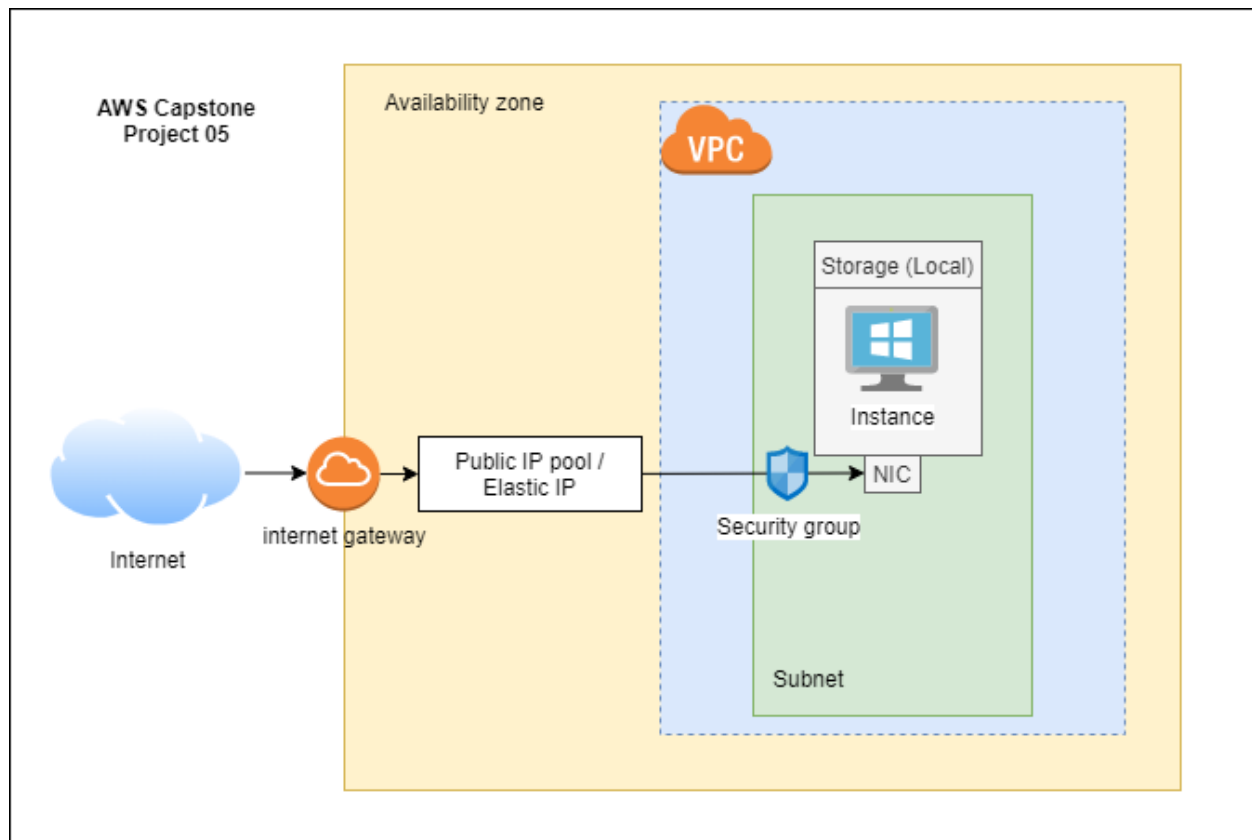


AWS Capstone Project Project 05

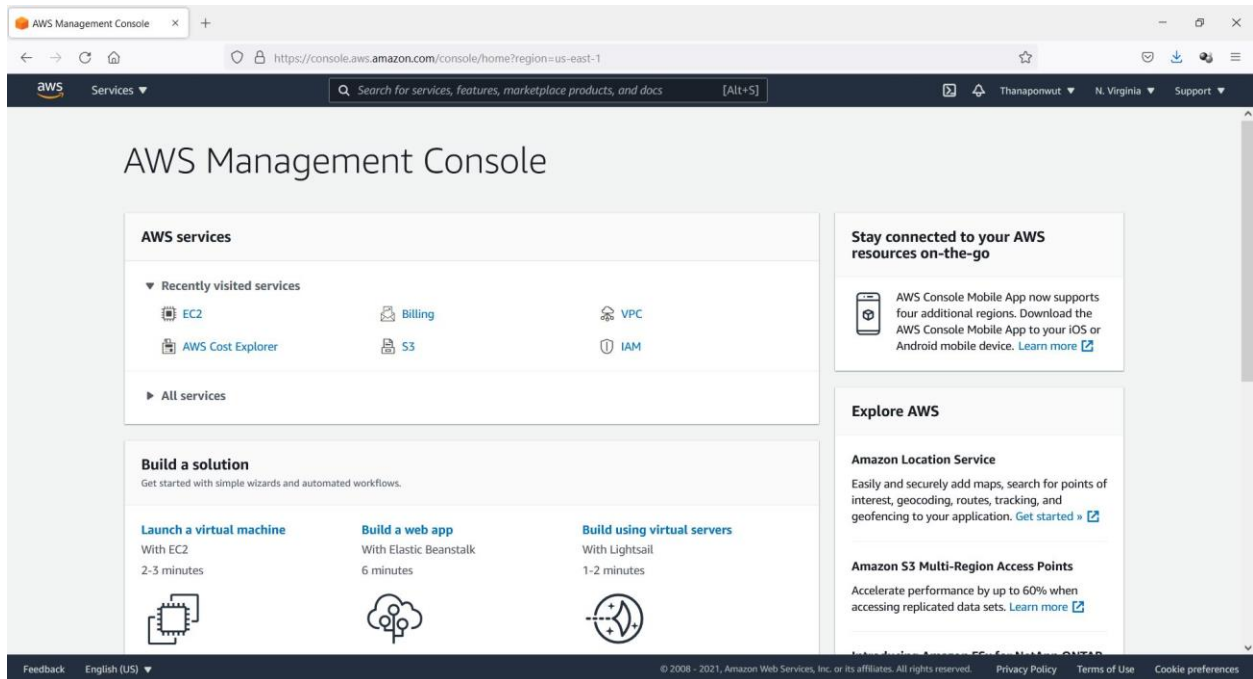
Requirements

1. How you configure security groups, subnets and basic IAM requirements
2. How you configure your VPC and networking requirements
3. How you deploy CloudFormation Templates

