

## 1. Launch CM-Master Node Lab in Course 4: PG DO – Configuration Management with Chef, Puppet and Ansible

BACK

SELF LEARNING

LIVE CLASSES

PRACTICE LABS

ASSESSMENT

CERTIFICATE

### PG DO - Configuration Management with Chef, Puppet and Ansible

0 Classes completed | 97% Self-Learning Videos Watched | 1/2 Projects Done

#### CM-Master Node AWS

Dear Learner,

**\*\*IMP:** We continuously work towards improving the learning experience. As a part of it, we are migrating the labs, we would request you to kindly download the instruction document which helps you to retain the data from the old lab to the new lab from [here](#).

Dear learner,

Please note: This lab is configured based on the curriculum covered during the live virtual classes.  
All details pertaining to the exercises in this lab are provided in the e-books available in your LMS account.

##### Lab Instructions

- When you go to the practice Lab page click on the LAUNCH LAB button and it would redirect you to the login credential page.
- To start the VM, just scroll in where the credentials are present and click on the Start Instance button and wait for some time.
- When the status in the Instance Action show as "The lab environment is available for access" then open the Auth URL provided there, enter the credentials.
- Copy-paste will not work on the login page and the credentials are to be entered manually.
- For the first time, the lab setup can take up to 15 mins to start the instance and you might need to refresh the page if it takes longer.
- Caps Lock does not work on the VM, hence use the LSHIFT key for CapsLock.
- Do not click on the terminate lab instance button as it will delete your VM.
- The Stop instance button will pause the VM until your next usage.
- Tools installed: GIT, Java, Docker, Eclipse

##### Session Time:

- There is an idle time set for 30 minutes.
- The Lab will automatically end after the idle time is complete.
- You can start a fresh session again if required.
- Your data will be restored when you start a fresh session

##### Important:

- Resources created during the Lab session is for learning purposes only.
- Please do NOT plan to use the resources for any other personal/commercial use.

Your Labs are ready. [LAUNCH LAB](#)

CM-Master Node AWS

This Lab will get reset on 07th August 2021, 2:10 PM

Current Lab : DevOps Practitioner

Access Information

Lab Details

Components

Log Details

Usage Details

Applications

RDP Access

RDP Access

Username

stephenwangnc

Copy

Password

.....

Eye

Copy

Auth Url

https://stephenwangnc.c


Copy

Instance Actions

Stop Instance

Instance status : **Running**

Last updated at : 2021-09-10 01:43




#### DevOps Practitioner

Category: DEVOPS  
Start Date: 2021-09-09 00:00  
End Date: 2021-09-25 23:59  
Code: SL-Lab0001

This is a dedicated lab contains DevOps tools pre-installed in the Ubuntu VM enabling DevOps practitioners to get hands-on experience on different use-cases.

[TERMINATE LAB ACCESS](#)

Powered by

 CORESTACK

[Terms & Conditions](#)

PG DO - Configuration Management with Chef, Puppet and Ansible


0 Classes completed | 97% Self-Learning Videos Watched | 1/2 Projects Done


CM-Master NodeAWS

Current Lab : AWS Certification - Dedicated Account

Access InformationLab DetailsComponentsLog DetailsUsage Details

Applications

AWS Web Console

AWS API Access

Auth Url


https://signin.aws.amazon.com/federation

Session Expires in: 7h 56m 48s

Refresh Link

1. Session Duration is for 8 Hours. Post the session duration all the resources will be cleaned up automatically.

2. Auth URL enables Single-Sign-On, so the URL will vary for each session and the same URL will not work next time. Refresh the Access Details page if the

AWS Certification - Dedicated Account

Category: Cloud Computing  
Start Date: 2021-09-09 00:00  
End Date: 2021-10-09 23:59  
Code: SLAWS

Amazon Web Services (AWS) offers a suite of cloud-computing services that make up an on-demand computing platform. AWS has more than 70 services, spanning a wide range, including compute, storage, networking, database, analytics, application services, deployment, management, mobile, developer tools and tools for the Internet of things.

