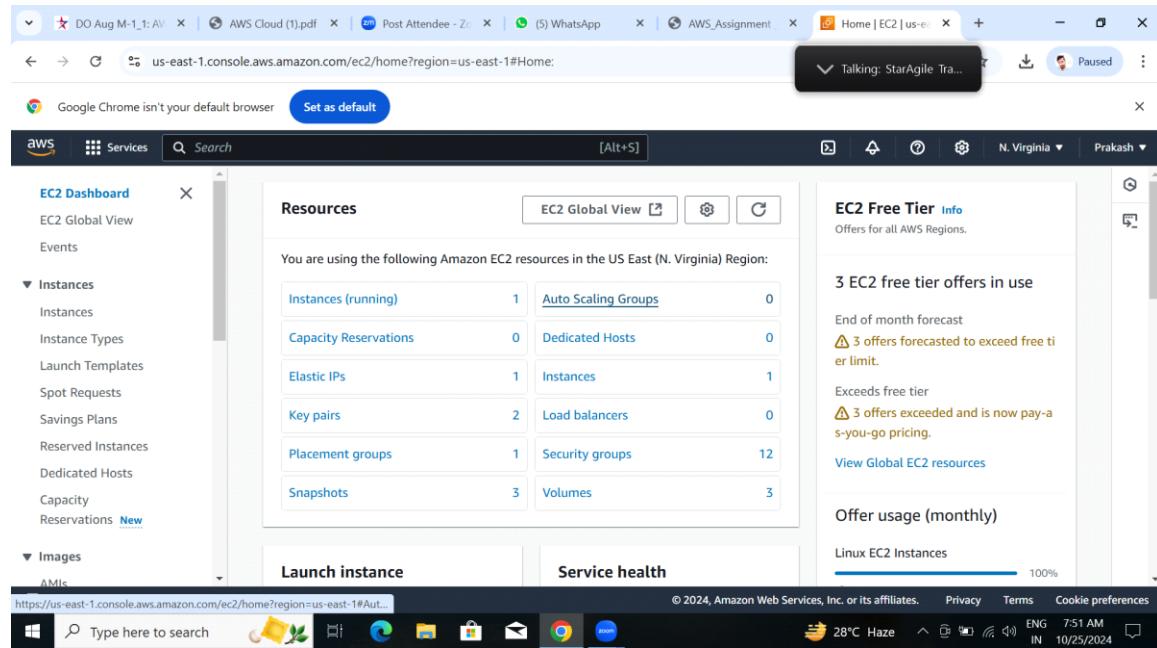


Assignment -1

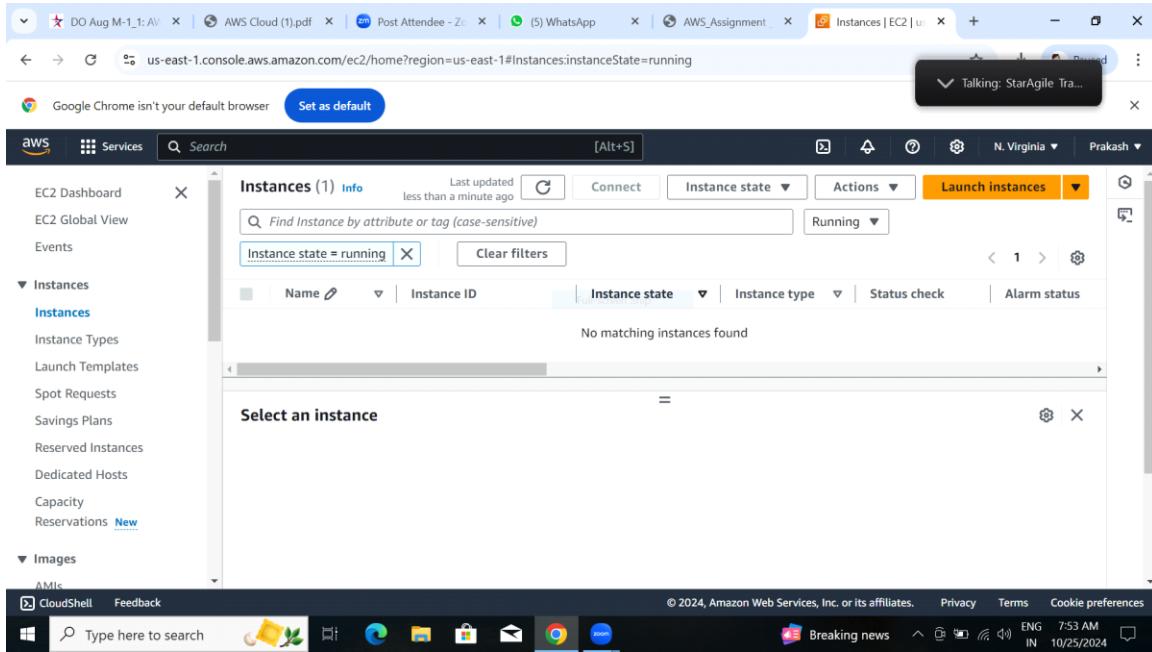
Aws Module -1

1. L1 - Demonstrate the AWS EC2 Ubuntu Instance Creation steps and connect to EC2 Instance using Mobaxterm/putty agent

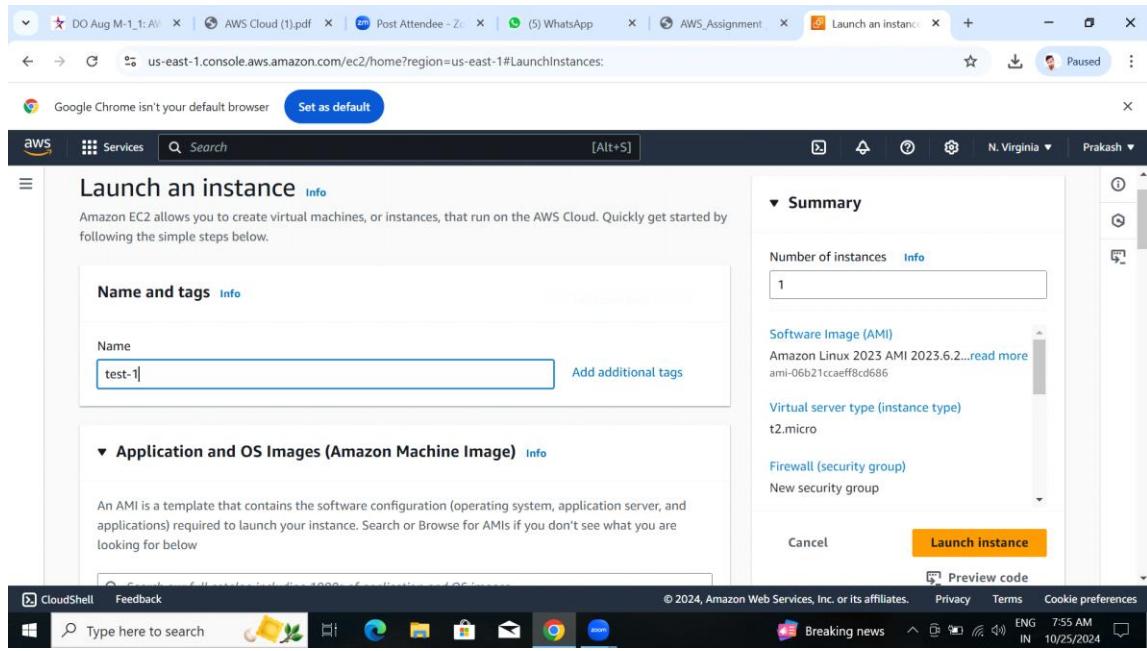
1. I have opened the AWS Web interface



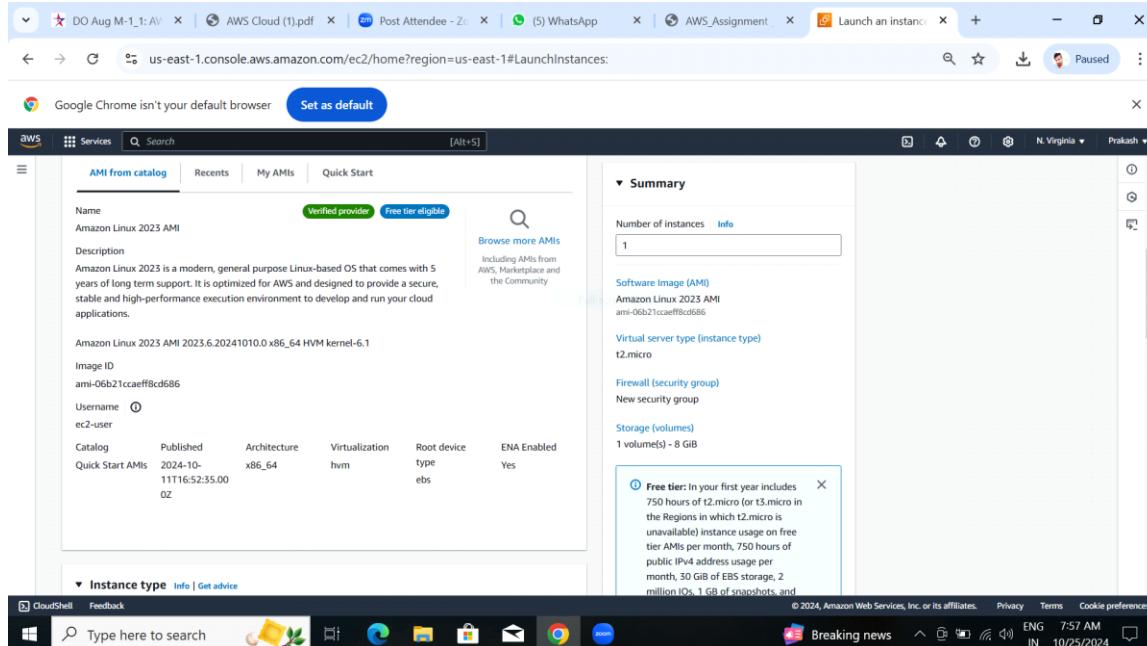
2. navigate to EC2 instance page



3. Then I have created the instance based on required configuration



Ec2 Instance name : test-1



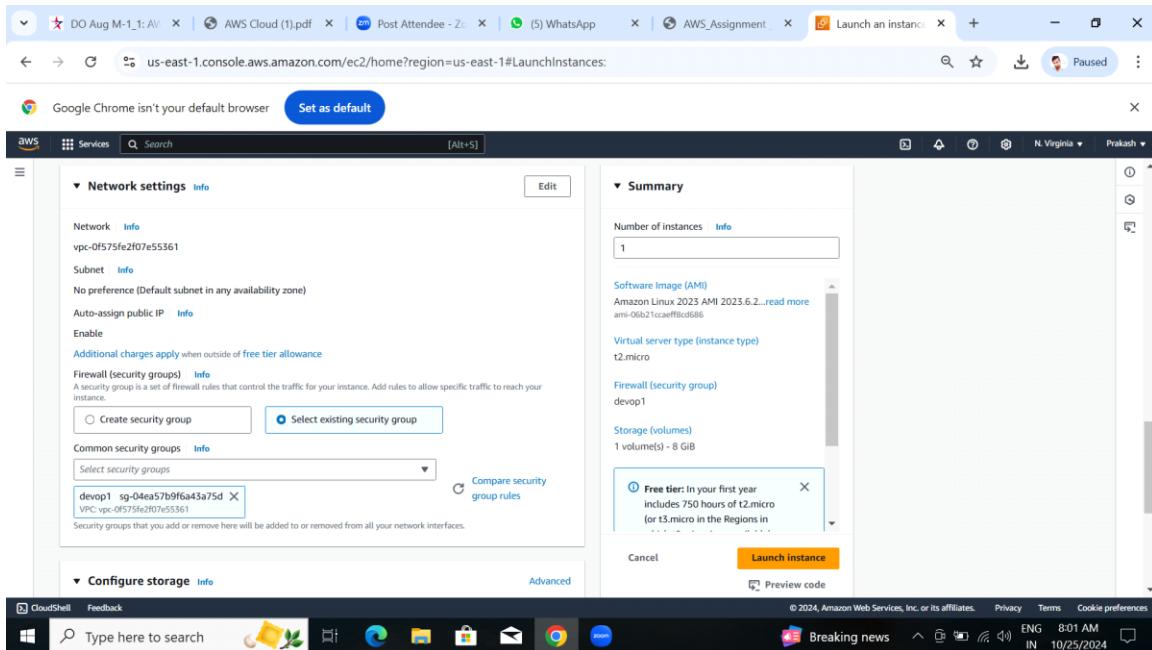
Instance Image : Amazon Linux

The screenshot shows the AWS Cloud console with the URL us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances. The browser tab is titled "Launch an instance". The main content area displays the "Launch an instance" wizard. The first step, "Select instance type", is selected. It shows a table with one row: "Quick Start AMIs" (t2.micro, 1 vCPU, 1 GiB Memory, x86_64, hvm, type ebs, 1 volume(s) - 8 GiB). Below the table, the "Instance type" section details the t2.micro configuration, including its family (t2), price (0.0167 USD per Hour), and usage (On-Demand Linux base pricing: 0.0167 USD per Hour). The "Additional costs apply for AMIs with pre-installed software" note is visible. The "Key pair (login)" section is expanded, showing a placeholder for a key pair name and a note about connecting to the instance. The "Summary" section on the right shows "Number of instances" set to 1, "Software Image (AMI)" as Amazon Linux 2023 AMI, and "Virtual server type (instance type)" as t2.micro. A tooltip for the "Free tier" is displayed, stating: "Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 1 volume(s) - 8 GiB". The bottom navigation bar includes CloudShell, Feedback, and a search bar.

