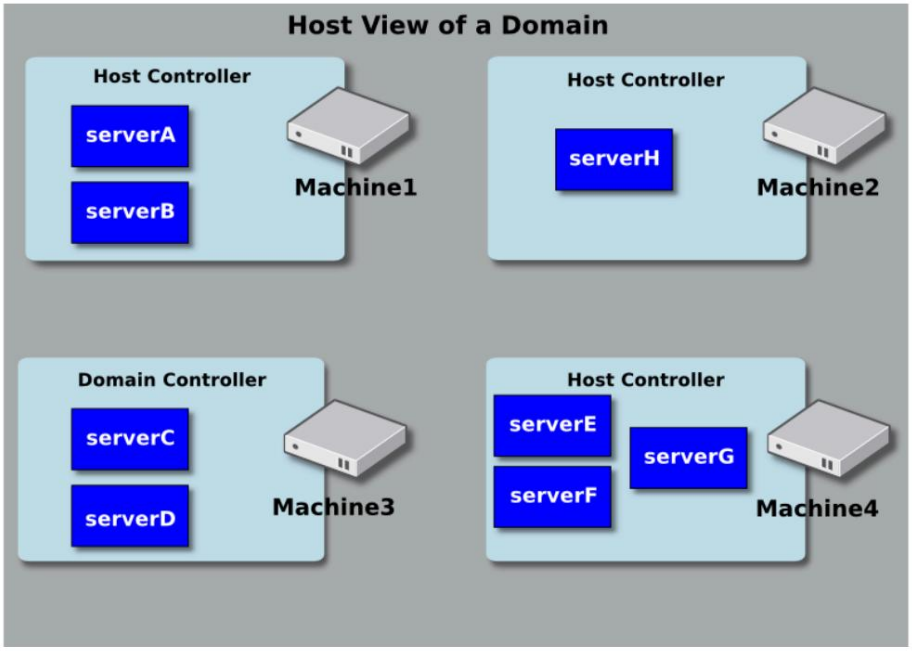


Figure 5.1: View of domain hosts

The graph above shows which servers are running on a host, without addressing which server groups each belongs to. It is difficult to identify which servers share the same set of configurations; however, for an administrator, it is useful to see how many processes are running on each host (physical or virtual machine). An administrator can plan the amount of memory and CPU capacity required for each host based on this information.

The following graphic shows the same EAP managed domain, only from the server point of view:

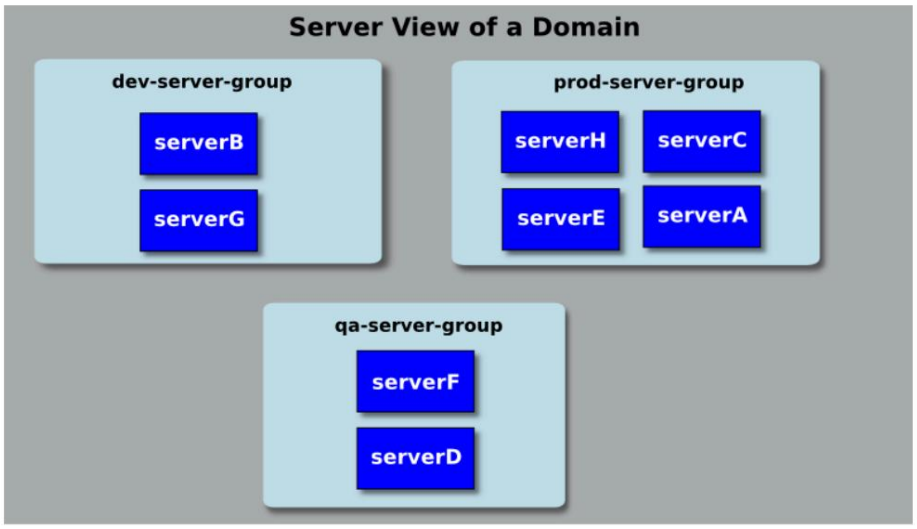


Figure 5.2: View of domain servers

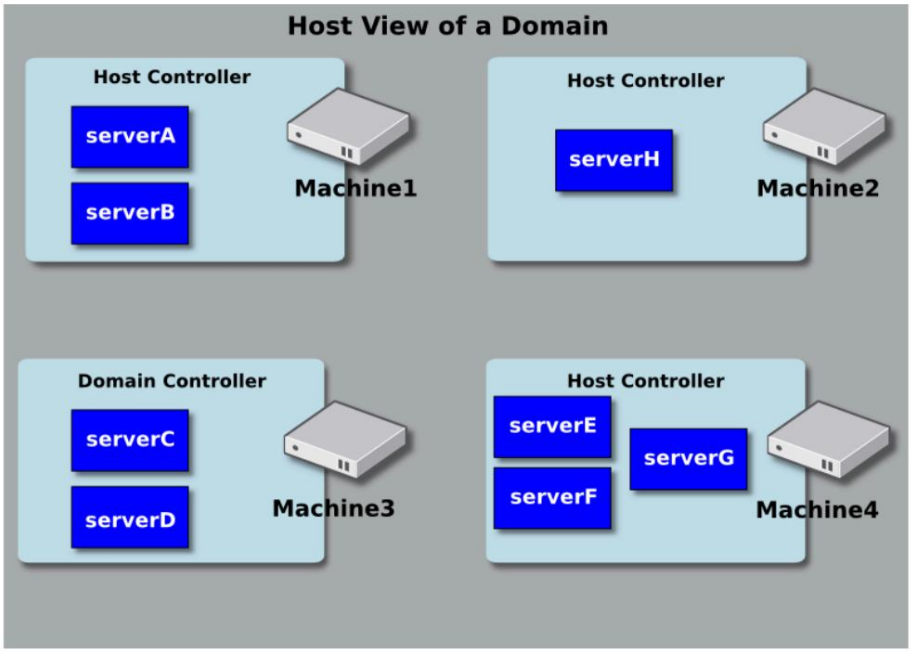
The above graphic shows the servers and the server group they belong to, providing a logical view of an EAP managed domain. It doesn't address hardware issues, but it does show which servers share the same set of configurations.
Notice that in the servers view, the important difference is the server group the server belongs to, not the computer the server is running on. When designing a managed domain, the server view is critical because applications are not deployed to servers, but to groups of servers.