TERMINATE LAB ACCESS

Powered by  
CORESTACK

Terms & Conditions

awsServices

Search for services, features, marketplace products, and docs

Option+5

Corestack\_Role/stephen.wang\_ngr @ 8638-0486-4429 N. Virginia Support

1. Choose AMI2. Choose Instance Type3. Configure Instance4. Add Storage5. Add Tags6. Configure Security Group7. Review

## Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace, or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"


Search by Systems Manager parameter

Quick Start

My AMIs

AWS Marketplace


Community AMIs

Amazon Linux

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-087c17d1fe0178315 (64-bit x86) / ami-029c64b3c205e6cce (64-bit Arm)

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is approaching end of life on December 31, 2020 and has been removed from this wizard.


Root device type: ebsVirtualization type: hvmENA Enabled: Yes

macOS

macOS Big Sur 11.5.2 - ami-098c730dfe1aabb81

The macOS Big Sur AMI is an EBS-backed, AWS-supported image. This AMI includes the AWS Command Line Interface, Command Line Tools for Xcode, Amazon SSM Agent, and Homebrew. The AWS Homebrew Tap includes the latest versions of multiple AWS packages included in the AMI.


Root device type: ebsVirtualization type: hvmENA Enabled: Yes

macOS

macOS Catalina 10.15.7 - ami-063d8009e26b3b6b2

The macOS Catalina AMI is an EBS-backed, AWS-supported image. This AMI includes the AWS Command Line Interface, Command Line Tools for Xcode, Amazon SSM Agent, and Homebrew. The AWS Homebrew Tap includes the latest versions of multiple AWS packages included in the AMI.


Root device type: ebsVirtualization type: hvmENA Enabled: Yes

macOS

macOS Mojave 10.14.6 - ami-065a39c2203f00885

The macOS Mojave AMI is an EBS-backed, AWS-supported image. This AMI includes the AWS Command Line Interface, Command Line Tools for Xcode, Amazon SSM Agent, and Homebrew. The AWS Homebrew Tap includes the latest versions of multiple AWS packages included in the AMI.

Root device type: ebsVirtualization type: hvmENA Enabled: Yes

Red Hat

Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-0b0af3577fe5a3532 (64-bit x86) / ami-01fc429821bf1f4b4 (64-bit Arm)

Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebsVirtualization type: hvmENA Enabled: Yes

## Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All Instance familiesCurrent generationShow/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GiB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	t2	t2.micro	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes
<input type="checkbox"/>	t2	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
<input type="checkbox"/>	t3	t3.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes
<input type="checkbox"/>	t3	t3.micro	2	1	EBS only	Yes	Up to 5 Gigabit	Yes
<input type="checkbox"/>	t3	t3.small	2	2	EBS only	Yes	Up to 5 Gigabit	Yes
<input type="checkbox"/>	t3	t3.medium	2	4	EBS only	Yes	Up to 5 Gigabit	Yes
<input type="checkbox"/>	t3	t3.large	2	8	EBS only	Yes	Up to 5 Gigabit	Yes
<input type="checkbox"/>	t3	t3.xlarge	4	16	EBS only	Yes	Up to 5 Gigabit	Yes

Cancel

Previous

Review and Launch

Next: Configure Instance Details

2

**Step 7: Review Instance Launch**  
Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

**⚠️ Improve your instances' security. Your security group, launch-wizard-1, is open to the world.**  
Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only. You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

**AMI Details** [Edit AMI](#)

**Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-087c17d1fe0178315**  
Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is a...  
Root Device Type: ebs    Virtualization type: hvm

**Instance Type** [Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	-	1	1	EBS only	-	Low to Moderate

**Security Groups** [Edit security groups](#)

Security group name: launch-wizard-1  
Description: launch-wizard-1 created 2021-09-09T18:49:20.092-07:00