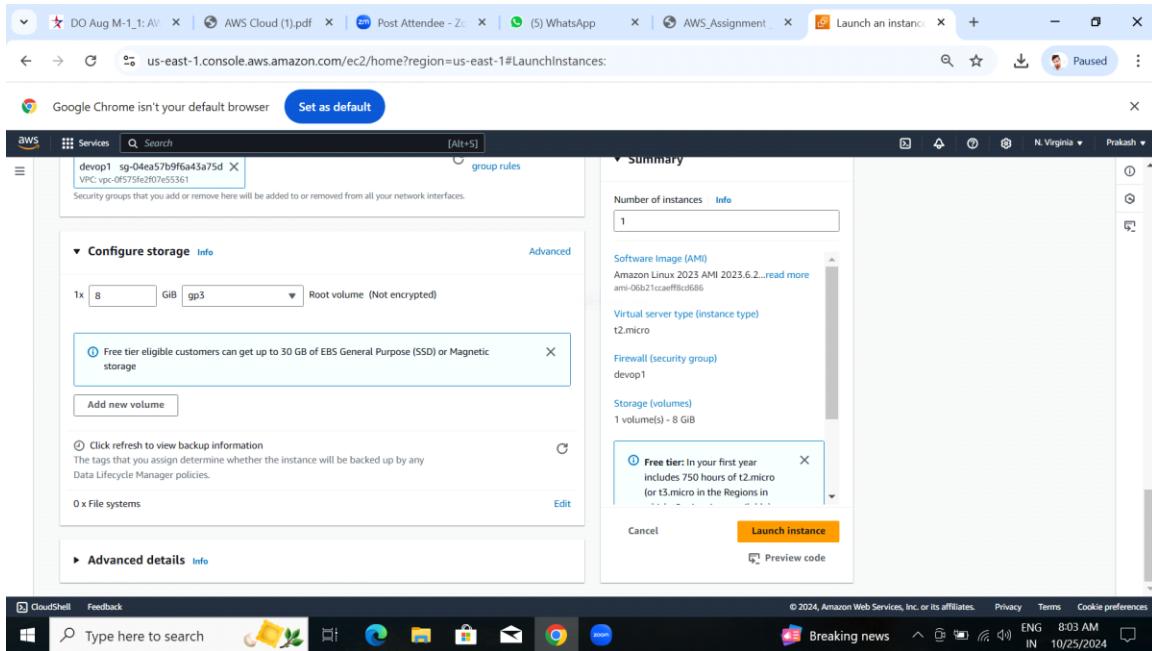
Instance type : t2.micro

The screenshot shows the AWS Cloud console with the URL us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances. The browser tab is titled "Launch an instance". The main content area displays the "Launch an instance" wizard. The second step, "Configure security and networking", is selected. A modal dialog box titled "Create key pair" is open. It contains fields for "Key pair name" (set to "key-pair-1") and "Key pair type" (set to "RSA"). It also includes sections for "Network settings" (with a note about subnet and auto-assigning public IP), "Private key file format" (set to "pem"), and a note about storing the private key securely. At the bottom of the dialog are "Cancel" and "Create key pair" buttons. The bottom navigation bar includes CloudShell, Feedback, and a search bar.

Login key-pair created and selected

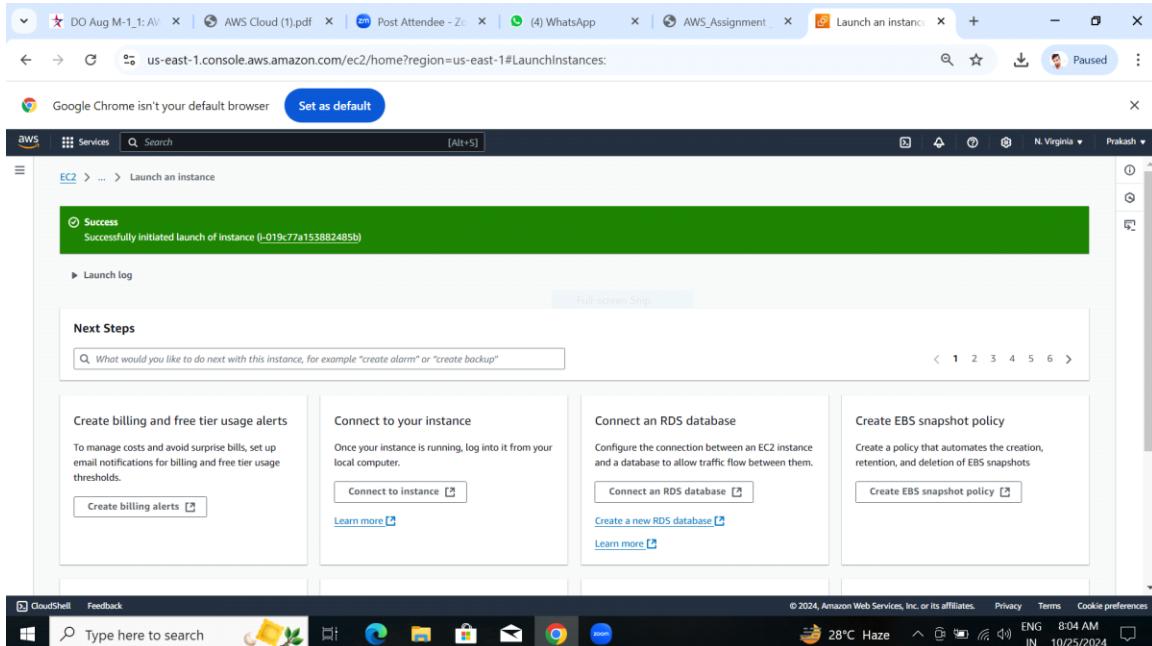


- Accept incoming request over HTTP protocol on all the ports making accept
- SSH connection on port 22