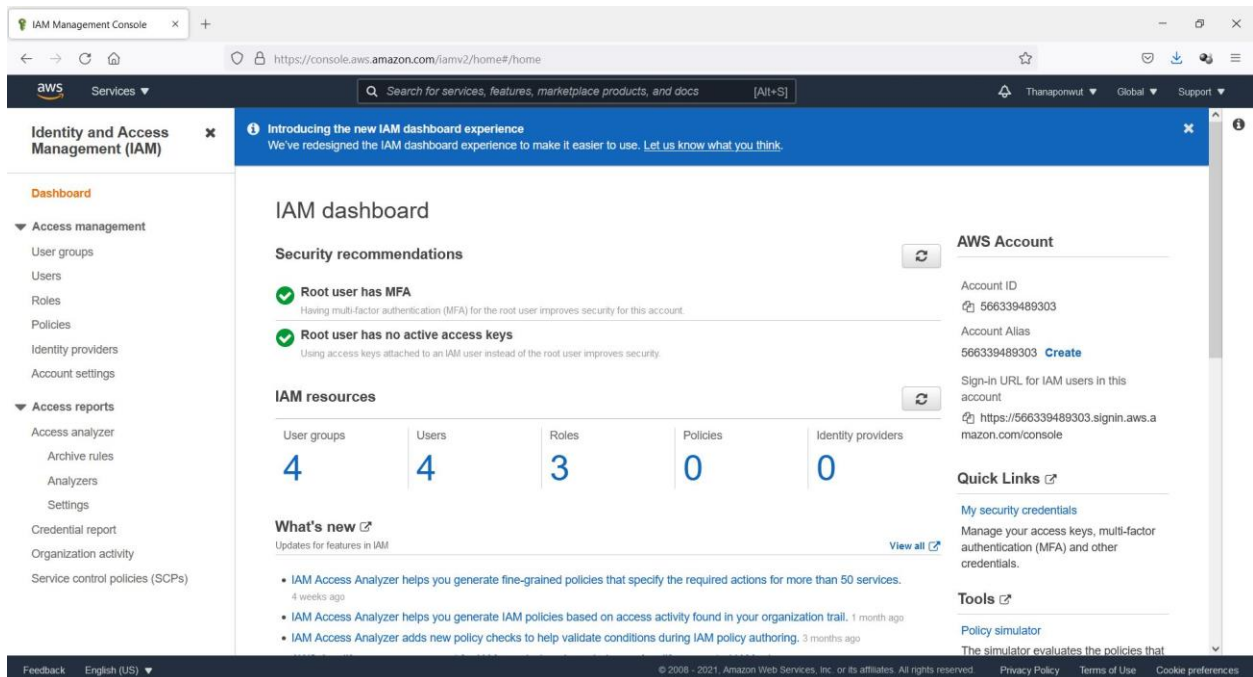
- High-level diagram



- AWS Console

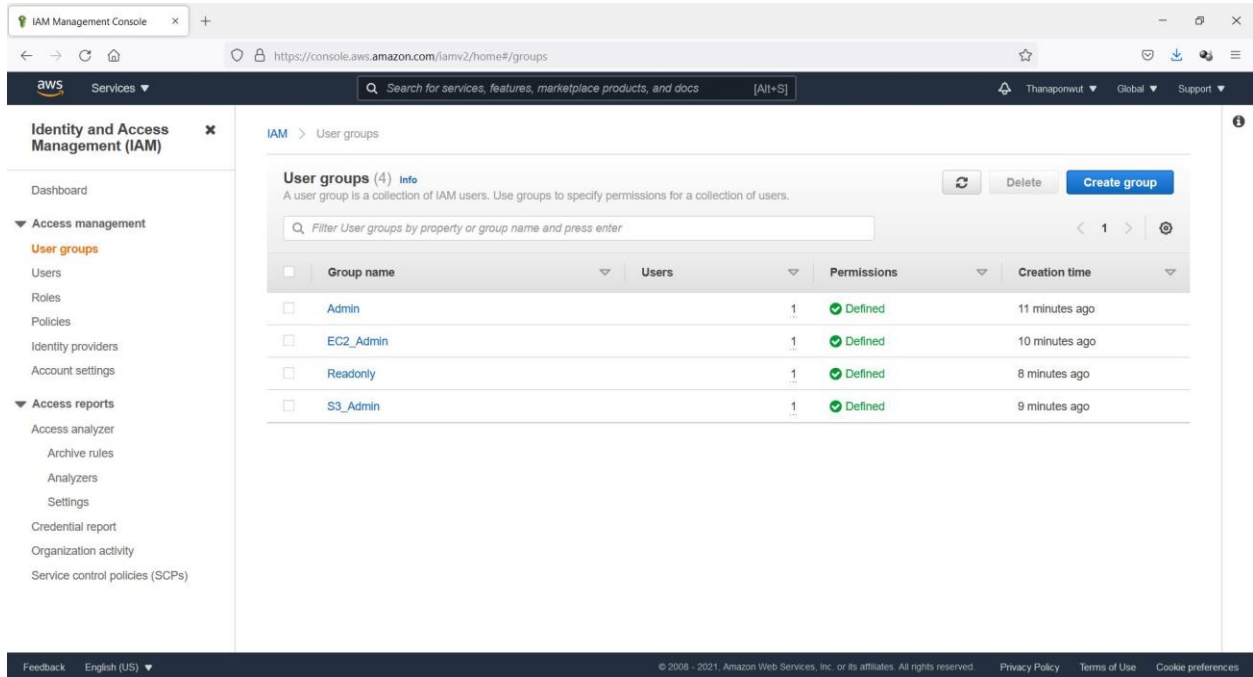


- IAM Dashboard



- User groups

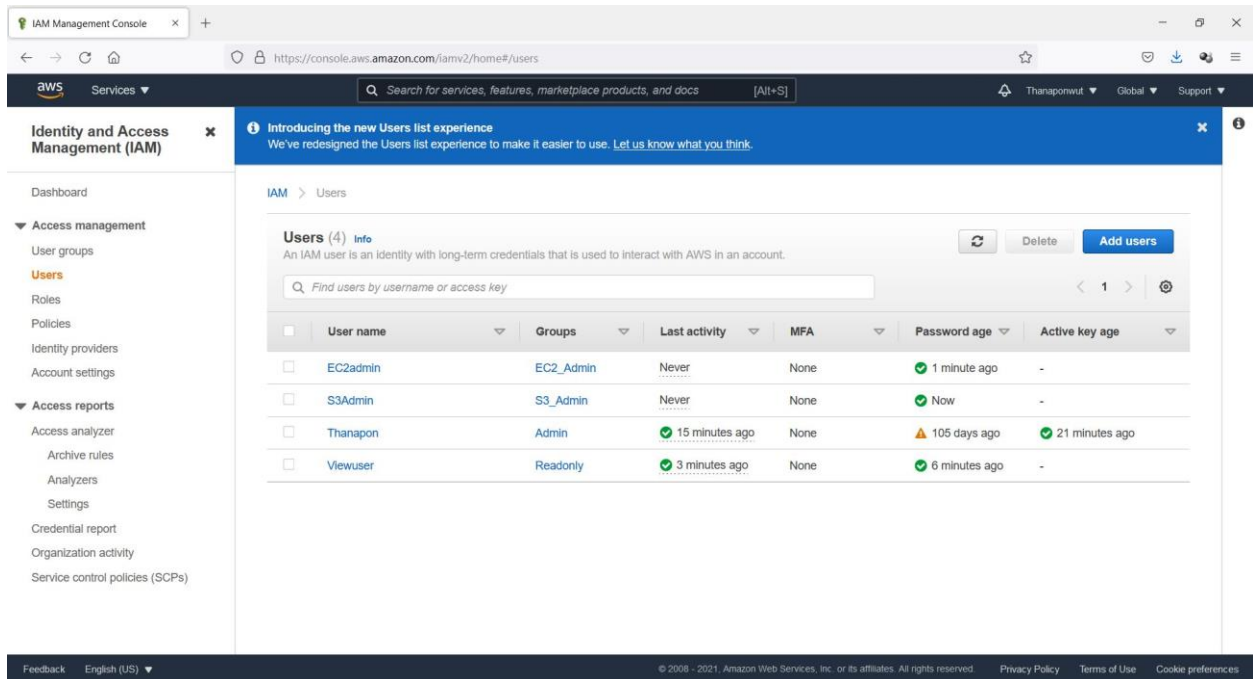
- Admin – Provides full access to AWS services and resources.
- EC2_Admin – Provides full access to Amazon EC2 via the AWS Management Console.
- Readonly - Provides read-only access to AWS services and resources.
- S3_Admin - Provides full access to all buckets via the AWS Management Console.



The screenshot displays the AWS IAM Management Console interface. The left sidebar shows the 'Identity and Access Management (IAM)' section with a list of options including Dashboard, Access management, User groups, Users, Roles, Policies, Identity providers, Account settings, Access reports, Access analyzer, Archive rules, Analyzers, Settings, Credential report, Organization activity, and Service control policies (SCPs). The main content area is titled 'User groups (4) Info' and includes a description: 'A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.' Below this is a search bar and a table listing the user groups. The table has columns for Group name, Users, Permissions, and Creation time. The listed groups are Admin, EC2_Admin, Readonly, and S3_Admin, each with 1 user and 'Defined' permissions. The bottom of the page shows the footer with 'Feedback', 'English (US)', and copyright information.

Group name	Users	Permissions	Creation time
Admin	1	Defined	11 minutes ago
EC2_Admin	1	Defined	10 minutes ago
Readonly	1	Defined	8 minutes ago
S3_Admin	1	Defined	9 minutes ago

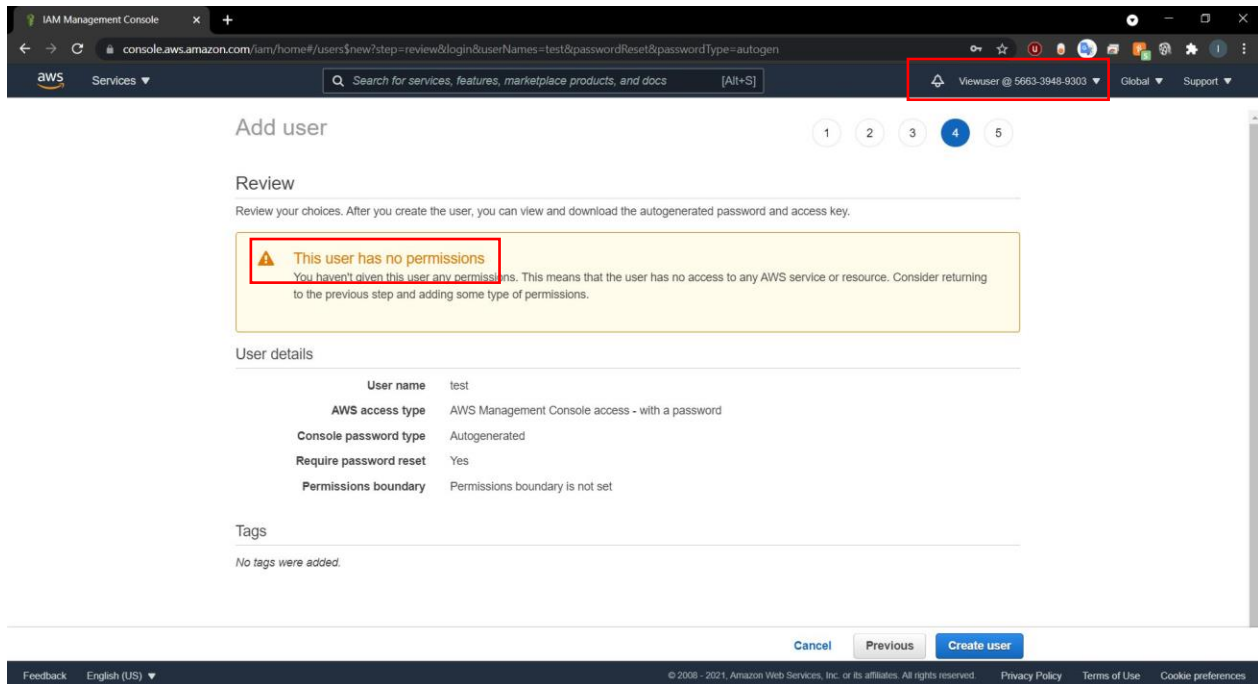
- Users



The screenshot shows the AWS IAM Management Console. The left sidebar contains the navigation menu with 'Users' highlighted under 'Access management'. The main content area displays the 'Users (4)' list. A blue banner at the top of the main area introduces the new Users list experience. Below the banner, there is a search bar and a table of users.

	User name	Groups	Last activity	MFA	Password age	Active key age
<input type="checkbox"/>	EC2admin	EC2_Admin	Never	None	1 minute ago	-
<input type="checkbox"/>	S3Admin	S3_Admin	Never	None	Now	-
<input type="checkbox"/>	Thanapon	Admin	15 minutes ago	None	105 days ago	21 minutes ago
<input type="checkbox"/>	Viewuser	Readonly	3 minutes ago	None	6 minutes ago	-

- Test log in to Viewuser (View only user) and try to create new user

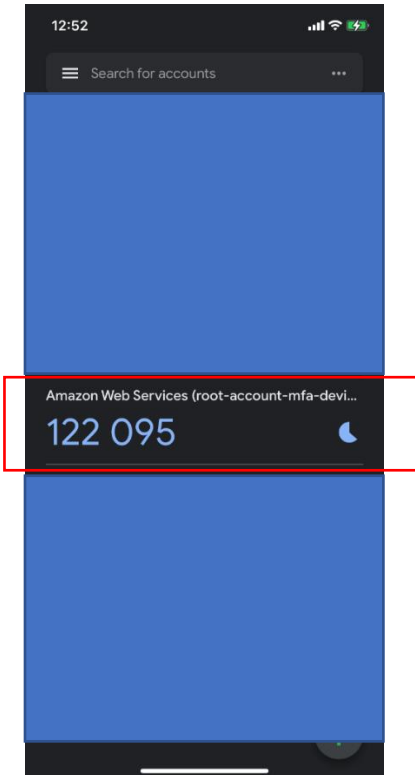
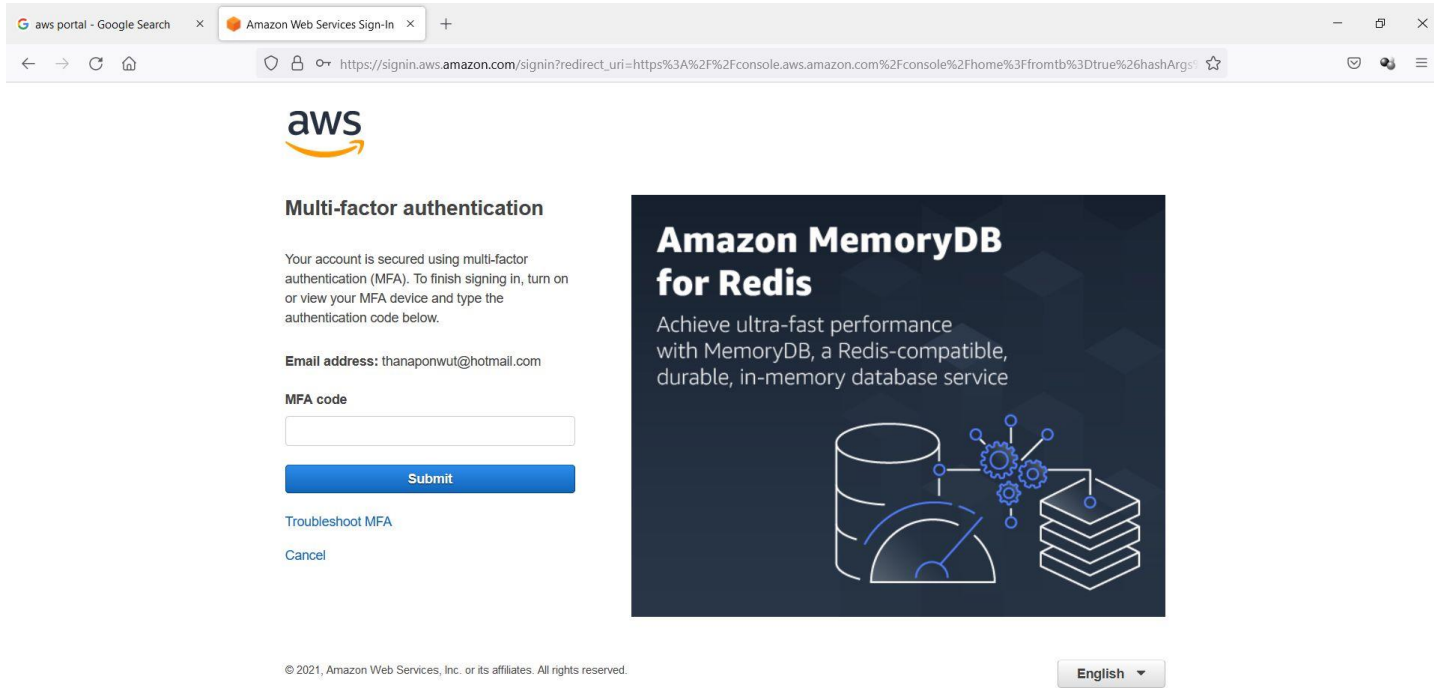