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	0.0.0.0/0	

**Instance Details** [Edit instance details](#)

**Storage** [Edit storage](#)

[Cancel](#)
[Previous](#)
[Launch](#)

## 2. Use Github to create and clone capstone project to CM-Master Note

```
stephenwangngc@stephenwangngc:~$ ls
Desktop  thinclient_drives
stephenwangngc@stephenwangngc:~$ git clone https://github.com/stephengineer/DevOpsCapstone.git
Cloning into 'DevOpsCapstone'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), done.
Checking connectivity... done.
stephenwangngc@stephenwangngc:~$
```

## 3. Set up the Jenkins server in master or slave architecture

```
stephenwangngc@stephenwangngc:~/DevOpsCapstone$ sudo systemctl start jenkins
stephenwangngc@stephenwangngc:~/DevOpsCapstone$ sudo systemctl status jenkins
● jenkins.service - LSB: Start Jenkins at boot time
   Loaded: loaded (/etc/init.d/jenkins; bad; vendor preset: enabled)
   Active: active (exited) since Thu 2021-09-09 20:41:48 UTC; 1min 48s ago
     Docs: man:systemd-sysv-generator(8)

Sep 09 20:41:46 stephenwangngc systemd[1]: Starting LSB: Start Jenkins at boot time...
Sep 09 20:41:46 stephenwangngc jenkins[31966]: Correct java version found
Sep 09 20:41:46 stephenwangngc jenkins[31966]: * Starting Jenkins Automation Server jenkins
Sep 09 20:41:46 stephenwangngc su[32022]: Successful su for jenkins by root
Sep 09 20:41:46 stephenwangngc su[32022]: + ??? root:jenkins
Sep 09 20:41:46 stephenwangngc su[32022]: pam_unix(su:session): session opened for user jenkins by (uid=0)
Sep 09 20:41:48 stephenwangngc jenkins[31966]: ...done.
Sep 09 20:41:48 stephenwangngc systemd[1]: Started LSB: Start Jenkins at boot time.
Sep 09 20:43:06 stephenwangngc systemd[1]: Started LSB: Start Jenkins at boot time.
stephenwangngc@stephenwangngc:~/DevOpsCapstone$
```

The screenshot shows the Jenkins Dashboard in a web browser. The browser's address bar indicates the URL is `localhost:8080`. The Jenkins logo is in the top left, and a search bar is in the top center. On the right of the top bar, there are notification icons (a bell with '1' and a shield with '1') and a user profile for 'Stephen Wang' with a 'log out' button.

The main content area is titled 'Welcome to Jenkins!' and includes the text: 'This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.' Below this, there are three main sections:

- Start building your software project**: Contains a button 'Create a job' with a right arrow.
- Set up a distributed build**: Contains two buttons: 'Set up an agent' and 'Configure a cloud', both with right arrows, and a link 'Learn more about distributed builds' with an external link icon.

On the left sidebar, under the 'Dashboard' header, there are links for 'New Item', 'People', 'Build History', 'Manage Jenkins', 'My Views', 'Lockable Resources', and 'New View'. Below these, the 'Build Queue' section shows 'No builds in the queue.' The 'Build Executor Status' section shows two executors, both in an 'Idle' state.

At the bottom right of the dashboard, it says 'REST API' and 'Jenkins 2.303.1'.

4. Build an image using the artifacts and deploy them on containers
5. Remove the container stack after completing the job

```

48 lines (46 sloc) | 1.05 KB
Raw Blame
1 pipeline {
2   environment {
3     registry = "stephenwangmath/devopscapstone"
4     registryCredential = 'dockerhub'
5     dockerImage = ''
6   }
7
8   agent any
9
10  stages {
11    stage('Clear Workspace') {
12      steps {
13        deleteDir()
14      }
15    }
16    stage('Checkout') {
17      steps {
18        checkout scm
19      }
20    }
21    stage('Setup') {
22      steps {
23        sh 'npm install'
24      }
25    }
26    stage('Build Image') {
27      steps {
28        script {
29          dockerImage = docker.build registry + ":\$BUILD_NUMBER"
30        }
31      }
32    }
33    stage('Deploy Image') {
34      steps {
35        script {
36          docker.withRegistry( '', registryCredential ) {
37            dockerImage.push()
38          }
39        }
40      }
41    }
42    stage('Remove Image') {
43      steps {
44        sh "docker rmi \$registry:\$BUILD_NUMBER"
45      }
46    }
47  }
48 }