- 8 GB of General Purpose SSD attached

4. Successfully initiated launch the instance



The screenshot shows the AWS CloudWatch Metrics dashboard. At the top, there are several tabs and links: DO Aug M-1..., AWS Cloud (1), Post Attendee, (4) WhatsApp, AWS_Assignm, Launch an ins..., Instances | EC, and Instances | EC. The main content area is titled "Instances (1) Info". It displays a table with one row for "test-1". The columns include Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, and Public IPv6 DNS. The instance is listed as "Running" with a t2.micro type and an "Initializing" status check. The Public IPv4 DNS is ec2-3-80-182-144.com... and the Public IPv6 DNS is 3.80. The bottom of the screen shows the Windows taskbar with icons for CloudShell, Feedback, File Explorer, Edge, File Explorer, Mail, Google Chrome, and Microsoft Edge. The system tray shows the date and time as 10/25/2024, temperature as 28°C Haze, and battery level.

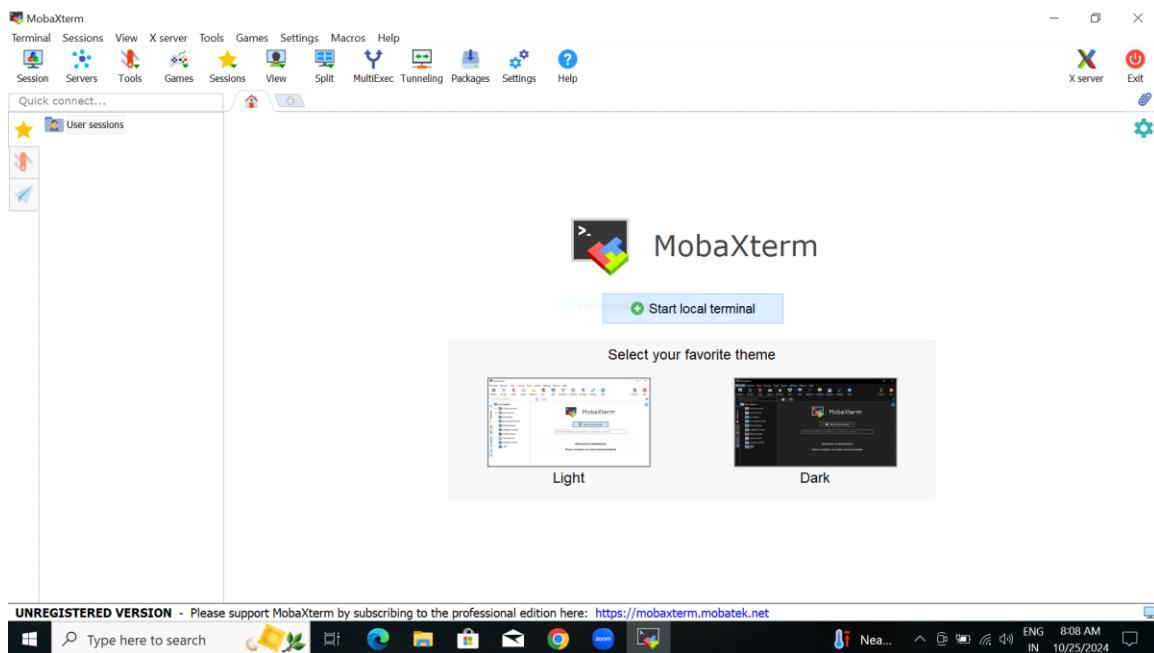
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv6 DNS
test-1	i-019c77a153882485b	Running	t2.micro	Initializing	View alarms +	us-east-1b	ec2-3-80-182-144.com...	3.80

Instance state : Running

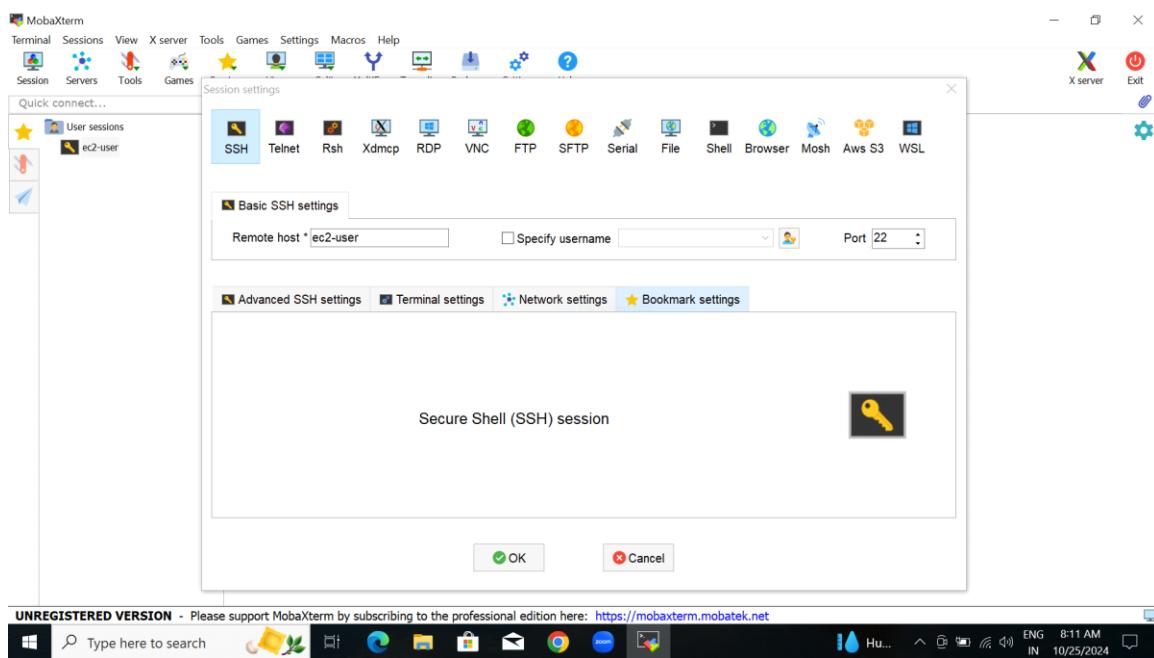
The screenshot shows the AWS CloudWatch Metrics interface. A metric named 'CPUUtilization' is selected. The chart displays CPU utilization over time, with a major peak occurring on August 25, 2024, reaching approximately 99.9%. A tooltip provides detailed information about this peak, including the maximum value of 99.9%, the timestamp of 2024-08-25T08:04:28Z, and the unit of %.