The screenshot shows the 'Add user' wizard in the AWS IAM Management Console, specifically the 'Review' step. A red box highlights a warning message: 'This user has no permissions. You haven't given this user any permissions. This means that the user has no access to any AWS service or resource. Consider returning to the previous step and adding some type of permissions.' Below the warning, the 'User details' section shows the following information:

User name	test
AWS access type	AWS Management Console access - with a password
Console password type	Autogenerated
Require password reset	Yes
Permissions boundary	Permissions boundary is not set

The 'Tags' section shows 'No tags were added.' At the bottom of the wizard, there are three buttons: 'Cancel', 'Previous', and 'Create user'.

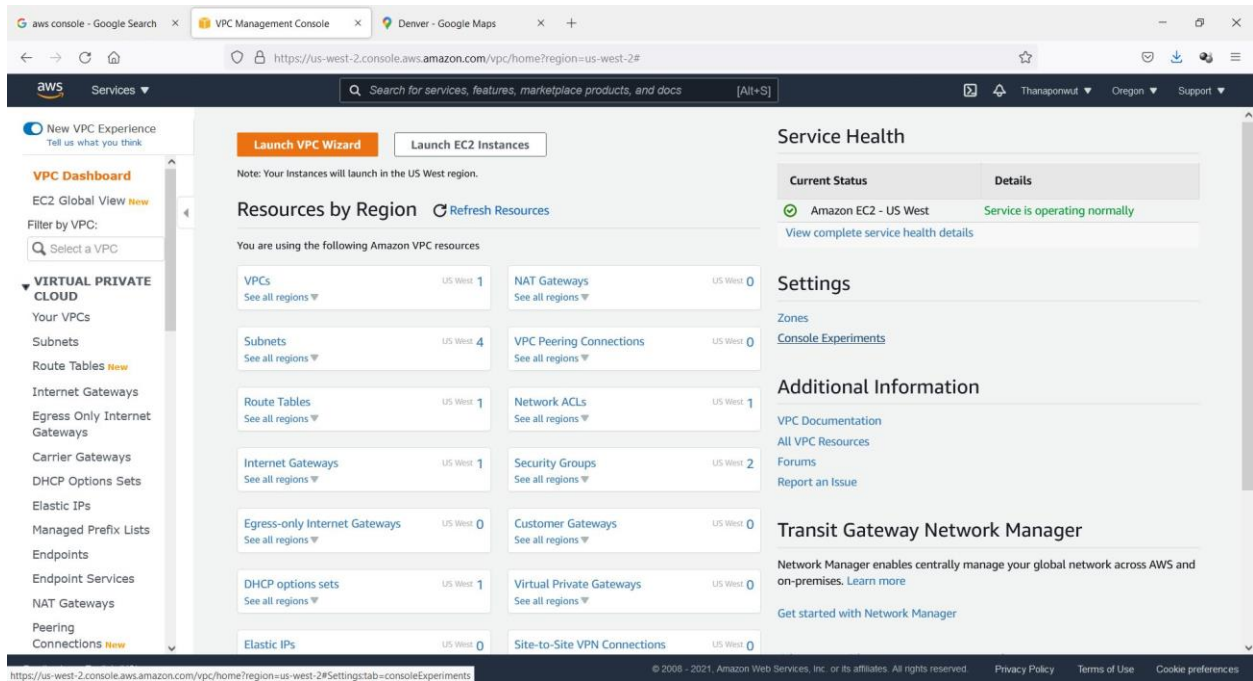
- Configure MFA for enhanced security



- Use US West (Oregon) region by defaults, there are 4 availability zones in the selected region.

The screenshot displays the AWS Management Console interface. The left-hand navigation pane includes links to 'VPC Dashboard', 'EC2 Global View', and 'VIRTUAL PRIVATE CLOUD'. The main content area is titled 'All Wavelength Zones for AWS Region, US West (Oregon). Select a different AWS region to manage Wavelength zones there.' It features two tabs: 'US West (Verizon) / us-west-2-wl1' (marked 'Opted out') and 'US West (Oregon) / us-west-2' (marked 'Opted in by default'). Under the 'Wavelength Zones' section, five zones are listed: us-west-2-wl1-den-wlz-1 (Denver), us-west-2-wl1-las-wlz-1 (Las Vegas), us-west-2-wl1-phx-wlz-1 (Phoenix), us-west-2-wl1-sea-wlz-1 (Seattle), and us-west-2-wl1-sfo-wlz-1 (San Francisco Bay Area). Below this, the 'Availability Zones' section is shown, listing four zones: us-west-2a (az1), us-west-2b (az2), us-west-2c (az3), and us-west-2d (az4). On the right side, a sidebar displays a list of AWS regions, with 'US West (Oregon)' highlighted in orange.

- VPC Dashboard



The screenshot shows the AWS VPC Dashboard in the US West region. The left sidebar contains navigation links for VPC Dashboard, EC2 Global View, and various VPC resources. The main content area displays 'Resources by Region' with a grid of resource counts for US West. The right sidebar includes 'Service Health' (showing Amazon EC2 - US West as 'Service is operating normally'), 'Settings' (Zones, Console Experiments), and 'Additional Information' (Documentation, Resources, Forums, Report an Issue). A 'Transit Gateway Network Manager' section is also present.

Resources by Region Refresh Resources

You are using the following Amazon VPC resources

Resource	US West
VPCs	1
NAT Gateways	0
Subnets	4
VPC Peering Connections	0
Route Tables	1
Network ACLs	1
Internet Gateways	1
Security Groups	2
Egress-only Internet Gateways	0
Customer Gateways	0
DHCP options sets	1
Virtual Private Gateways	0
Elastic IPs	0
Site-to-Site VPN Connections	0

Service Health

Current Status	Details
Amazon EC2 - US West	Service is operating normally

Settings

Zones
[Console Experiments](#)

Additional Information

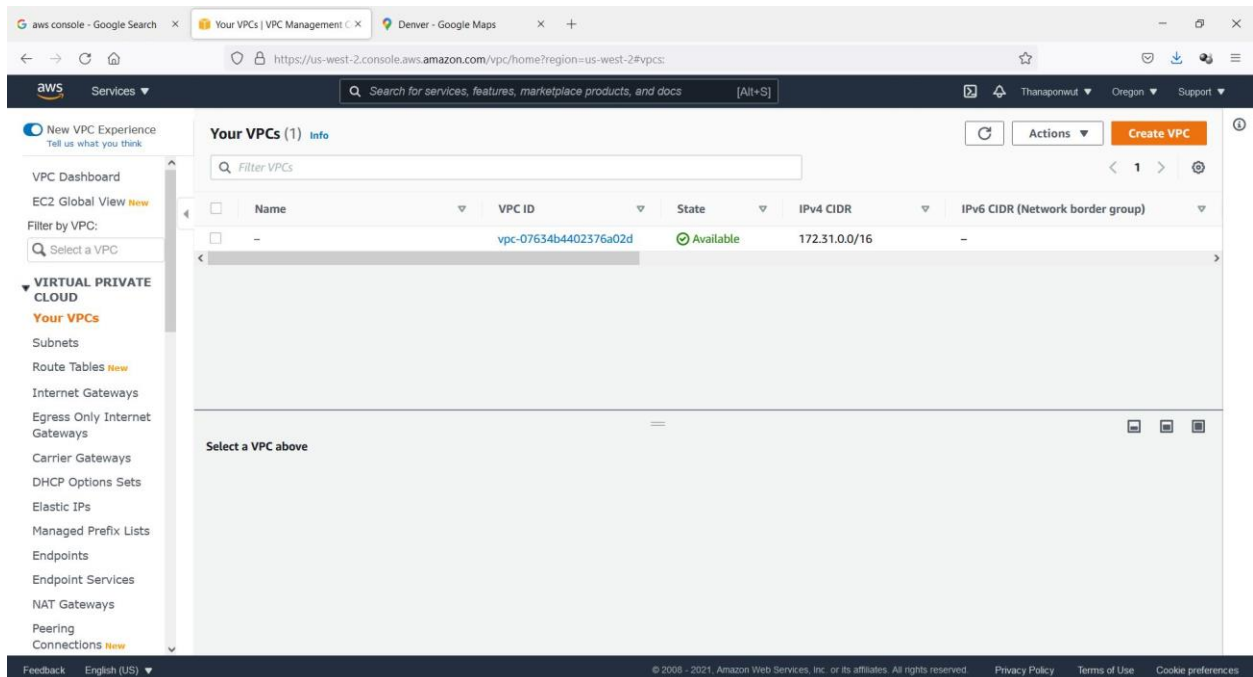
[VPC Documentation](#)
[All VPC Resources](#)
[Forums](#)
[Report an Issue](#)

Transit Gateway Network Manager

Network Manager enables centrally manage your global network across AWS and on-premises. [Learn more](#)

[Get started with Network Manager](#)

- Your VPCs



The screenshot shows the 'Your VPCs' page in the AWS console. The left sidebar is the same as the previous screenshot. The main content area shows a table with one VPC listed. The table has columns for Name, VPC ID, State, IPv4 CIDR, and IPv6 CIDR. Below the table, there is a section titled 'Select a VPC above' with three icons.