```

6. Use the Jenkins plugins to perform the computation part on the Docker containers

The image shows two screenshots of the Jenkins web interface. The top screenshot displays the 'Available Plugins' page with a search for 'docker'. The bottom screenshot shows the 'Global credentials (unrestricted)' page.

**Jenkins Plugin Manager - Available Plugins**

Search: docker

Updates | **Available** | Installed | Advanced

Install	Name	Version	Released
<input checked="" type="checkbox"/>	<b>Docker</b> Cloud Providers Cluster Management and Distributed Build This plugin integrates Jenkins with <b>Docker</b>	1.2.3	21 days ago
<input type="checkbox"/>	<b>Docker Commons</b> api-plugin Library plugins (for use by other plugins) Provides the common shared functionality for various Docker-related plugins.	1.17	1 yr 2 mo ago
<input type="checkbox"/>	<b>Docker Pipeline</b> Deployment DevOps Build and use Docker containers from pipelines.	1.26	6 mo 17 days ago
<input type="checkbox"/>	<b>Docker API</b> api-plugin This plugin provides <b>docker-java</b> API for other plugins. <b>This plugin is up for adoption!</b> We are looking for new maintainers. Visit our <a href="#">Adopt a Plugin</a> initiative	3.1.5.2	1 yr 5 mo ago

Buttons: Install without restart, Download now and install after restart

Update information obtained: 28 min a

**Jenkins Global credentials (unrestricted)**

System » Global credential »

Back to credential domains

Add Credentials

Credentials that should be available irrespective of domain specification to requirements matching.

ID	Name	Kind	Description
<b>stephenGithub</b>	stephengineer/***** (StephenGithub)	Username with password	StephenGithub

Icon: S M L

## 7. Create Jenkins pipeline script

The screenshot shows the Jenkins web interface for a pipeline named 'Capstone'. The left sidebar contains navigation links: Back to Dashboard, Status, Changes, Build Now (highlighted), Configure, Delete Pipeline, Full Stage View, Rename, Pipeline Syntax, GitHub Hook Log, and Git Polling Log. The main content area is titled 'Pipeline Capstone' and includes a 'Recent Changes' section. Below this is the 'Stage View' table, which shows the duration of each stage for the latest build (#20).

	Declarative: Checkout SCM	Clear Workspace	Checkout	Setup	Build Image	Deploy Image	Remove Image
Average stage times: (Average full run time: ~12s)	696ms	290ms	951ms	3s	2s	1s	1s
#20 Sep 10 01:37 2 commits	696ms	290ms	951ms	3s	2s	1s	1s

Below the stage view is the 'Permalinks' section, which lists links to the last build, last stable build, last successful build, and last completed build, all of which are build #20.

## 8. Use the GIT web hook to schedule the job on check-in or poll SCM

The screenshot shows the 'Build Triggers' configuration page for the 'Capstone' pipeline. The page has tabs for General, Build Triggers (selected), Advanced Project Options, and Pipeline. Under 'Build Triggers', the following options are checked: 'GitHub hook trigger for GITscm polling' and 'Poll SCM'. The 'Schedule' field is set to 'H 8 \* \* \*'. Below this, a message indicates the last run was on Thursday, September 9, 2021 at 8:08:35 AM UTC, and the next run will be on Friday, September 10, 2021 at 8:08:35 AM UTC. Other options like 'Ignore post-commit hooks', 'Disable this project', 'Quiet period', and 'Trigger builds remotely' are unchecked. At the bottom, there are 'Save' and 'Apply' buttons.