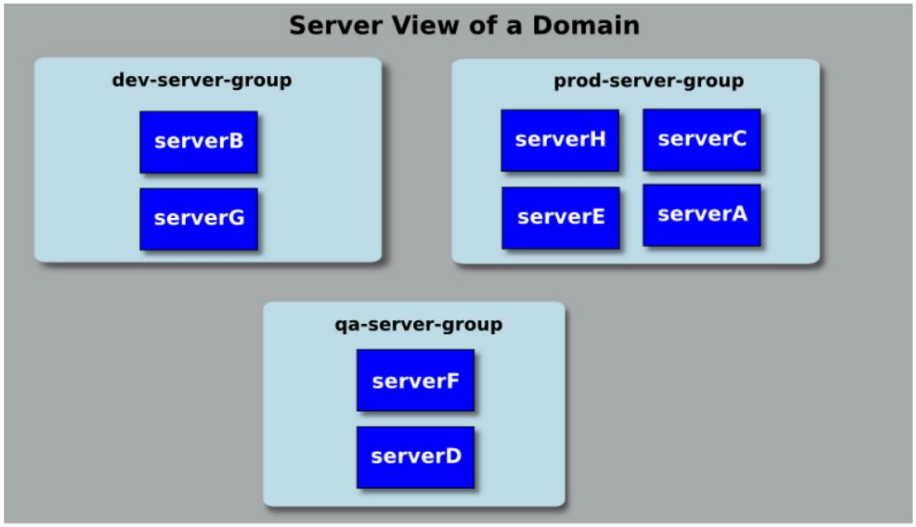
These views complement each other and help the administrator plan how many EAP host controllers to deploy and how many servers to create.

use

Different groups of servers can be configured with different *profiles* and *implementations (applications).* Different groups of servers can run the same profile and have the same deployments. One benefit of having identical server groups is the ability to support rolling application upgrade scenarios, where complete service disruption is avoided by first updating the application on one server group, and then , by updating the application on the second pool.

Quiz: Hosts and servers





Look at the host view and server view graphs above, and choose the correct answer to the following questions:

1. How many server groups are defined in the managed domain? (Choose one option).

a. 1
b. 2
c. 3

d. 4

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

a. 3
b. 8
c. 6
d. 5

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

a. 2
b. 4
c. 6
d. 8

4. If an application example.war can be accessed from serverA, which of the following statements are true? (Choose two options.)

a. example.war can be accessed from serverB. b. example.war is implemented in qa-server-group.
c. example.war can be accessed from serverH.
of example.war can be accessed from serverG. example.war is deployed to dev-server-group. F. example.war is implemented in prod-server-group.

5. Which of the following statements about prod-server-group are correct? (Choose two options.)

a. An application can be implemented in such a way that it only runs on serverH and not on other servers in the prod-server-group. serverH is part of the prod-server-group.
b.
c. All servers in the prod-server-group must be running on a single host.
d. Multiple applications can be deployed in prod-server-group.

6. Which of the following statements about the Machine4 host are correct? (Choose two options).

a. serverE, serverF and serverG must belong to the same server group.
b. There can only be one server group created on host Machine4. serverE, serverF, and serverG
c. can be assigned to any server group.
d. Multiple applications can run simultaneously on the Machine4 host.

It is. The configuration for the Machine4 host is stored in a file named domain.xml.

7. Which of the following statements about the managed domain are correct? (Choose three options).

to. prod-server-group and dev-server-group must be set to the same profile.