Your VPCs (1) info

Filter VPCs

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR (Network border group)
-	vpc-07634b4402376a02d	Available	172.31.0.0/16	-

Select a VPC above

- VPC details

The screenshot shows the AWS Management Console interface for a VPC. The left sidebar contains navigation links for VPC Dashboard, EC2 Global View, and various VPC services. The main content area displays the details for VPC **vpc-07634b4402376a02d**. The details are organized into a grid with the following information:

Details	
VPC ID vpc-07634b4402376a02d	State Available
Tenancy Default	DNS hostnames Enabled
Default VPC Yes	DNS resolution Enabled
Route 53 Resolver DNS Firewall rule groups -	Main route table rtb-01e992eae5a2db456
DHCP options set dopt-0ed517d83fb397711	Main network ACL acl-0d1def3383d42ad0b
IPv4 CIDR 172.31.0.0/16	IPv6 pool -
Owner ID 566339489303	IPv6 CIDR (Network border group) -

Below the details grid, there are tabs for **CIDRs**, **Flow logs**, and **Tags**. The **IPv4 CIDRs** tab is currently selected, showing a list of subnets.

- VPC subnets

The screenshot shows the AWS Management Console interface for VPC subnets. The left sidebar contains navigation links for VPC Dashboard, EC2 Global View, and various VPC services. The main content area displays the details for VPC **vpc-07634b4402376a02d**. The subnets are listed in a table with the following information:

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
-	subnet-05bb8cf9573bdb679	Available	vpc-07634b4402376a02d	172.31.0.0/20	-
-	subnet-0fcf18c356612d3fe	Available	vpc-07634b4402376a02d	172.31.48.0/20	-
-	subnet-0cfb1fb762f35c576	Available	vpc-07634b4402376a02d	172.31.16.0/20	-
-	subnet-0a75febdcaf88cf7	Available	vpc-07634b4402376a02d	172.31.32.0/20	-

Below the table, there is a section titled **Select a subnet** with a search bar and a list of subnets.

- Routing table
 - o IP in 172.31.0.0/16 is local IP.
 - o All other IP are routed to internet gateway.

The screenshot shows the AWS Management Console interface for a VPC. The left sidebar contains navigation links for VPC Dashboard, EC2 Global View, and various VPC resources. The main content area displays the 'Details' for a specific routing table (rtb-01e992eae5a2db456). The 'Routes' tab is active, showing a table with two routes. A red rectangular box highlights the first two rows of the routes table.

Destination	Target	Status	Propagated
172.31.0.0/16	local	Active	No
0.0.0.0/0	igw-0181c5839f8d111de	Active	No

- Internet gateways: virtual router that connect VPC to the internet.

The screenshot shows the AWS Management Console interface for Internet Gateways. The left sidebar contains navigation links for VPC Dashboard, EC2 Global View, and various VPC resources. The main content area displays the 'Internet gateways (1/1)' list. Below the list, the 'Details' tab for the selected gateway (igw-0181c5839f8d111de) is shown.

Name	Internet gateway ID	State	VPC ID	Owner
-	igw-0181c5839f8d111de	Attached	vpc-07634b4402376a02d	566339489303

Internet gateway ID	State	VPC ID	Owner
igw-0181c5839f8d111de	Attached	vpc-07634b4402376a02d	566339489303

Network ACLs: Inbound and Outbound rules for all subnets in a VPC (default value is allow all).

The screenshot shows the AWS VPC Management Console for a specific Network ACL. The left sidebar contains navigation links for various VPC services, with 'Network ACLs' highlighted under the 'SECURITY' section. The main content area displays the 'Details' tab for a Network ACL with ID 'acl-0d1def3383d42ad0b'. Key information includes: Associated with 4 Subnets, Default status is Yes, and VPC ID is 'vpc-07634b4402376a02d'. Below the details, there are tabs for 'Inbound rules', 'Outbound rules', 'Subnet associations', and 'Tags'. The 'Inbound rules' tab is active, showing a list of 2 rules. A message at the top of the rules section states: 'You can now check network connectivity with Reachability Analyzer' with a 'Run Reachability Analyzer' button. The rules table is as follows:

Rule number	Type	Protocol	Port range	Source	Allow/Deny
100	All traffic	All	All	0.0.0.0/0	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

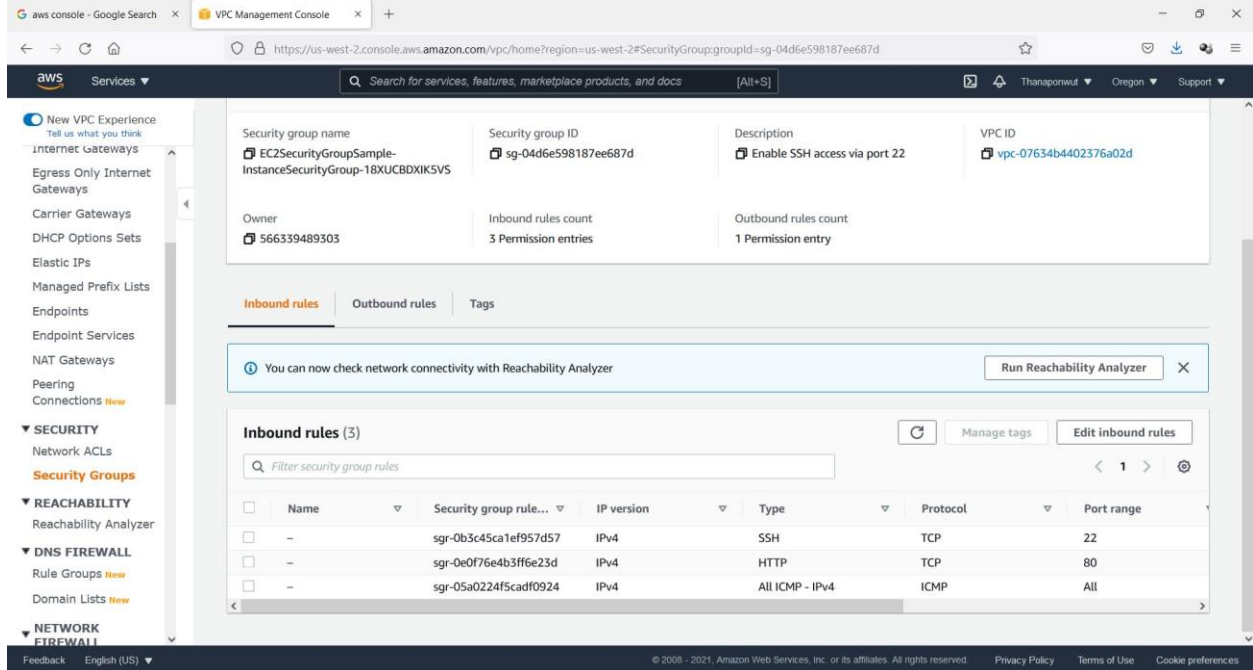
- Security groups: Inbound and Outbound rules for instances in VPC.

The screenshot shows the AWS VPC Management Console for Security Groups. The left sidebar is the same as the previous screenshot, with 'Security Groups' highlighted under the 'SECURITY' section. The main content area displays the 'Security Groups (1/2)' page. At the top, there are buttons for 'Actions', 'Export security groups to CSV', and 'Create security group'. A search bar is present. Below is a table listing the security groups:

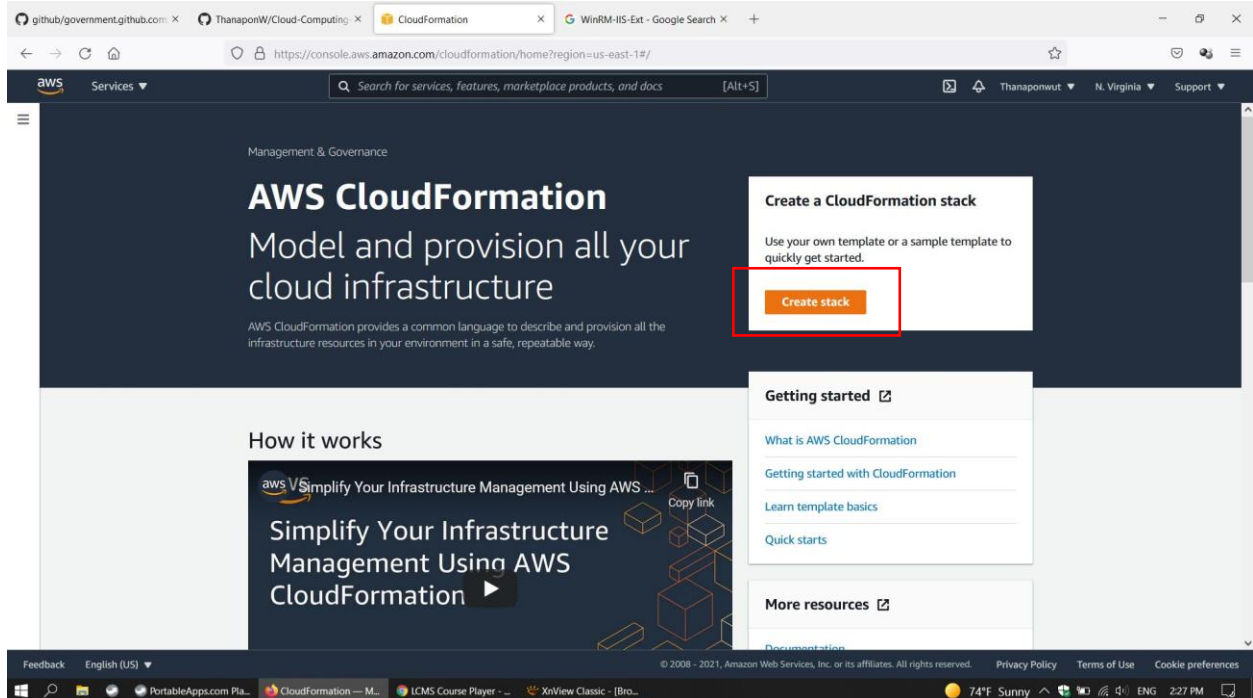
	Name	Security group ID	Security group name	VPC ID	Description	Owner
<input checked="" type="checkbox"/>	-	sg-04d6e598187ee687d	EC2SecurityGroupSample-InstanceSecurityGroup-18XUCBDXIK5VS	vpc-07634b4402376a02d	Enable SSH access via ...	566339489303
<input type="checkbox"/>	-	sg-061639ee12f646e98	default	vpc-07634b4402376a02d	default VPC security gr...	566339489303

Below the table, the 'Details' tab for the selected security group 'sg-04d6e598187ee687d - EC2SecurityGroupSample-InstanceSecurityGroup-18XUCBDXIK5VS' is shown. It includes tabs for 'Details', 'Inbound rules', 'Outbound rules', and 'Tags'. A message at the top of the rules section states: 'You can now check network connectivity with Reachability Analyzer' with a 'Run Reachability Analyzer' button. The 'Details' section is currently empty.

- Configure inbound rules for new instance.
 - o HTTP for web services access.
 - o ICMP for PING command.
 - o SSH for remote access from my IP only. For security reason.



- Use CloudFormation template to create the VM.



- Use EC2Instance template from S3 bucket.