- Configuration has been changed to accept the HTTP request on port 80 only
- Configuration has been accepting the SSH request on port 22 only from the
- security group configuration verified

5 I have open MobaXterm interface

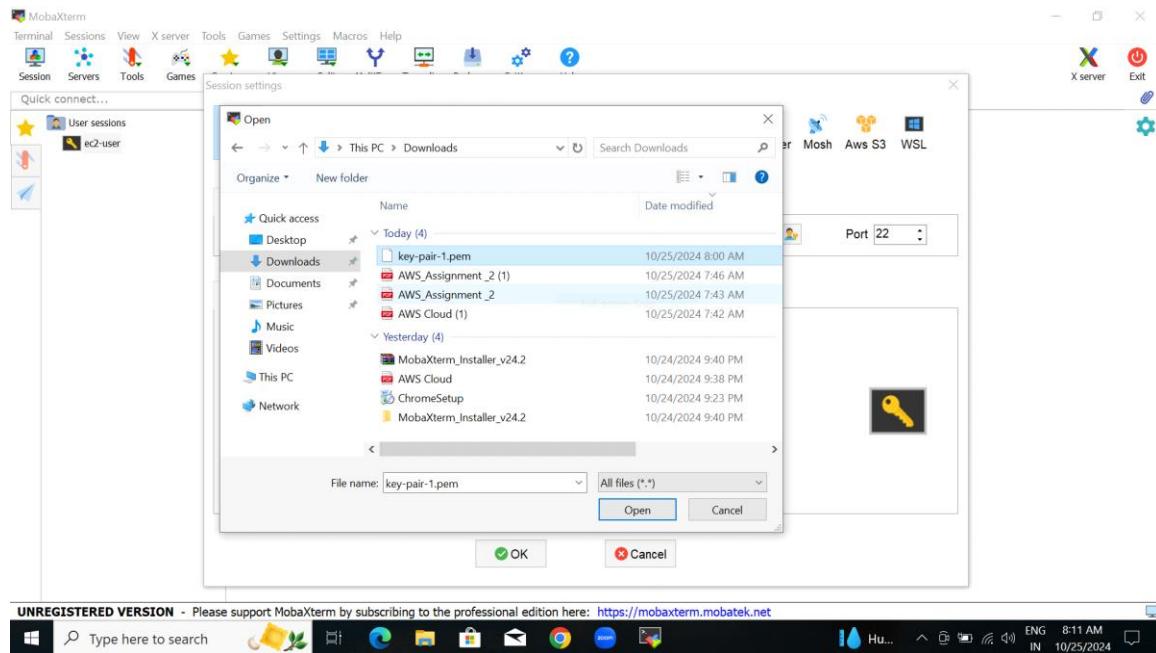


MobaXterm interface



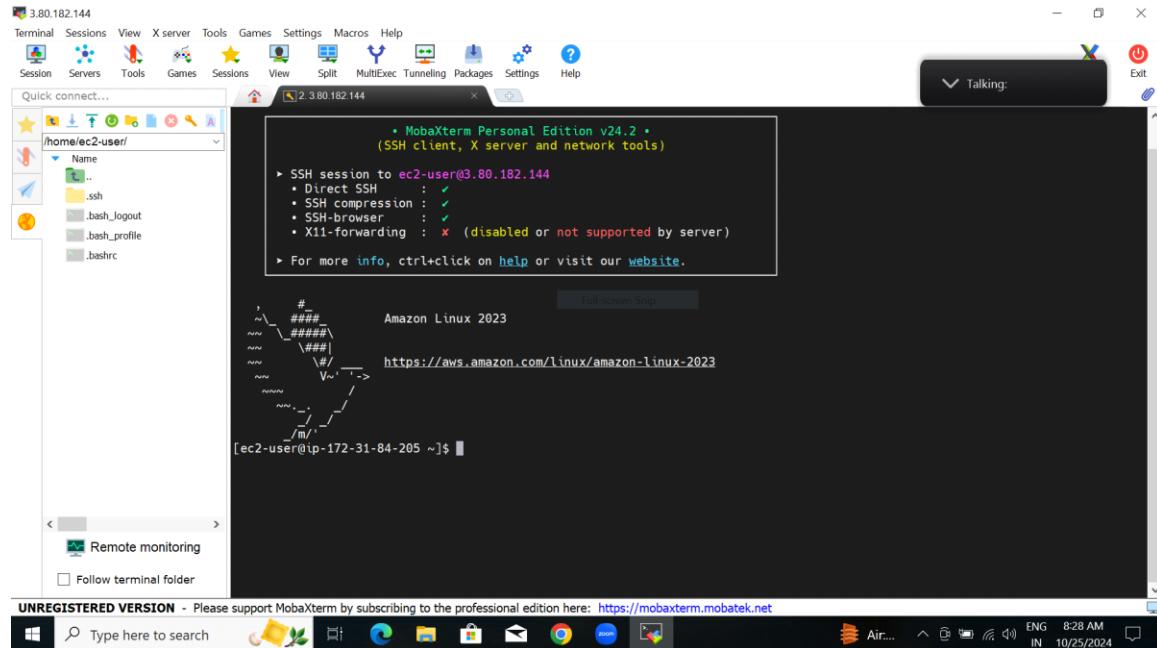
- SSH Via MobaXterm

- Username: ec2-user
- Password: Created key pair



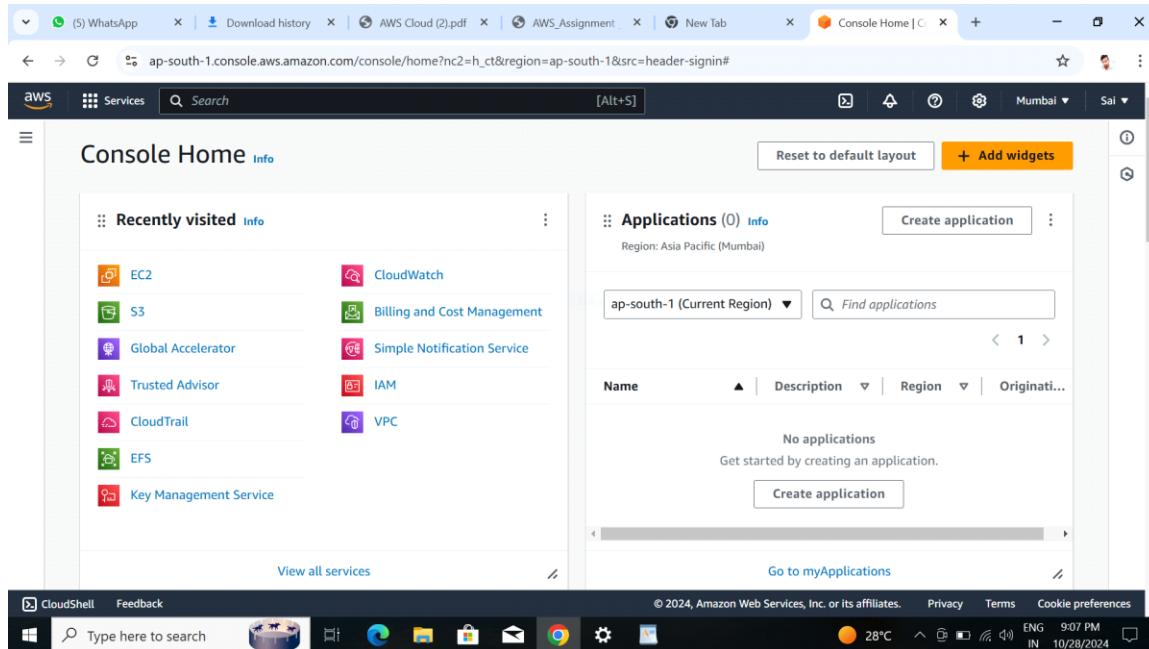
- Selected key-pair for login webserver through MobaXterm

6 . Successfully Instance connect to se

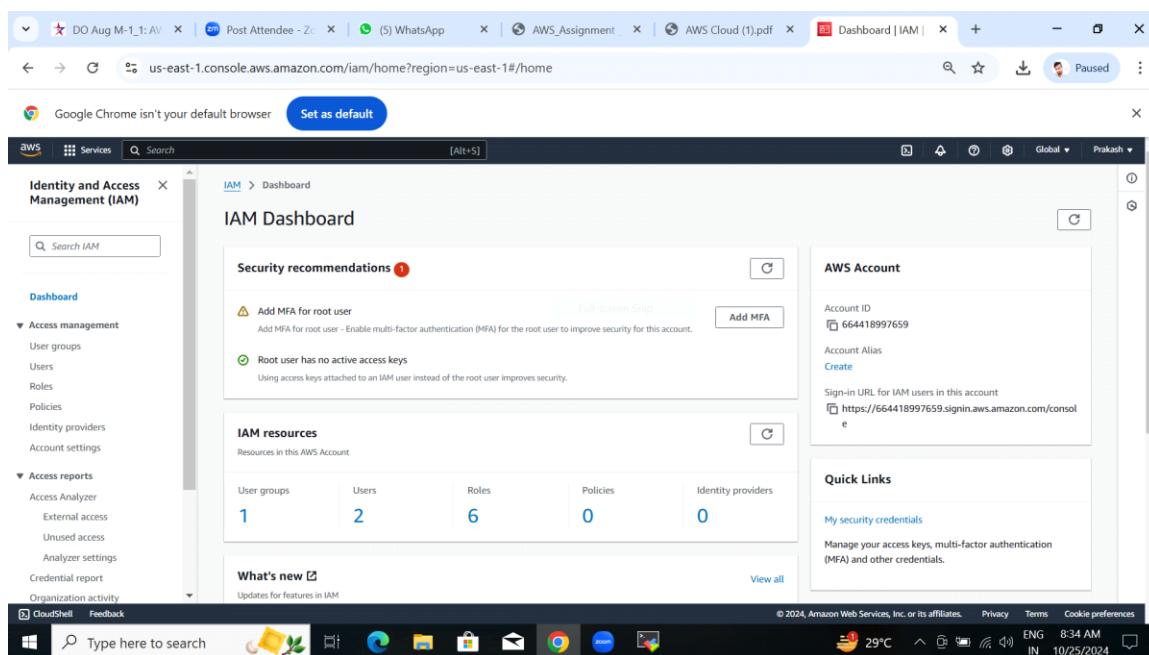


2. L2 - Login to AWS Console and Create IAM User,Role, and Group

1.I have opened the AWS Web interface



Aws console login successfully



Navigate to IAM dashboard page

Specify user details

User details

User name: iam-user-1

Provide user access to the AWS Management Console - optional

If you're providing console access to a person, it's a best practice [link] to manage their access in IAM Identity Center.

IAM user name - iam user-1

User type

Specify a user in Identity Center - Recommended

I want to create an IAM user

If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keypairs, you can generate them after you create this IAM user. [Learn more \[link\]](#)

Created custom password successfully

The screenshot shows a browser window with multiple tabs open, including 'DO Aug M-1: AW', 'Post Attendee - Z', '(5) WhatsApp', 'AWS Assignment', 'AWS Cloud (1).pdf', and 'Create user | IAM'. The main content is the 'Create user | IAM' page, specifically the 'Choose one or more policies to attach to your new user' section. A search bar at the top left contains 'ec2'. Below it, a table lists AWS managed policies. The 'AmazonEC2FullAccess' policy is selected, indicated by a checked checkbox and highlighted with a blue border. Other policies listed include 'AmazonEC2ContainerRegistryFullAccess', 'AmazonEC2ContainerRegistryPowerUser', 'AmazonEC2ContainerRegistryPullOnly', 'AmazonEC2ContainerRegistryReadOnly', 'AmazonEC2ContainerServiceAutoscaleRole', 'AmazonEC2ContainerServiceEventsRole', 'AmazonEC2ContainerServiceforEC2Role', and 'AmazonEC2ContainerServiceRole'. The table has columns for 'Policy name', 'Type', and 'Attached entities'. The footer of the browser window shows standard navigation icons and a status bar indicating the date and time.