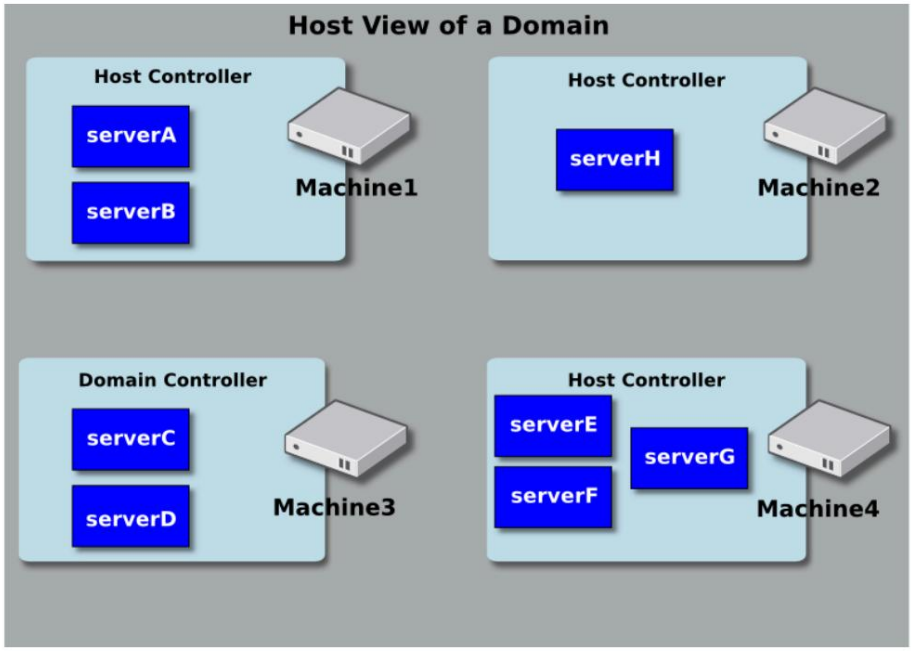
b. prod-server-group and dev-server-group can be configured with different profiles.

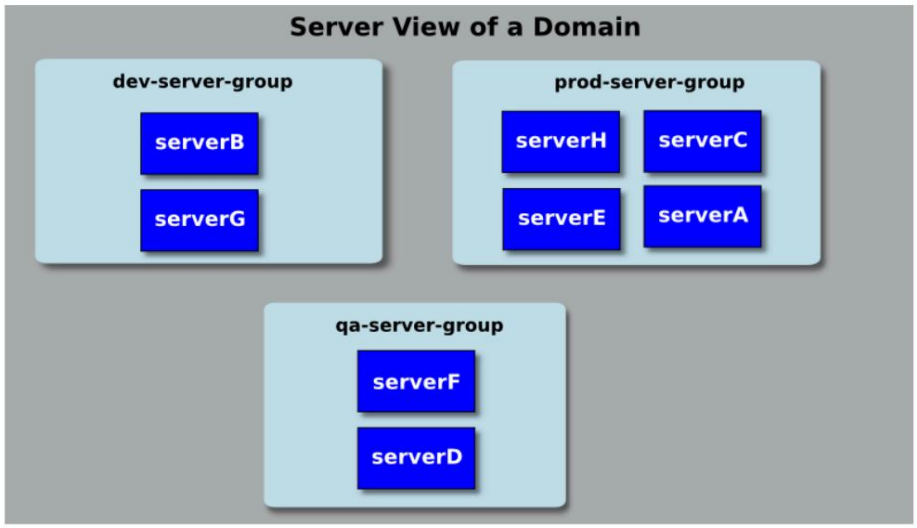
c. All server groups must be configured with the same profile and the same deployments (applications).

d. Multiple different applications can be deployed in prod-server-group and dev-server-group. and. A server group may have no applications. F.

A host must always have at least one server configured and running.

Solution





Look at the host view and server view graphs above, and choose the correct answer to the following questions:

1. How many server groups are defined in the managed domain? (Choose one option).

a.
b. 1 2
c. **3**
d. 4

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

a. 3
b. **8**
c. 6
d. 5

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

a. 2
b. **4**
c. 6
d. 8

4. If an application example.war can be accessed from serverA, which of the following statements are true? (Choose two options.)

a. example.war can be accessed from serverB. **b.** example.war is implemented in qa-server-group.
c. **example.war can be accessed from serverH.**
example.war can be accessed from serverG.
of example.war is deployed to dev-server-group. **F.**
example.war is implemented in prod-server-group.

5. Which of the following statements about prod-server-group are correct? (Choose two options.)

a. An application can be implemented in such a way that it only runs on serverH and not on other servers in the prod-server-group. **serverH is part of the prod-**
b. **server-group.**
c. All servers in the prod-server-group must be running on a single host.
d. **Multiple applications can be deployed in prod-server-group.**

6. Which of the following statements about the Machine4 host are correct? (Choose two options).

a. serverE, serverF and serverG must belong to the same server group.
b. There can only be one server group created on host Machine4. **serverE, serverF,**
c. **and serverG can be assigned to any server group.**

d. **Multiple applications can run simultaneously on the Machine4 host.**

It is. The configuration for the Machine4 host is stored in a file named domain.xml.

7. Which of the following statements about the managed domain are correct? (Choose three options).

to. prod-server-group and dev-server-group must be set to the same profile.

b. prod-server-group and dev-server-group can be configured with different profiles.

c. All server groups must be configured with the same profile and the same deployments (applications).

d. Multiple different applications can be deployed in prod-server-group and dev-server-group. and. A server group may have no applications. F. A host must always have at least one server configured and running.

Configuring server groups

Goals

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

- **Configure a group of servers in a managed domain.**
- **Describe the different options available for creating and managing groups of servers.**
- **Describe the different attributes of a group of servers that are configured in the domain.xml file of the domain controller.**
- **Describe the different options available for deploying applications in an EAP managed domain.**

server groups

A server group is a logical grouping of servers in an EAP managed domain, which are managed as a single unit. Server groups are defined and managed by the domain controller, but each individual server is managed by respective host controllers that communicate with the domain controller and ensure that all servers within a managed domain have the same settings.