The screenshot shows the 'Create stack' wizard in the AWS Management Console. The left sidebar indicates the current step is 'Step 1: Specify template'. The main content area is titled 'Create stack' and contains two sections: 'Prerequisite - Prepare template' and 'Specify template'. In the 'Prerequisite' section, the 'Template is ready' radio button is selected. In the 'Specify template' section, the 'Amazon S3 URL' radio button is selected. Below this, the 'Amazon S3 URL' field contains the path: `https://s3.us-west-2.amazonaws.com/cloudformation-templates-us-west-2/EC2InstanceWithSecurityGroupSample.template`. The 'S3 URL' field also contains the same path. A 'View in Designer' button is visible next to the S3 URL. At the bottom right, there are 'Cancel' and 'Next' buttons.

- Select instance type, and key name.

The screenshot shows the 'Specify stack details' wizard in the AWS Management Console. The left sidebar indicates the current step is 'Step 2: Specify stack details'. The main content area is titled 'Specify stack details' and contains two sections: 'Stack name' and 'Parameters'. In the 'Stack name' section, the 'Stack name' field contains the text: `EC2SecurityGroupSample`. In the 'Parameters' section, the 'InstanceType' dropdown menu is set to 't2.micro', the 'KeyName' dropdown menu is set to 'My_key', and the 'SSHLocation' field contains the IP address range: `0.0.0.0/0`. At the bottom right, there are 'Cancel', 'Previous', and 'Next' buttons.

- Instance will be created from CloudFormation template.

The screenshot shows the AWS Management Console interface for a CloudFormation stack named 'EC2SecurityGroupSample'. The 'Events' tab is selected, displaying a list of 6 events. The stack is in the 'CREATE_IN_PROGRESS' state. The left sidebar shows the 'Stacks (1)' section with the stack name and a 'CREATE_IN_PROGRESS' status. The main content area shows the 'Events' tab with a table of events.

Timestamp	Logical ID	Status	Status reason
2021-11-09 14:49:21 UTC-0600	EC2Instance	CREATE_IN_PROGRESS	Resource creation Initiated
2021-11-09 14:49:19 UTC-0600	EC2Instance	CREATE_IN_PROGRESS	-
2021-11-09 14:49:17 UTC-0600	InstanceSecurityGroup	CREATE_COMPLETE	-
2021-11-09 14:49:16 UTC-0600	InstanceSecurityGroup	CREATE_IN_PROGRESS	Resource creation Initiated
2021-11-09 14:49:10 UTC-0600	InstanceSecurityGroup	CREATE_IN_PROGRESS	-
2021-11-09 14:49:05 UTC-0600	EC2SecurityGroupSample	CREATE_IN_PROGRESS	User Initiated

- Instance summary

The screenshot shows the AWS Management Console interface for an EC2 instance named 'i-0ece23cc01c0163a0'. The 'Instance summary' tab is selected, displaying various details about the instance. The instance is in the 'Running' state. The left sidebar shows the 'Instances' section with the instance name and a 'Running' status. The main content area shows the 'Instance summary' tab with a table of instance details.

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0ece23cc01c0163a0	34.217.96.78 open address	172.31.18.49

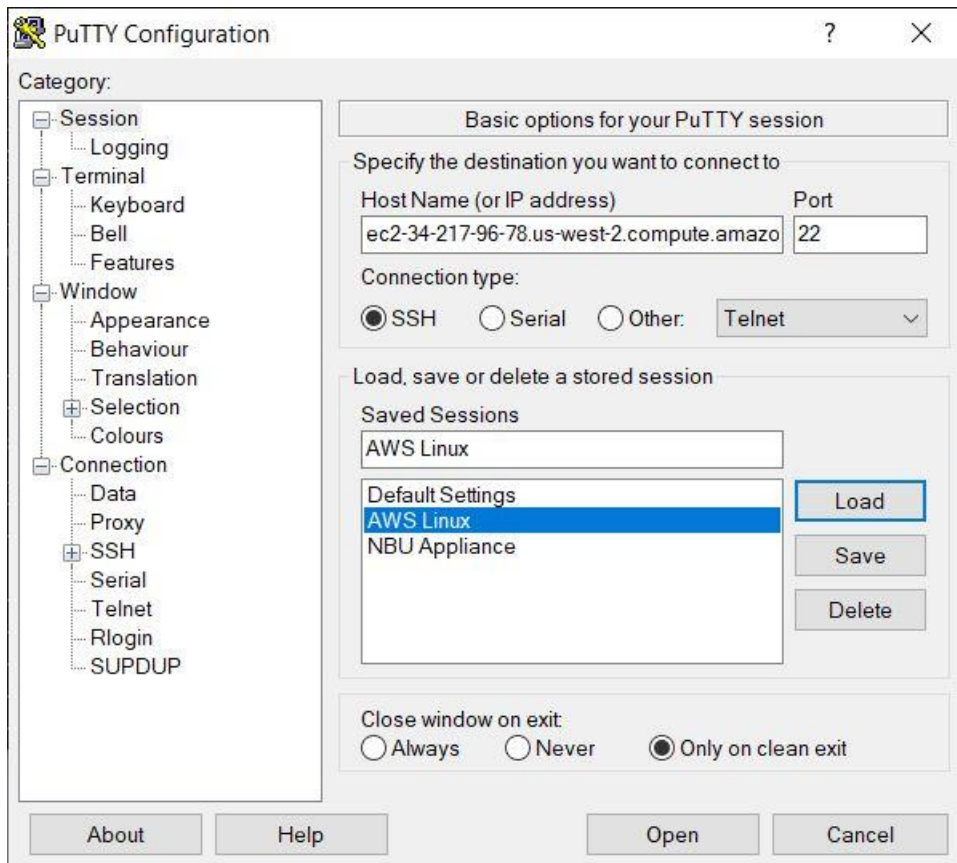
Instance state	Public IPv4 DNS
Running	ec2-34-217-96-78.us-west-2.compute.amazonaws.com open address

Private IPv4 DNS	Instance type	Elastic IP addresses
ip-172-31-18-49.us-west-2.compute.internal	t2.micro	-

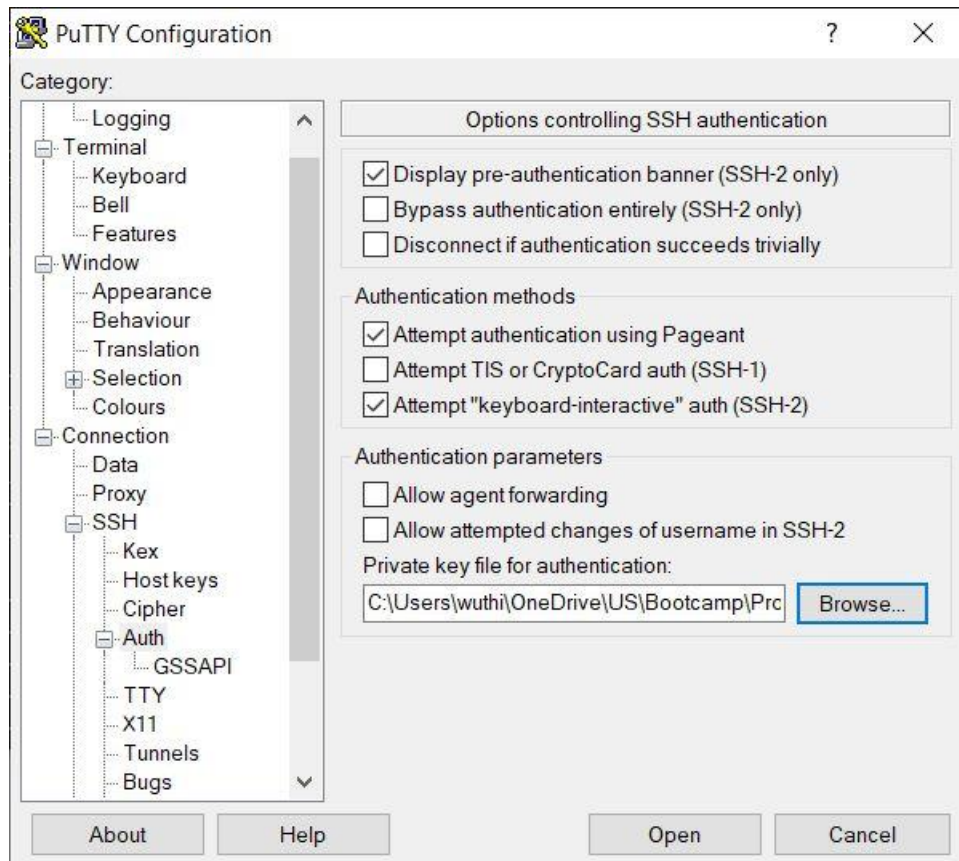
VPC ID	AWS Compute Optimizer finding	IAM Role
vpc-07634b4402376a02d	Opt-in to AWS Compute Optimizer for recommendations. Learn more	-

Subnet ID	Platform	Monitoring
subnet-0cfb1fb762f35c576	Amazon Linux (Info)	disabled

- Use Putty to connect to an instance by using Public DNS name of an instance.



- Select private key to connect then click open.



- Login to AWS Linux instance, update all packages, install web service, and grant permission to an ec2-user.

login as: ec2-user

Authenticating with public key "My_key"

```
__| __|_ )
```

```
_| ( / Amazon Linux 2 AMI
```

```
__|\__|__|
```

<https://aws.amazon.com/amazon-linux-2/>

1 package(s) needed for security, out of 14 available

Run "sudo yum update" to apply all updates.

```
[ec2-user@ip-172-31-18-49 ~]$ sudo yum update
```

Loaded plugins: extras_suggestions, langpacks, priorities, update-motd

Resolving Dependencies

--> Running transaction check

---> Package aws-cfn-bootstrap.noarch 0:2.0-6.amzn2 will be updated

---> Package aws-cfn-bootstrap.noarch 0:2.0-9.amzn2 will be an update

---> Package dracut-config-ec2.noarch 0:2.0-1.amzn2 will be updated

---> Package dracut-config-ec2.noarch 0:2.0-2.amzn2 will be an update

---> Package ec2-instance-connect.noarch 0:1.1-14.amzn2 will be updated

---> Package ec2-instance-connect.noarch 0:1.1-15.amzn2 will be an update

---> Package glibc.x86_64 0:2.26-55.amzn2 will be updated

---> Package glibc.x86_64 0:2.26-56.amzn2 will be an update

---> Package glibc-all-langpacks.x86_64 0:2.26-55.amzn2 will be updated

---> Package glibc-all-langpacks.x86_64 0:2.26-56.amzn2 will be an update

---> Package glibc-common.x86_64 0:2.26-55.amzn2 will be updated

---> Package glibc-common.x86_64 0:2.26-56.amzn2 will be an update

---> Package glibc-locale-source.x86_64 0:2.26-55.amzn2 will be updated

---> Package glibc-locale-source.x86_64 0:2.26-56.amzn2 will be an update

---> Package glibc-minimal-langpack.x86_64 0:2.26-55.amzn2 will be updated

---> Package glibc-minimal-langpack.x86_64 0:2.26-56.amzn2 will be an update
---> Package kernel.x86_64 0:4.14.252-195.483.amzn2 will be installed
---> Package kpatch-runtime.noarch 0:0.9.2-4.amzn2 will be updated
---> Package kpatch-runtime.noarch 0:0.9.4-2.amzn2 will be an update
---> Package libcrypt.x86_64 0:2.26-55.amzn2 will be updated
---> Package libcrypt.x86_64 0:2.26-56.amzn2 will be an update
---> Package openssl.x86_64 1:1.0.2k-19.amzn2.0.8 will be updated
---> Package openssl.x86_64 1:1.0.2k-19.amzn2.0.10 will be an update
---> Package openssl-libs.x86_64 1:1.0.2k-19.amzn2.0.8 will be updated
---> Package openssl-libs.x86_64 1:1.0.2k-19.amzn2.0.10 will be an update
---> Package system-release.x86_64 1:2-13.amzn2 will be updated
---> Package system-release.x86_64 1:2-14.amzn2 will be an update
--> Finished Dependency Resolution