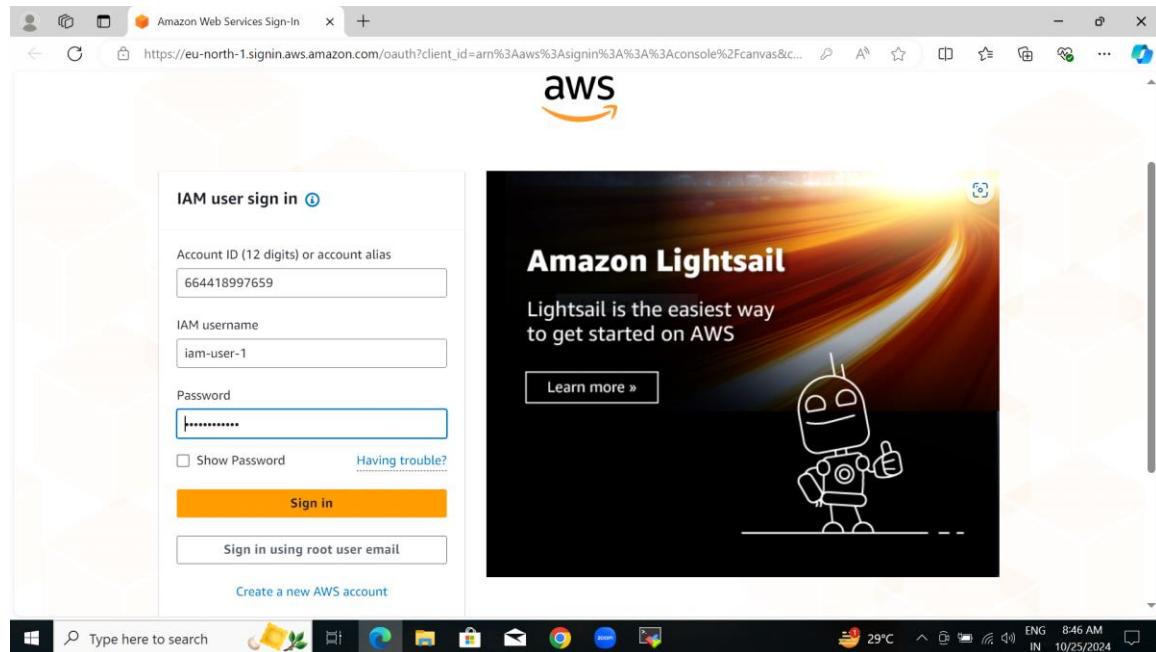
Attach the policy to IAM user

The screenshot shows the AWS IAM User Creation Step 2: Set permissions page. The user name is set to 'iam-user-1'. Under 'Permissions summary', two policies are attached: 'AmazonEC2FullAccess' and 'IAMUserChangePassword'. Both are listed as 'AWS managed' type and 'Permissions policy'. There are no tags associated with the user.

Successfully attach Ec2 Fulladministration access to IAM user

The screenshot shows the AWS IAM User Creation Step 3: Review and create page. A green banner at the top indicates 'User created successfully'. The user can view and download the user's password and email instructions for signing in to the AWS Management Console. The 'Console sign-in details' section shows the console sign-in URL as <https://664418997659.signin.aws.amazon.com/console>, the user name as 'iam-user-1', and the console password as '*****'. There is a 'Show' link next to the password field.

Successfully create IAM user



IAM user sign in page

The screenshot shows the AWS Console Home page. At the top, there is a message: "Failed to load your custom layout. Try refreshing the page." Below this, the "Recently visited" section shows "No recently visited services". The "Applications" section shows "0 Applications" and a "Create application" button. A message "Access denied" is displayed in a red-bordered box. The bottom of the screen shows the Windows taskbar with various pinned icons and system status.

Successfully login IAM user account

Group: groups is used to gave generic permission to users

The screenshot shows the AWS IAM User groups page. The left sidebar includes options like Dashboard, Access management (User groups, Users, Roles, Policies, Identity providers, Account settings), Access reports (External access, Unused access, Analyzer settings, Credential report, Organization activity), CloudShell, and Feedback. The main content area shows "User groups (0)" and a message: "A user group is a collection of IAM users. Use groups to specify permissions for a collection of users." A search bar and a "Create group" button are present. The bottom of the screen shows the Windows taskbar with various pinned icons and system status.

user groups dashboard

The screenshot shows the AWS IAM User Groups 'Create user group' interface. In the 'Name the group' section, the user has entered 'group-1'. Under 'Add users to the group - Optional', three IAM users are listed: 'iam-user-1', 'user-1', and 'user-2'. The 'Attach permissions policies - Optional' section is collapsed. The left sidebar shows navigation options like Dashboard, Access management (User groups, Roles, Policies, Identity providers, Account settings), and Access reports (Access Analyzer, External access, Unused access, Analyzer settings, Credential report, Organization activity). The bottom of the screen shows the Windows taskbar with various pinned icons and system status.

User group name - group-1

The screenshot shows the AWS IAM User Groups creation interface. On the left, a sidebar lists 'Identity and Access Management (IAM)' services like Dashboard, User groups, Roles, Policies, and Access reports. The main area shows two users selected: 'iam-user-1' and 'user-2'. Below this, a section titled 'Attach permissions policies - Optional (2/958)' displays a list of AWS managed policies. One policy, 'IAMFullAccess', is checked and highlighted in blue. Other policies listed include AWSIdentityCenterAllowList, AWSQuickSightListIAM, IAMAccessAdvisorReadOnly, IAMAccessAnalyzerFullAccess, IAMAccessAnalyzerReadOnly, IAMReadOnlyAccess, and IAMReadonlyAccess.

Attach generic permission to IAM users

The screenshot shows the AWS IAM User Groups creation interface after a group has been created. A green success message at the top states 'group-1 user group created.' The main area shows the 'User groups (1)' table with one entry: 'group-1'. The table includes columns for Group name, Users, Permissions, and Creation time. The 'Permissions' column for 'group-1' shows 'Defined' with a green checkmark.

Successfully created group

Roles : Roles is to give temporary access to user

The screenshot shows the AWS IAM Roles dashboard. The left sidebar navigation includes 'Identity and Access Management (IAM)', 'Dashboard', 'Access management' (with 'User groups', 'Users', and 'Roles' selected), 'Policies', 'Identity providers', 'Account settings', 'Access reports' (with 'Access Analyzer', 'External access', 'Unused access', 'Analyzer settings', and 'Credential report'), and 'Organization activity'. The main content area displays a table titled 'Roles (5) Info' with columns for 'Role name', 'Trusted entities', and 'Last activity'. The roles listed are: 'AWSServiceRoleForBackup' (AWS Service: backup (Service-Linked), 9 hours ago); 'AWSServiceRoleForElasticLoadBalancing' (AWS Service: elasticloadbalancing (S), 7 days ago); 'AWSServiceRoleForSupport' (AWS Service: support (Service-Linked), -); and 'AWSServiceRoleForTrustedAdvisor' (AWS Service: trustedadvisor (Service-Linked), -). Below the table, there are sections for 'Roles Anywhere' (info: 'Authenticate your non AWS workloads and securely provide access to AWS services.', 'Manage' button), 'X.509 Standard' (info: 'Operate your non AWS workloads using the same authentication and authorization strategy that you use within AWS.'), and 'Temporary credentials' (info: 'Use temporary credentials with ease and benefit from the enhanced security they provide.').