Each group of servers is assigned a unique profile. A profile consists of the list of EAP subsystems and their configuration, which will be used by the different subsystems that make up the profile.

Server groups are configured in the domain.xml file on the domain controller. The recommended approach to creating and managing server groups in a managed domain is to use the EAP administration console or the JBoss EAP CLI (recommended).

Configuring server groups using the administrative console *Server groups* in a managed domain can be created and managed in the Runtime section of the administrative console.

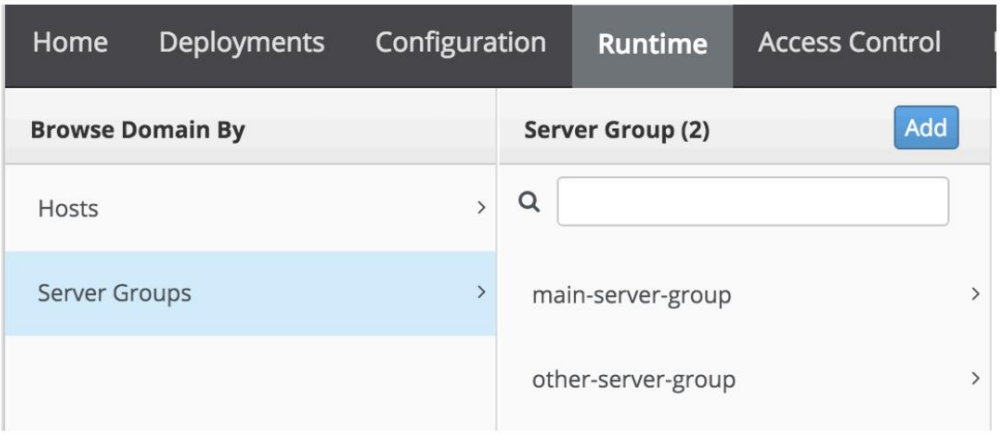


Figure 5.5: View of the server groups on the Runtime page

To add a new server group, three key attributes must be present:

- **a unique Name for the server group.**
- **a valid Profile from the list of profiles in the domain.xml configuration file, which controls the active subsystems in a given profile.**
- **a Socket Binding, which defines a set of network ports that will be used by servers running within a server group.**

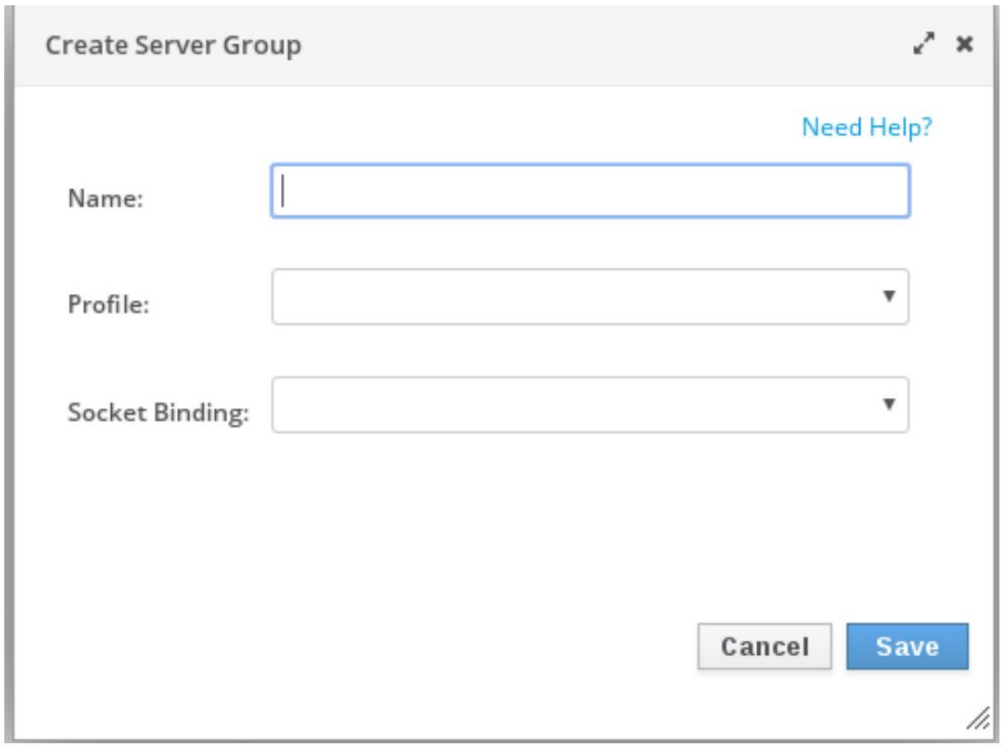


Figure 5.6: Create a server group

To delete a server group, make sure there are no servers assigned to the server group. If there are servers assigned, first stop them, and then remove the servers from the server pool. Finally, delete the server group.