Dependencies Resolved

=====				
Package	Arch	Version	Repository	Size
=====				

Installing:

kernel	x86_64	4.14.252-195.483.amzn2	amzn2-core	21 M
--------	--------	------------------------	------------	------

Updating:

aws-cfn-bootstrap	noarch	2.0-9.amzn2	amzn2-core	762 k
dracut-config-ec2	noarch	2.0-2.amzn2	amzn2-core	4.9 k
ec2-instance-connect	noarch	1.1-15.amzn2	amzn2-core	23 k
glibc	x86_64	2.26-56.amzn2	amzn2-core	3.3 M
glibc-all-langpacks	x86_64	2.26-56.amzn2	amzn2-core	7.0 M
glibc-common	x86_64	2.26-56.amzn2	amzn2-core	772 k
glibc-locale-source	x86_64	2.26-56.amzn2	amzn2-core	3.2 M

glibc-minimal-langpack	x86_64	2.26-56.amzn2	amzn2-core	32 k
kpatch-runtime	noarch	0.9.4-2.amzn2	amzn2-core	27 k
libcrypt	x86_64	2.26-56.amzn2	amzn2-core	52 k
openssl	x86_64	1:1.0.2k-19.amzn2.0.10	amzn2-core	495 k
openssl-libs	x86_64	1:1.0.2k-19.amzn2.0.10	amzn2-core	1.2 M
system-release	x86_64	1:2-14.amzn2	amzn2-core	18 k

Transaction Summary

=====

Install 1 Package

Upgrade 13 Packages

Total download size: 38 M

Is this ok [y/d/N]: Y

Downloading packages:

Delta RPMs disabled because /usr/bin/applydeltarpm not installed.

(1/14): dracut-config-ec2-2.0-2.amzn2.noarch.rpm	4.9 kB	00:00
(2/14): ec2-instance-connect-1.1-15.amzn2.noarch.rpm	23 kB	00:00
(3/14): aws-cfn-bootstrap-2.0-9.amzn2.noarch.rpm	762 kB	00:00
(4/14): glibc-2.26-56.amzn2.x86_64.rpm	3.3 MB	00:00
(5/14): glibc-common-2.26-56.amzn2.x86_64.rpm	772 kB	00:00
(6/14): glibc-all-langpacks-2.26-56.amzn2.x86_64.rpm	7.0 MB	00:00
(7/14): glibc-locale-source-2.26-56.amzn2.x86_64.rpm	3.2 MB	00:00
(8/14): glibc-minimal-langpack-2.26-56.amzn2.x86_64.rpm	32 kB	00:00
(9/14): kpatch-runtime-0.9.4-2.amzn2.noarch.rpm	27 kB	00:00
(10/14): libcrypt-2.26-56.amzn2.x86_64.rpm	52 kB	00:00
(11/14): openssl-1.0.2k-19.amzn2.0.10.x86_64.rpm	495 kB	00:00
(12/14): openssl-libs-1.0.2k-19.amzn2.0.10.x86_64.rpm	1.2 MB	00:00
(13/14): system-release-2-14.amzn2.x86_64.rpm	18 kB	00:00

(14/14): kernel-4.14.252-195.483.amzn2.x86_64.rpm | 21 MB 00:00

Total 39 MB/s | 38 MB 00:00

Running transaction check

Running transaction test

Transaction test succeeded

Running transaction

Updating : glibc-common-2.26-56.amzn2.x86_64 1/27

Updating : glibc-2.26-56.amzn2.x86_64 2/27

Updating : glibc-minimal-langpack-2.26-56.amzn2.x86_64 3/27

Updating : 1:openssl-libs-1.0.2k-19.amzn2.0.10.x86_64 4/27

Updating : 1:openssl-1.0.2k-19.amzn2.0.10.x86_64 5/27

ec2-instance-connect:x:997:995::/home/ec2-instance-connect:/sbin/nologin

Updating : ec2-instance-connect-1.1-15.amzn2.noarch 6/27

Created symlink from /etc/systemd/system/multi-user.target.wants/ec2-instance-connect-harvest-hostkeys.service to /usr/lib/systemd/system/ec2-instance-connect-harvest-hostkeys.service.

Updating : libcrypt-2.26-56.amzn2.x86_64 7/27

Updating : glibc-all-langpacks-2.26-56.amzn2.x86_64 8/27

Updating : glibc-locale-source-2.26-56.amzn2.x86_64 9/27

Updating : aws-cfn-bootstrap-2.0-9.amzn2.noarch 10/27

Updating : kpatch-runtime-0.9.4-2.amzn2.noarch 11/27

Updating : dracut-config-ec2-2.0-2.amzn2.noarch 12/27

Updating : 1:system-release-2-14.amzn2.x86_64 13/27

Installing : kernel-4.14.252-195.483.amzn2.x86_64 14/27

Cleanup : glibc-locale-source-2.26-55.amzn2.x86_64 15/27

Cleanup : glibc-all-langpacks-2.26-55.amzn2.x86_64 16/27

Cleanup : ec2-instance-connect-1.1-14.amzn2.noarch 17/27

Cleanup : 1:openssl-1.0.2k-19.amzn2.0.8.x86_64 18/27

Cleanup : 1:openssl-libs-1.0.2k-19.amzn2.0.8.x86_64 19/27

Cleanup	: libcrypt-2.26-55.amzn2.x86_64	20/27
Cleanup	: aws-cfn-bootstrap-2.0-6.amzn2.noarch	21/27
Cleanup	: kpatch-runtime-0.9.2-4.amzn2.noarch	22/27
Cleanup	: dracut-config-ec2-2.0-1.amzn2.noarch	23/27
Cleanup	: 1:system-release-2-13.amzn2.x86_64	24/27
Cleanup	: glibc-common-2.26-55.amzn2.x86_64	25/27
Cleanup	: glibc-minimal-langpack-2.26-55.amzn2.x86_64	26/27
Cleanup	: glibc-2.26-55.amzn2.x86_64	27/27
Verifying	: kernel-4.14.252-195.483.amzn2.x86_64	1/27
Verifying	: libcrypt-2.26-56.amzn2.x86_64	2/27
Verifying	: 1:openssl-1.0.2k-19.amzn2.0.10.x86_64	3/27
Verifying	: 1:system-release-2-14.amzn2.x86_64	4/27
Verifying	: 1:openssl-libs-1.0.2k-19.amzn2.0.10.x86_64	5/27
Verifying	: dracut-config-ec2-2.0-2.amzn2.noarch	6/27
Verifying	: glibc-minimal-langpack-2.26-56.amzn2.x86_64	7/27
Verifying	: glibc-2.26-56.amzn2.x86_64	8/27
Verifying	: ec2-instance-connect-1.1-15.amzn2.noarch	9/27
Verifying	: kpatch-runtime-0.9.4-2.amzn2.noarch	10/27
Verifying	: glibc-all-langpacks-2.26-56.amzn2.x86_64	11/27
Verifying	: aws-cfn-bootstrap-2.0-9.amzn2.noarch	12/27
Verifying	: glibc-locale-source-2.26-56.amzn2.x86_64	13/27
Verifying	: glibc-common-2.26-56.amzn2.x86_64	14/27
Verifying	: 1:system-release-2-13.amzn2.x86_64	15/27
Verifying	: glibc-2.26-55.amzn2.x86_64	16/27
Verifying	: glibc-common-2.26-55.amzn2.x86_64	17/27
Verifying	: aws-cfn-bootstrap-2.0-6.amzn2.noarch	18/27
Verifying	: glibc-all-langpacks-2.26-55.amzn2.x86_64	19/27
Verifying	: ec2-instance-connect-1.1-14.amzn2.noarch	20/27
Verifying	: dracut-config-ec2-2.0-1.amzn2.noarch	21/27

Verifying : glibc-minimal-langpack-2.26-55.amzn2.x86_64 22/27
Verifying : 1:openssl-libs-1.0.2k-19.amzn2.0.8.x86_64 23/27
Verifying : kpatch-runtime-0.9.2-4.amzn2.noarch 24/27
Verifying : 1:openssl-1.0.2k-19.amzn2.0.8.x86_64 25/27
Verifying : glibc-locale-source-2.26-55.amzn2.x86_64 26/27
Verifying : libcrypt-2.26-55.amzn2.x86_64 27/27

Installed:

kernel.x86_64 0:4.14.252-195.483.amzn2

Updated:

aws-cfn-bootstrap.noarch 0:2.0-9.amzn2
dracut-config-ec2.noarch 0:2.0-2.amzn2
ec2-instance-connect.noarch 0:1.1-15.amzn2
glibc.x86_64 0:2.26-56.amzn2
glibc-all-langpacks.x86_64 0:2.26-56.amzn2
glibc-common.x86_64 0:2.26-56.amzn2
glibc-locale-source.x86_64 0:2.26-56.amzn2
glibc-minimal-langpack.x86_64 0:2.26-56.amzn2
kpatch-runtime.noarch 0:0.9.4-2.amzn2
libcrypt.x86_64 0:2.26-56.amzn2
openssl.x86_64 1:1.0.2k-19.amzn2.0.10
openssl-libs.x86_64 1:1.0.2k-19.amzn2.0.10
system-release.x86_64 1:2-14.amzn2

Complete!

```
[ec2-user@ip-172-31-18-49 ~]$ sudo amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2
```

