**Automate JBoss Patching with Jenkins Pipeline | Full Step-by-Step Guide**

**Description**

A Jenkins pipeline can be used to automate the process of patching JBoss servers. This video demonstrates how to streamline and improve the efficiency of JBoss patching by leveraging the power of Jenkins pipelines.

**The Problem with Manual Patching**

- Patching JBoss servers manually is time-consuming and error-prone.

- It can lead to downtime and security vulnerabilities.

- Manual patching also makes it difficult to track which servers have been patched and which versions of patches are installed.

**Why Automate JBoss Patching?**

- Automating JBoss patching with Jenkins pipelines can streamline the process.

- This approach minimizes downtime and ensures that all servers are patched consistently and securely.

- It also improves efficiency and allows you to focus on other tasks.

**Prerequisites**

1. Jenkins installed on Windows.
2. Basic knowledge of Jenkins pipelines.

**Step 1: Installing JBoss EAP 7.4**

1. Download JBoss from the official Red Hat website.
2. Extract the archive and set up the folder structure.
3. Start JBoss with the CLI command.
4. Access JBoss Admin Console

**Step 2: Manually Applying a Patch in JBoss**

1. Download the latest patch file.
2. Open JBoss CLI and run
   1. C:\Users\admin\EAP-7.4.0\bin\jboss-cli.bat –connect
3. Apply the patch manually.
4. Restart JBoss after applying the patch.
5. Verify the new version.

**Step 3: Automating JBoss Patching with Jenkins**

1. Add **Jenkins job** and paste the pipeline script.

pipeline {

agent any

environment {

JBOSS\_HOME = "C:\\Users\\admin\\EAP-7.4.0"

PATCH\_PATH = "C:\\Users\\admin\\EAP-7.4.0\\Patch\\jboss-eap-7.4.1-patch.zip"

}

stages {

stage('Backup Existing JBoss') {

steps {

bat """

xcopy /E /I /Y %JBOSS\_HOME% %JBOSS\_HOME%\_backup

"""

}

}

stage('Apply Patch') {

steps {

bat """

"C:\\Users\\admin\\EAP-7.4.0\\bin\\jboss-cli.bat" --connect --command="patch apply %PATCH\_PATH%"

"""

}

}

stage('Restart JBoss') {

steps {

bat """

"C:\\Users\\admin\\EAP-7.4.0\\bin\\jboss-cli.bat" --connect --command="shutdown --restart=true"

"""

}

}

stage('Verify Patch') {

steps {

bat """

type %JBOSS\_HOME%\\version.txt

"""

}

}

}

post {

success {

echo 'JBoss Patching Completed Successfully!'

}

failure {

echo 'Patching Failed. Check logs for details.'

}

}

}

1. Run the pipeline and check logs.

**Step 4: Running the Jenkins Job & Validating Patch**

1. Execute the pipeline and verify that:
   1. Backup is created.
   2. Patch is applied.
   3. JBoss is restarted.
   4. Version file is updated.