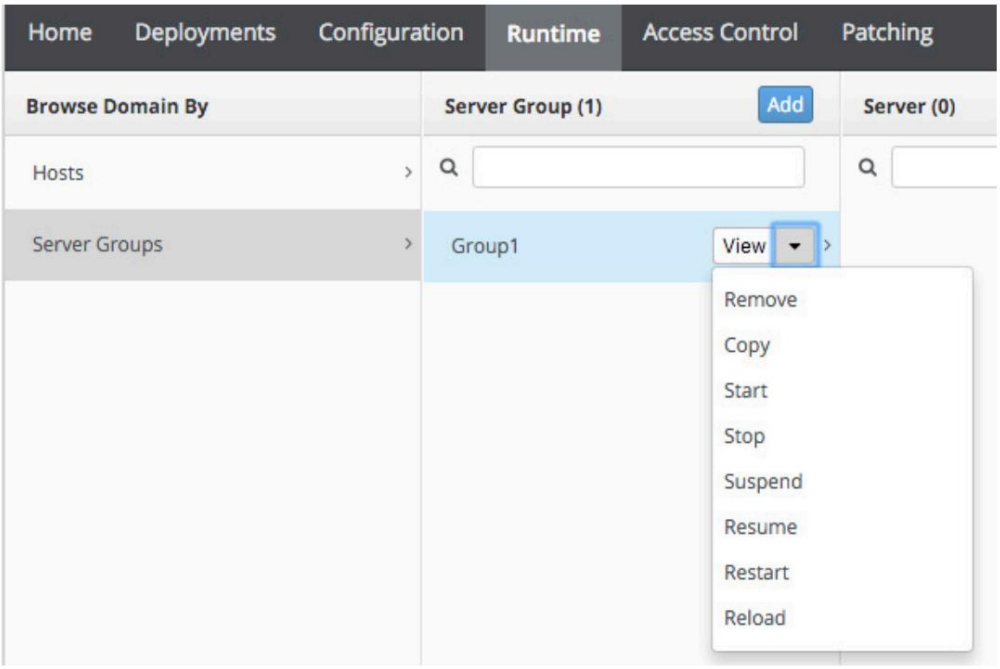


Figure 5.7: Delete a server group

Configure server groups using the CLI

JBoss EAP

Server groups can also be created and managed using the JBoss EAP CLI. The advantage of this approach is that the steps to create and manage the server groups can be performed with scripts as part of an automated workflow, and it is faster for creating multiple server groups in batch mode. This approach also has the option to atomically allocate or revert multiple steps in batch mode, which ensures the integrity of the domain.xml configuration file in the event of an error during CLI command execution.

Server groups are configured in the /server-group namespace of the JBoss EAP CLI. There are commands to create, view, modify, and delete server groups in this namespace. To add a new server group, run the following command:

```
[domain@workstation:9990 /] /server-group=Group1:add\
(profile=full,socket-binding-group=full-sockets) {

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

To verify that the server groups have been created, use the read-resource attribute in the EAP CLI:

```
[domain@workstation:9990 /] /server-group=Group1:read-resource {

        "outcome" => "success",
        "result" =>
                { "management-subsystem-endpoint" => false,
```

```
            "profile" => "default", "socket-
            binding-default-interface" => undefined, "socket-binding-group"
            => "standard-sockets", "socket-binding-port-offset" => 0,
            "deployment" => undefined, "deployment-
            overlay" => undefined, "jvm" =>
            undefined, "system-property" => undefined


        }
}
```

To delete a server group, make sure there are no servers assigned to the server group. Next, to delete the server group, run the following command:

```
[domain@workstation:9990 /] /server-group=Group1:remove() {

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

use

Regardless of the technique used to modify a configuration setting, all changes are synchronized with the domain.xml configuration file. If, for example, a value is changed through the management console, the underlying domain.xml file is updated, and the CLI will detect the change instantly. In the next section, the server group configuration attributes in the domain.xml file will be discussed in more detail.

Server group configuration

The server groups in a managed domain are defined in the domain.xml file of the domain controller. Server groups are defined by the <server-groups> element. The <server-group> child element within the <server-groups> parent element can be used to define a group of servers.
Here is a sample <server-group> definition:

```
<server-groups>
    <server-group name="main-server-group" profile="default" >
        <jvm name="default" >
            ...
        </jvm>
        <socket-binding-group ref="standard-sockets" />
        <deployments>
            ...
        </deployments> </
    server-group> </
server-groups>
```

1 **The name attribute is required and must be unique within the managed domain. When defining a server in host.xml, the server references this name attribute.**

2. The profile attribute is also required and refers to the name of a <profile> defined in the domain configuration file.
3. The jvm name attribute references a <jvm> memory configuration for all servers in this server group. Also, the server group values for the JVM can be overridden by the server in host.xml. JVM memory allocation in EAP is discussed later in this course.
4. The socket-binding-group attribute refers to the name of a socket binding group defined in this domain configuration file. This value can be overridden by the server in host.xml.
5. The <deployments> section lists the applications that should be deployed to each server that is part of the group. Instead of manually adding entries in <deployment>, applications are deployed through the management console or CLI.

use

The domain controller and host controllers must ensure that all servers in a server group have a consistent configuration.
All servers are configured with the same profile assigned to the server group, and will have the same content deployed.