Installing php-pdo, php-mysqlnd, php-fpm, php-cli, php-json, mariadb

Loaded plugins: extras_suggestions, langpacks, priorities, update-motd

Cleaning repos: amzn2-core amzn2extra-docker amzn2extra-lamp-mariadb10.2-php7.2

: amzn2extra-php7.2

12 metadata files removed

4 sqlite files removed

0 metadata files removed

Loaded plugins: extras_suggestions, langpacks, priorities, update-motd

amzn2-core | 3.7 kB 00:00

amzn2extra-docker | 3.0 kB 00:00

amzn2extra-lamp-mariadb10.2-php7.2 | 3.0 kB 00:00

amzn2extra-php7.2 | 3.0 kB 00:00

(1/9): amzn2-core/2/x86_64/group_gz | 2.5 kB 00:00

(2/9): amzn2-core/2/x86_64/updateinfo | 423 kB 00:00

(3/9): amzn2extra-lamp-mariadb10.2-php7.2/2/x86_64/primary | 506 kB 00:00

(4/9): amzn2extra-php7.2/2/x86_64/updateinfo | 76 B 00:00

(5/9): amzn2extra-php7.2/2/x86_64/primary_db | 580 kB 00:00

(6/9): amzn2extra-docker/2/x86_64/updateinfo | 76 B 00:00

(7/9): amzn2extra-docker/2/x86_64/primary_db | 85 kB 00:00

(8/9): amzn2extra-lamp-mariadb10.2-php7.2/2/x86_64/updatei | 76 B 00:00

(9/9): amzn2-core/2/x86_64/primary_db | 58 MB 00:01

Resolving Dependencies

--> Running transaction check

---> Package mariadb.x86_64 3:10.2.38-1.amzn2.0.1 will be installed

--> Processing Dependency: mariadb-libs(x86-64) = 3:10.2.38-1.amzn2.0.1 for package: 3:mariadb-10.2.38-1.amzn2.0.1.x86_64

--> Processing Dependency: mariadb-common(x86-64) = 3:10.2.38-1.amzn2.0.1 for package: 3:mariadb-10.2.38-1.amzn2.0.1.x86_64

---> Package php-cli.x86_64 0:7.2.34-1.amzn2 will be installed

--> Processing Dependency: php-common(x86-64) = 7.2.34-1.amzn2 for package: php-cli-7.2.34-1.amzn2.x86_64

---> Package php-fpm.x86_64 0:7.2.34-1.amzn2 will be installed

```

---> Package php-json.x86_64 0:7.2.34-1.amzn2 will be installed
---> Package php-mysqlnd.x86_64 0:7.2.34-1.amzn2 will be installed
---> Package php-pdo.x86_64 0:7.2.34-1.amzn2 will be installed
--> Running transaction check
---> Package mariadb-common.x86_64 3:10.2.38-1.amzn2.0.1 will be installed
--> Processing Dependency: /etc/my.cnf for package: 3:mariadb-common-10.2.38-1.amzn2.0.1.x86_64
---> Package mariadb-libs.x86_64 1:5.5.68-1.amzn2 will be updated
---> Package mariadb-libs.x86_64 3:10.2.38-1.amzn2.0.1 will be an update
---> Package php-common.x86_64 0:7.2.34-1.amzn2 will be installed
--> Processing Dependency: libzip.so.5()(64bit) for package: php-common-7.2.34-1.amzn2.x86_64
--> Running transaction check
---> Package libzip.x86_64 0:1.3.2-1.amzn2.0.1 will be installed
---> Package mariadb-config.x86_64 3:10.2.38-1.amzn2.0.1 will be installed
---> Package mariadb-libs.x86_64 1:5.5.68-1.amzn2 will be updated
--> Finished Dependency Resolution

```

Dependencies Resolved

```

=====
Package   Arch Version      Repository      Size
=====

```

Installing:

```

mariadb   x86_64 3:10.2.38-1.amzn2.0.1
          amzn2extra-lamp-mariadb10.2-php7.2 6.1 M
php-cli   x86_64 7.2.34-1.amzn2  amzn2extra-php7.2      4.4 M
php-fpm   x86_64 7.2.34-1.amzn2  amzn2extra-php7.2      1.5 M
php-json   x86_64 7.2.34-1.amzn2  amzn2extra-php7.2      71 k
php-mysqlnd x86_64 7.2.34-1.amzn2  amzn2extra-php7.2      238 k
php-pdo    x86_64 7.2.34-1.amzn2  amzn2extra-php7.2      132 k

```

Installing for dependencies:

libzip	x86_64 1.3.2-1.amzn2.0.1	amzn2-core	62 k
mariadb-common			
	x86_64 3:10.2.38-1.amzn2.0.1		
		amzn2extra-lamp-mariadb10.2-php7.2	58 k
mariadb-config			
	x86_64 3:10.2.38-1.amzn2.0.1		
		amzn2extra-lamp-mariadb10.2-php7.2	34 k
php-common	x86_64 7.2.34-1.amzn2	amzn2extra-php7.2	1.1 M

Updating for dependencies:

mariadb-libs	x86_64 3:10.2.38-1.amzn2.0.1		
		amzn2extra-lamp-mariadb10.2-php7.2	154 k

Transaction Summary

=====

Install 6 Packages (+4 Dependent packages)
Upgrade (1 Dependent package)

Total download size: 14 M

Downloading packages:

Delta RPMs disabled because /usr/bin/applydeltarpm not installed.

(1/11): libzip-1.3.2-1.amzn2.0.1.x86_64.rpm	62 kB	00:00
(2/11): mariadb-common-10.2.38-1.amzn2.0.1.x86_64.rpm	58 kB	00:00
(3/11): mariadb-config-10.2.38-1.amzn2.0.1.x86_64.rpm	34 kB	00:00
(4/11): mariadb-libs-10.2.38-1.amzn2.0.1.x86_64.rpm	154 kB	00:00
(5/11): mariadb-10.2.38-1.amzn2.0.1.x86_64.rpm	6.1 MB	00:00
(6/11): php-common-7.2.34-1.amzn2.x86_64.rpm	1.1 MB	00:00
(7/11): php-cli-7.2.34-1.amzn2.x86_64.rpm	4.4 MB	00:00
(8/11): php-fpm-7.2.34-1.amzn2.x86_64.rpm	1.5 MB	00:00

(9/11): php-json-7.2.34-1.amzn2.x86_64.rpm | 71 kB 00:00
(10/11): php-mysqlnd-7.2.34-1.amzn2.x86_64.rpm | 238 kB 00:00
(11/11): php-pdo-7.2.34-1.amzn2.x86_64.rpm | 132 kB 00:00

Total 27 MB/s | 14 MB 00:00

Running transaction check

Running transaction test

Transaction test succeeded

Running transaction

Installing : libzip-1.3.2-1.amzn2.0.1.x86_64 1/12
Installing : php-common-7.2.34-1.amzn2.x86_64 2/12
Installing : php-json-7.2.34-1.amzn2.x86_64 3/12
Installing : php-pdo-7.2.34-1.amzn2.x86_64 4/12
Installing : 3:mariadb-config-10.2.38-1.amzn2.0.1.x86_64 5/12
Installing : 3:mariadb-common-10.2.38-1.amzn2.0.1.x86_64 6/12
Updating : 3:mariadb-libs-10.2.38-1.amzn2.0.1.x86_64 7/12
Installing : 3:mariadb-10.2.38-1.amzn2.0.1.x86_64 8/12
Installing : php-mysqlnd-7.2.34-1.amzn2.x86_64 9/12
Installing : php-fpm-7.2.34-1.amzn2.x86_64 10/12
Installing : php-cli-7.2.34-1.amzn2.x86_64 11/12
Cleanup : 1:mariadb-libs-5.5.68-1.amzn2.x86_64 12/12
Verifying : php-fpm-7.2.34-1.amzn2.x86_64 1/12
Verifying : php-cli-7.2.34-1.amzn2.x86_64 2/12
Verifying : 3:mariadb-common-10.2.38-1.amzn2.0.1.x86_64 3/12
Verifying : 3:mariadb-libs-10.2.38-1.amzn2.0.1.x86_64 4/12
Verifying : php-json-7.2.34-1.amzn2.x86_64 5/12
Verifying : 3:mariadb-config-10.2.38-1.amzn2.0.1.x86_64 6/12
Verifying : libzip-1.3.2-1.amzn2.0.1.x86_64 7/12
Verifying : php-mysqlnd-7.2.34-1.amzn2.x86_64 8/12

Verifying : php-pdo-7.2.34-1.amzn2.x86_64	9/12
Verifying : 3:mariadb-10.2.38-1.amzn2.0.1.x86_64	10/12
Verifying : php-common-7.2.34-1.amzn2.x86_64	11/12
Verifying : 1:mariadb-libs-5.5.68-1.amzn2.x86_64	12/12

Installed:

mariadb.x86_64 3:10.2.38-1.amzn2.0.1 php-cli.x86_64 0:7.2.34-1.amzn2
php-fpm.x86_64 0:7.2.34-1.amzn2 php-json.x86_64 0:7.2.34-1.amzn2
php-mysqlnd.x86_64 0:7.2.34-1.amzn2 php-pdo.x86_64 0:7.2.34-1.amzn2

Dependency Installed:

libzip.x86_64 0:1.3.2-1.amzn2.0.1
mariadb-common.x86_64 3:10.2.38-1.amzn2.0.1
mariadb-config.x86_64 3:10.2.38-1.amzn2.0.1
php-common.x86_64 0:7.2.34-1.amzn2

Dependency Updated:

mariadb-libs.x86_64 3:10.2.38-1.amzn2.0.1

Complete!

0 ansible2 available \

[=2.4.2 =2.4.6 =2.8 =stable]

2 httpd_modules available [=1.0 =stable]

3 memcached1.5 available \

[=1.5.1 =1.5.16 =1.5.17]

5 postgresql9.6 available \

[=9.6.6 =9.6.8 =stable]

6 postgresql10 available [=10 =stable]

9 R3.4 available [=3.4.3 =stable]

10 rust1 available \

[=1.22.1 =1.26.0 =1.26.1 =1.27.2 =1.31.0 =1.38.0

=stable]

11 vim available [=8.0 =stable]

15 php7.2=latest enabled \

[=7.2.0 =7.2.4 =7.2.5 =7.2.8 =7.2.11 =7.2.13 =7.2.14

=7.2.16 =7.2.17 =7.2.19 =7.2.21 =7.2.22 =7.2.23

=7.2.24 =7.2.26 =stable]

17 lamp-mariadb10.2-php7.2=latest enabled \

[=10.2.10_7.2.0 =10.2.10_7.2.4 =10.2.10_7.2.5

=10.2.10_7.2.8 =10.2.10_7.2.11 =10.2.10_7.2.13

=10.2.10_7.2.14 =10.2.10_7.2.16 =10.2.10_7.2.17

=10.2.10_7.2.19 =10.2.10_7.2.22 =10.2.10_7.2.23

=10.2.10_7.2.24 =stable]

18 libreoffice available \

[=5.0.6.2_15 =5.3.6.1 =stable]

19 gimp available [=2.8.22]

20 docker=latest enabled \

[=17.12.1 =18.03.1 =18.06.1 =18.09.9 =stable]

21 mate-desktop1.x available \

[=1.19.0 =1.20.0 =stable]

22 GraphicsMagick1.3 available \

[=1.3.29 =1.3.32 =1.3.34 =stable]

23 tomcat8.5 available \

[=8.5.31 =8.5.32 =8.5.38 =8.5.40 =8.5.42 =8.5.50

=stable]

24 epel available [=7.11 =stable]

25 testing available [=1.0 =stable]

26 ecs available [=stable]

27 corretto8 available \

 [=1.8.0_192 =1.8.0_202 =1.8.0_212 =1.8.0_222 =1.8.0_232

 =1.8.0_242 =stable]