Deploying applications in a managed domain

Deploying an application to a managed domain requires a different process than that used to deploy in standalone mode. You cannot select a specific server to deploy an application to. All applications must be deployed to a pool, and then all servers belonging to the specified pool deploy that application. The host controller communicates with the domain controller and ensures that deployments are synchronized across all servers in the managed domain that are part of the pool.

There are two ways to deploy an application using domain mode:

- The management console
- The CLI tool

Unlike standalone mode, it cannot be deployed using the deployment-scanner subsystem. This is because it is not possible to guarantee that the application is available to the entire pool (manually deployed for each server).

Application deployment using the management console

Application deployment using the management console is a two-step process, in which the application (EAR, WAR, JAR, etc.) is first uploaded to the *content repository* , and then deployed to a server pool selected in the managed domain.
The content repository is a folder on the domain controller located at DOMAIN_BASE_DIR/domain/content, where DOMAIN_BASE_DIR is the base directory where the managed domain configuration files are stored.

Using the Content Repository menu, click Add to upload a new implementation to the repository.

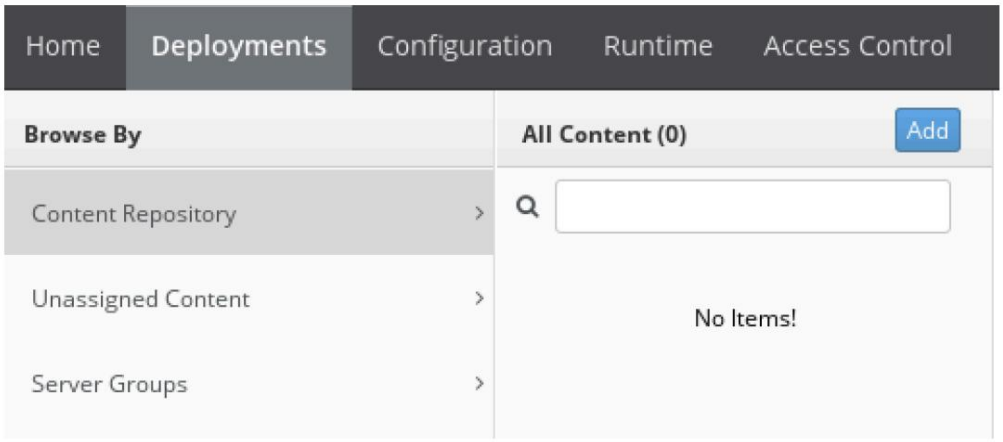


Figure 5.8: EAP Managed Console - Empty Content Repository page

A wizard will begin uploading a new application to the repository. In the first step of the wizard, the deployment type is requested. Two options are available:

- Upload a new implementation – Using this option uploads an application, which is available in the repository. A reference to the application is created in the domain.xml file.
- Create an unmanaged deployment – When using this option, you must specify a path where the application file is available. Deployment content will not be uploaded to the repository and will be deployed directly from the specified location.

use

To deploy an application using an unmanaged method, the application file must be available in the same path to all hosts. This is NOT a recommended practice, as you may end up having different application files on some hosts. The unmanaged method exists for applications that must be deployed as expanded applications. Also note that such applications violate the JEE specifications.

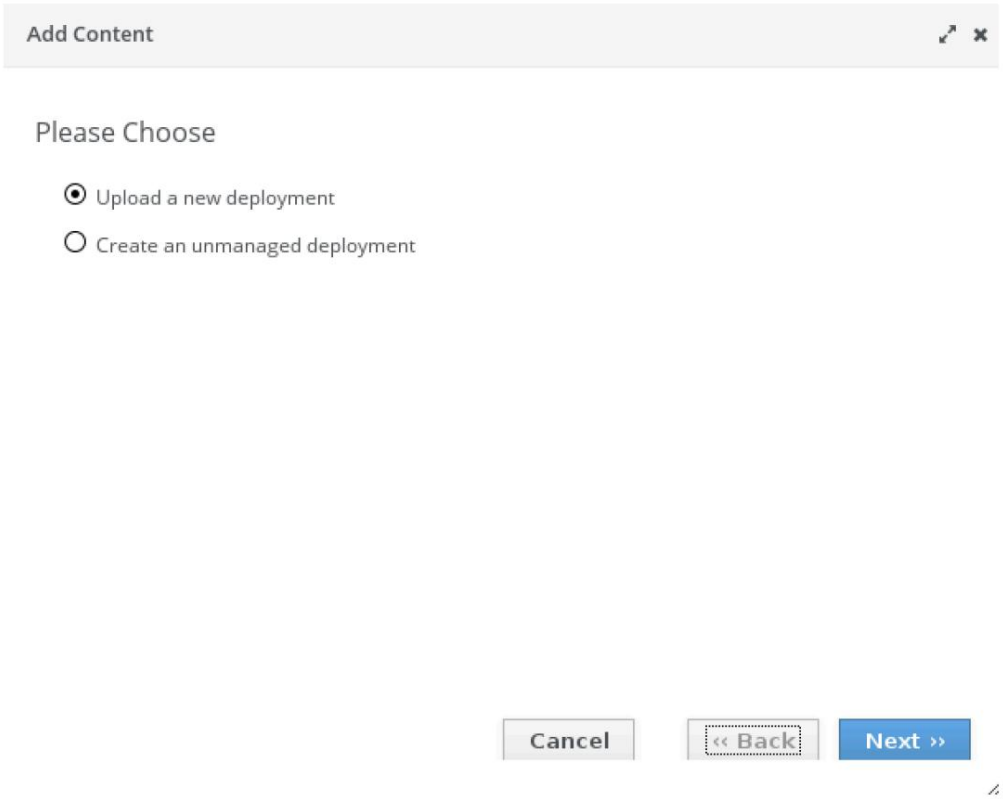


Figure 5.9:
EAP Managed Console - Managed and Unmanaged Deployments

When using the Upload a new implementation option, the second step prompts you to upload the file. Click Browse to navigate to and select the required file.

In the next step of the wizard, two options must be defined:

- **Name: The identifier of the implementation. This value must be unique across all implementations.**
- **Runtime Name: Defines the context of the application. The context is the name of the application in the runtime environment. If a deployment has a runtime name defined as myapp.war, it will be available at http://server:port/myapp.**

Applications added to the Content Repository can then be assigned to one or more server groups using the Assign button next to the deployments.

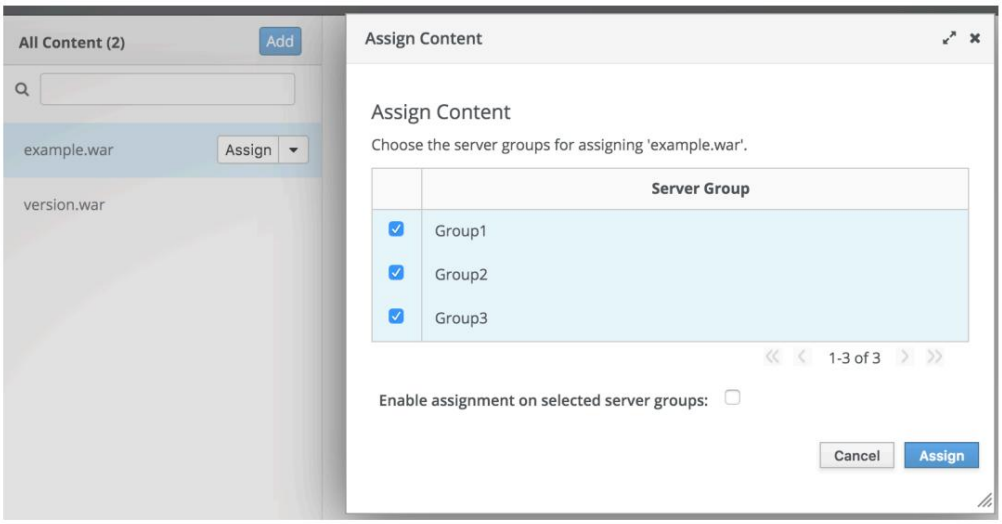


Figure 5.10: Assignment of a content

Another approach is to choose a server group from the Server Group section of the Deployments tab in the management console and select applications that should be deployed to this server group.

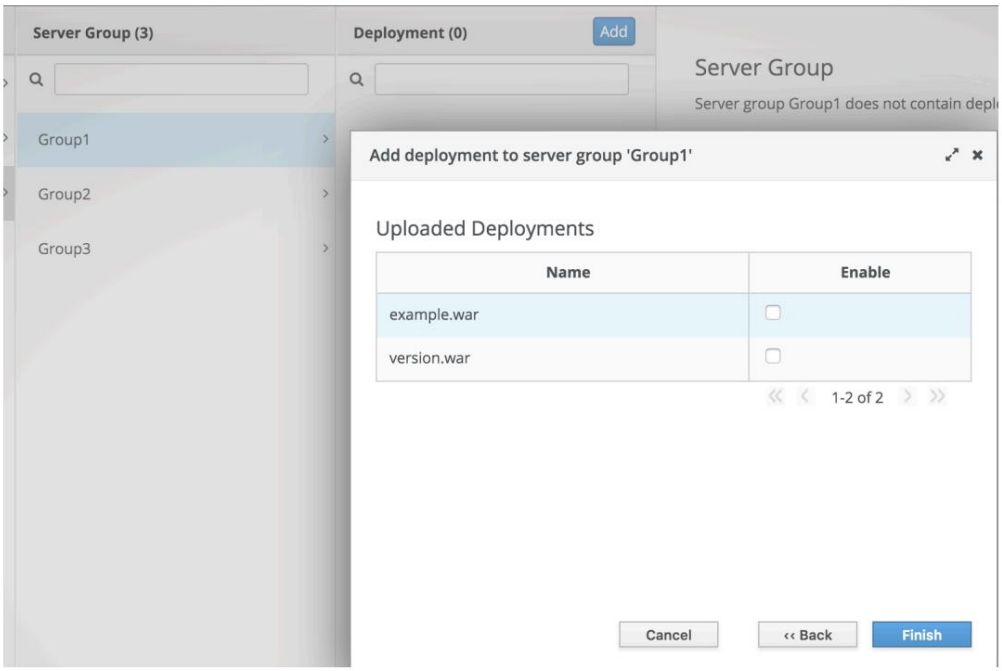


Figure 5.11: Adding a Deployment to a Server Group

Applications can be deallocated from a pool to which they were assigned prior to undeploying pool applications.