28 firecracker available [=0.11 =stable]

29 golang1.11 available \

 [=1.11.3 =1.11.11 =1.11.13 =stable]

30 squid4 available [=4 =stable]

_ php7.3 available \

 [=7.3.2 =7.3.3 =7.3.4 =7.3.6 =7.3.8 =7.3.9 =7.3.10

 =7.3.11 =7.3.13 =stable]

32 lustre2.10 available \

 [=2.10.5 =2.10.8 =stable]

33 java-openjdk11 available [=11 =stable]

34 lynis available [=stable]

35 kernel-ng available [=stable]

36 BCC available [=0.x =stable]

37 mono available [=5.x =stable]

38 nginx1 available [=stable]

39 ruby2.6 available [=2.6 =stable]

40 mock available [=stable]

41 postgresql11 available [=11 =stable]

_ php7.4 available [=stable]

43 livepatch available [=stable]

44 python3.8 available [=stable]

45 haproxy2 available [=stable]

46 collectd available [=stable]

47 aws-nitro-enclaves-cli available [=stable]

48 R4 available [=stable]

49 kernel-5.4 available [=stable]

```
50 selinux-ng          available [ =stable ]
_ php8.0              available [ =stable ]
52 tomcat9            available [ =stable ]
53 unbound1.13        available [ =stable ]
_ mariadb10.5         available [ =stable ]
55 kernel-5.10        available [ =stable ]
56 redis6             available [ =stable ]
57 ruby3.0            available [ =stable ]
58 postgresql12       available [ =stable ]
59 postgresql13       available [ =stable ]
60 mock2              available [ =stable ]
61 dnsmasq2.85        available [ =stable ]
```

```
[ec2-user@ip-172-31-18-49 ~]$ cat /etc/system-release
```

```
Amazon Linux release 2 (Karoo)
```

```
[ec2-user@ip-172-31-18-49 ~]$ sudo yum install -y httpd
```

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
```

```
Resolving Dependencies
```

```
--> Running transaction check
```

```
---> Package httpd.x86_64 0:2.4.51-1.amzn2 will be installed
```

```
--> Processing Dependency: httpd-tools = 2.4.51-1.amzn2 for package: httpd-2.4.51-1.amzn2.x86_64
```

```
--> Processing Dependency: httpd-filesystem = 2.4.51-1.amzn2 for package: httpd-2.4.51-1.amzn2.x86_64
```

```
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.51-1.amzn2.x86_64
```

```
--> Processing Dependency: mod_http2 for package: httpd-2.4.51-1.amzn2.x86_64
```

```
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.51-1.amzn2.x86_64
```

```
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.51-1.amzn2.x86_64
```

```
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.51-1.amzn2.x86_64
```

```
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.51-1.amzn2.x86_64
```

```
--> Running transaction check
```

---> Package apr.x86_64 0:1.7.0-9.amzn2 will be installed
---> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package: apr-util-1.6.1-5.amzn2.0.2.x86_64
---> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
---> Package httpd-filesystem.noarch 0:2.4.51-1.amzn2 will be installed
---> Package httpd-tools.x86_64 0:2.4.51-1.amzn2 will be installed
---> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
---> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.1 will be installed
--> Running transaction check
---> Package apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

```
=====
Package      Arch   Version      Repository    Size
=====
```

Installing:

httpd	x86_64	2.4.51-1.amzn2	amzn2-core	1.3 M
-------	--------	----------------	------------	-------

Installing for dependencies:

apr	x86_64	1.7.0-9.amzn2	amzn2-core	122 k
apr-util	x86_64	1.6.1-5.amzn2.0.2	amzn2-core	99 k
apr-util-bdb	x86_64	1.6.1-5.amzn2.0.2	amzn2-core	19 k
generic-logos-httpd	noarch	18.0.0-4.amzn2	amzn2-core	19 k
httpd-filesystem	noarch	2.4.51-1.amzn2	amzn2-core	24 k
httpd-tools	x86_64	2.4.51-1.amzn2	amzn2-core	88 k
mailcap	noarch	2.1.41-2.amzn2	amzn2-core	31 k
mod_http2	x86_64	1.15.19-1.amzn2.0.1	amzn2-core	149 k

Transaction Summary

=====

Install 1 Package (+8 Dependent packages)

Total download size: 1.9 M

Installed size: 5.2 M

Downloading packages:

(1/9): apr-util-1.6.1-5.amzn2.0.2.x86_64.rpm	99 kB	00:00
(2/9): apr-1.7.0-9.amzn2.x86_64.rpm	122 kB	00:00
(3/9): generic-logos-httpd-18.0.0-4.amzn2.noarch.rpm	19 kB	00:00
(4/9): apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64.rpm	19 kB	00:00
(5/9): httpd-2.4.51-1.amzn2.x86_64.rpm	1.3 MB	00:00
(6/9): httpd-filesystem-2.4.51-1.amzn2.noarch.rpm	24 kB	00:00
(7/9): httpd-tools-2.4.51-1.amzn2.x86_64.rpm	88 kB	00:00
(8/9): mailcap-2.1.41-2.amzn2.noarch.rpm	31 kB	00:00
(9/9): mod_http2-1.15.19-1.amzn2.0.1.x86_64.rpm	149 kB	00:00

Total	7.6 MB/s	1.9 MB	00:00
-------	----------	--------	-------

Running transaction check

Running transaction test

Transaction test succeeded

Running transaction

Installing : apr-1.7.0-9.amzn2.x86_64	1/9
Installing : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64	2/9
Installing : apr-util-1.6.1-5.amzn2.0.2.x86_64	3/9
Installing : httpd-tools-2.4.51-1.amzn2.x86_64	4/9
Installing : generic-logos-httpd-18.0.0-4.amzn2.noarch	5/9
Installing : mailcap-2.1.41-2.amzn2.noarch	6/9

Installing : httpd-filesystem-2.4.51-1.amzn2.noarch	7/9
Installing : mod_http2-1.15.19-1.amzn2.0.1.x86_64	8/9
Installing : httpd-2.4.51-1.amzn2.x86_64	9/9
Verifying : apr-util-1.6.1-5.amzn2.0.2.x86_64	1/9
Verifying : httpd-2.4.51-1.amzn2.x86_64	2/9
Verifying : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64	3/9
Verifying : httpd-filesystem-2.4.51-1.amzn2.noarch	4/9
Verifying : apr-1.7.0-9.amzn2.x86_64	5/9
Verifying : mailcap-2.1.41-2.amzn2.noarch	6/9
Verifying : generic-logos-httpd-18.0.0-4.amzn2.noarch	7/9
Verifying : mod_http2-1.15.19-1.amzn2.0.1.x86_64	8/9
Verifying : httpd-tools-2.4.51-1.amzn2.x86_64	9/9

Installed:

httpd.x86_64 0:2.4.51-1.amzn2

Dependency Installed:

apr.x86_64 0:1.7.0-9.amzn2

apr-util.x86_64 0:1.6.1-5.amzn2.0.2

apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2

generic-logos-httpd.noarch 0:18.0.0-4.amzn2

httpd-filesystem.noarch 0:2.4.51-1.amzn2

httpd-tools.x86_64 0:2.4.51-1.amzn2

mailcap.noarch 0:2.1.41-2.amzn2

mod_http2.x86_64 0:1.15.19-1.amzn2.0.1

Complete!

```
[ec2-user@ip-172-31-18-49 ~]$ sudo systemctl start httpd
```

```
[ec2-user@ip-172-31-18-49 ~]$ sudo systemctl enable httpd
```


Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.

```
[ec2-user@ip-172-31-18-49 ~]$ sudo usermod -a -G apache ec2-user
```

```
[ec2-user@ip-172-31-18-49 ~]$ groups
```

```
ec2-user adm wheel systemd-journal
```

```
[ec2-user@ip-172-31-18-49 ~]$ sudo chown -R ec2-user:apache /var/www
```

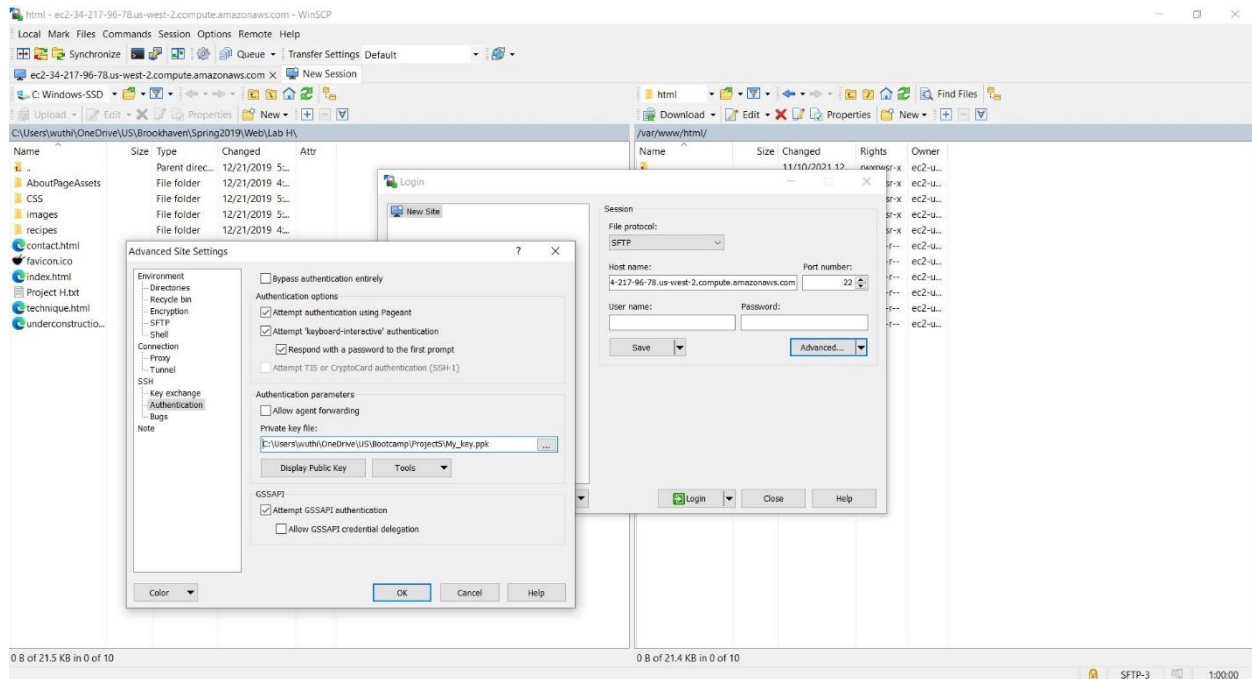
```
[ec2-user@ip-172-31-18-49 ~]$ sudo chmod 2775 /var/www
```

```
[ec2-user@ip-172-31-18-49 ~]$ find /var/www -type d -exec sudo chmod 2775 {} \;
```

```
[ec2-user@ip-172-31-18-49 ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
```

```
[ec2-user@ip-172-31-18-49 ~]$
```

- Use WinSCP to copy web documents into html directory on web server.



- Open web browser, enter web server name, and test access the website.