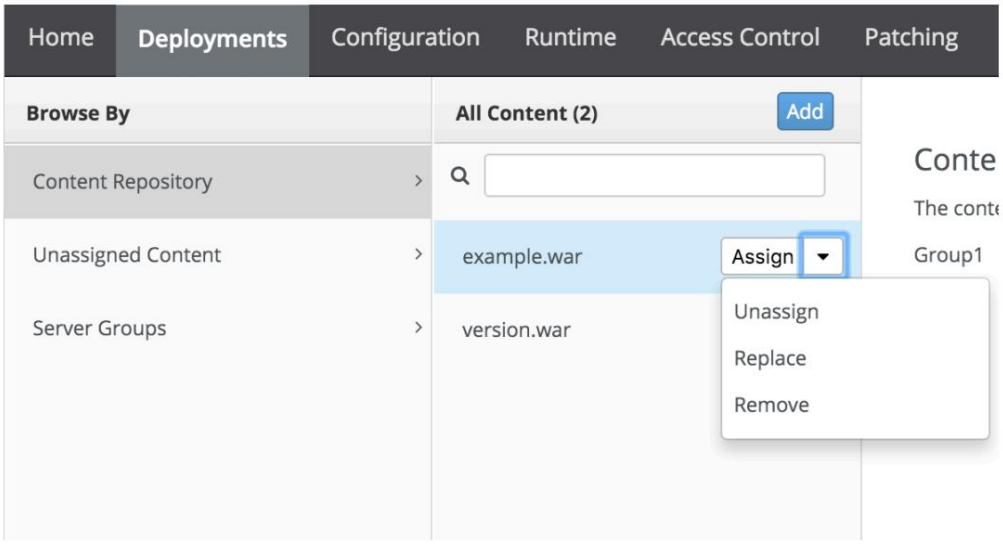


Figure 5.12: Deallocate a deployment from all server groups

Alternatively, applications can be deallocated or disabled from the Server Group section of the Deployments tab in the management console.

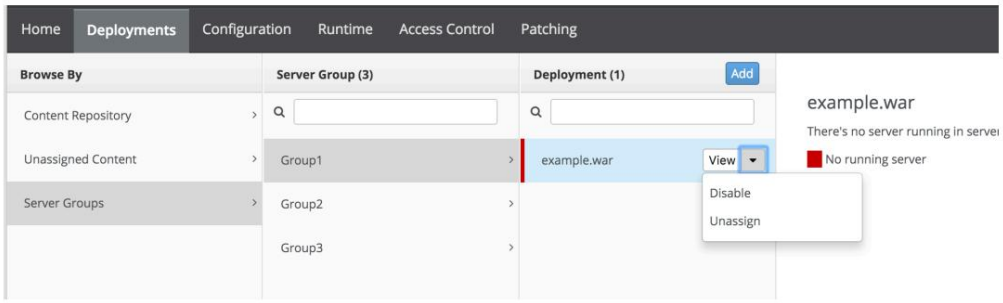


Figure 5.13: Disable a deployment on a server group

Deploying applications using the CLI

JBoss EAP

Deploying applications using the EAP CLI gives administrators the benefit of a Command Line Interface with the ability to create and run deployment scripts. An administrator can use this scripting capability to configure deployment and management scenarios for specific applications.
An administrator can manage deployments for an entire network of servers running in a managed domain from a single point of control.

The advantage of this approach is that the steps to deploy and manage the applications can be scripted as part of an automated workflow, and it is faster for deploying multiple applications to a managed domain in batch mode.

The JBoss EAP CLI provides the deploy and undeploy commands in the default top-level namespace to deploy and undeploy applications to a managed domain. One of the advantages of the CLI approach is that an application can be deployed to *all* server groups using a single command:

```
[domain@workstation:9990 /] deploy /path/to/example.war --all-server-groups
```

To deploy an application to a specific server group, provide a comma-separated list of server groups as arguments to the deploy command:

```
[domain@workstation:9990 /] deploy /path/to/example.war \ --server-
groups=Group1,Group2,Group3
```

To undeploy a server group application, the EAP CLI provides a handy --all-relevant-server-groups option, because the CLI has already detected the deployments and keeps track of the server groups to which they are deployed. which application was assigned:

```
[domain@workstation:9990 /] undeploy example.war --all-relevant-server-groups
```

An explicit list of server groups from which to undeploy an application can be provided, if the undeployment of the application is not to be performed on all relevant server groups:

```
[domain@workstation:9990 /] undeploy example.war --server-groups=Group1,Group2
```

Undeploy an Application Using the CLI The undeploy command undeploys an application from the domain. To undeploy and remove an application from the entire domain, the name argument and the --all-relevant-server-groups option must be declared:

```
[domain@172.25.250.254:9990 /] undeploy myapp.war --all-relevant-server-groups
```

In the example above, the app is also removed from the content repository. To keep the app in the repository, use the --keep-content argument:

```
[domain@172.25.250.254:9990 /] undeploy myapp.war --all-relevant-server-groups \ --keep-content
```

It is possible to specify only specific groups with the --server-groups argument:

```
[domain@172.25.250.254:9990 /] undeploy myapp.war --server-groups=main-server-group \ --keep-content
```

Important

If the application is assigned to two or more groups, the --keep content argument is required, since the content of the application cannot be removed.