Roles dashboard page

The screenshot shows the AWS IAM 'Create role' wizard. The current step is 'Step 2: Add permissions'. The 'Trusted entity type' section is displayed, showing five options:

- AWS service: Allows AWS services like EC2, Lambda, or others to perform actions in this account.
- AWS account: Allows entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- SAML 2.0 federation: Allows users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- Custom trust policy: Creates a custom trust policy to enable others to perform actions in this account.
- Web identity: Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

Below this, the 'An AWS account' section is expanded, showing two options:

- This account (664418997659)
- Another AWS account

Under 'Options', there are two checkboxes:

- Require external ID (Best practice when a third party will assume this role)
- Require MFA (Requires that the assuming entity use multi-factor authentication)

The browser's address bar shows: us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/roles/create

Select the trusted entity type : AWS account

The screenshot shows the AWS IAM 'Create role' wizard. The current step is 'Step 3: Name, review, and create'. The 'Add permissions' section is displayed, showing a list of 'Permissions policies (1/958)'. A search bar at the top of the list is set to 'admin'.

Policy name	Type	Description
AdministratorAccess	AWS managed - job function	Provides full access to AWS services an...
AdministratorAccess-Amplify	AWS managed	Grants account administrative permisi...
AdministratorAccess-AWSElasticBeanstalk	AWS managed	Grants account administrative permisi...
AmazonAPIGatewayAdministrator	AWS managed	Provides full access to create/edit/delete...
AmazonNimbleStudio-StudioAdmin	AWS managed	This policy grants access to Amazon Ni...
AmazonSageMakerAdmin-ServiceCatalogProductsS...	AWS managed	Service role policy used by the AWS Se...
AmazonSecurityLakeAdministrator	AWS managed	Provides full access to Amazon Securi...
AmazonWorkSpacesAdmin	AWS managed	Provides access to Amazon WorkSpace...
AmazonWorkSpacesApplicationManagerAdminAccess	AWS managed	Provides administrator access for pack...

The browser's address bar shows: us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/roles/create?trustedEntityType=AWS_ACCOUNT

Add permissions to Role User

The screenshot shows the AWS IAM 'Create role' wizard. The current step is 'Step 2: Add permissions'. In the 'Role details' section, the role name is 'role-1' and the description is 'allow to access Alladministration policies'. In the 'Step 1: Select trusted entities' section, a trust policy is defined:

```
1+ [ "Version": "2012-10-17", 2- "Statement": [ 3- { 4- "Effect": "Allow", 5- "Action": "sts:AssumeRole", 6- "Principal": { 7- "AWS": "66441897659" 8- }, 9- "Condition": {} 10- } ] }
```

At the bottom of the browser window, the taskbar shows various open tabs and system status.

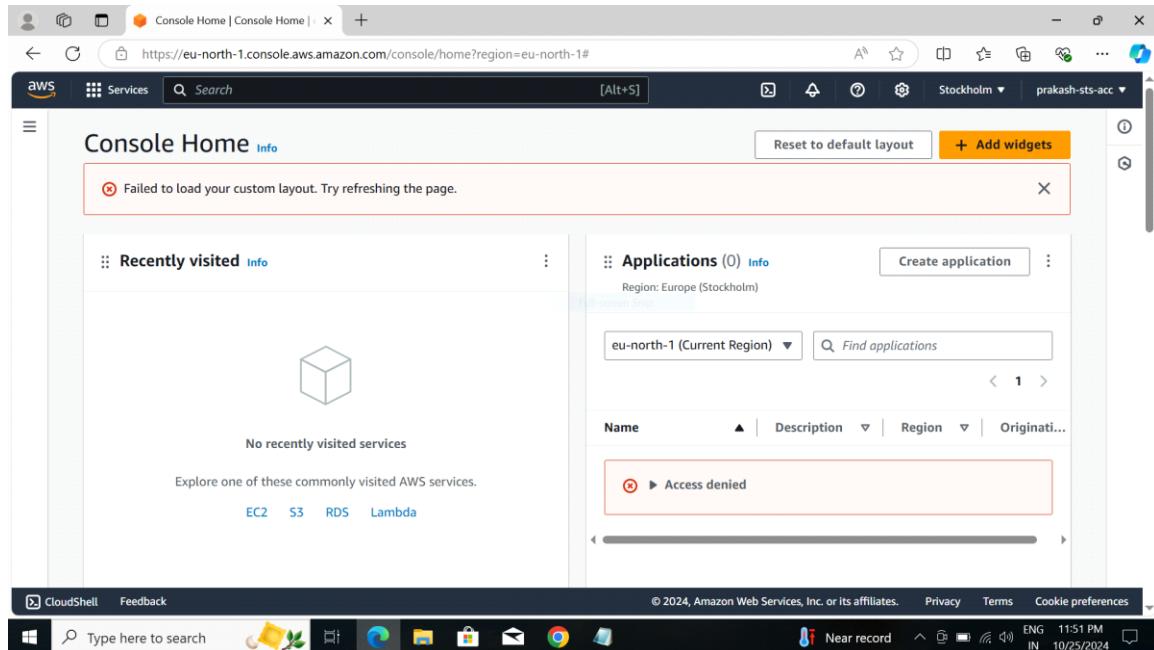
Role name : role-1

The screenshot shows the AWS IAM Roles page. At the top, a banner says "Role role-1 created." Below it, the "Roles (6) Info" section defines an IAM role as an identity that can be assumed by entities. A table lists six roles, including "role-1" which was just created. The table columns are "Role name", "Trusted entities", and "Last activity". The "Create role" button is visible at the top right. The left sidebar shows navigation options like Dashboard, Access management, and Roles.

Successfully created Role user

The screenshot shows the "Switch Role" page. It allows users to switch roles across accounts. The "Account ID" field is filled with "664418997659". The "IAM role name" field contains "role-1". The "Display name - optional" field has "role-1 @ 664418997659". The "Display color - optional" dropdown is set to "None". The page includes a note about temporarily taking on new role permissions and returning to original ones. The bottom navigation bar shows standard Windows icons and the date/time as 10/25/2024.

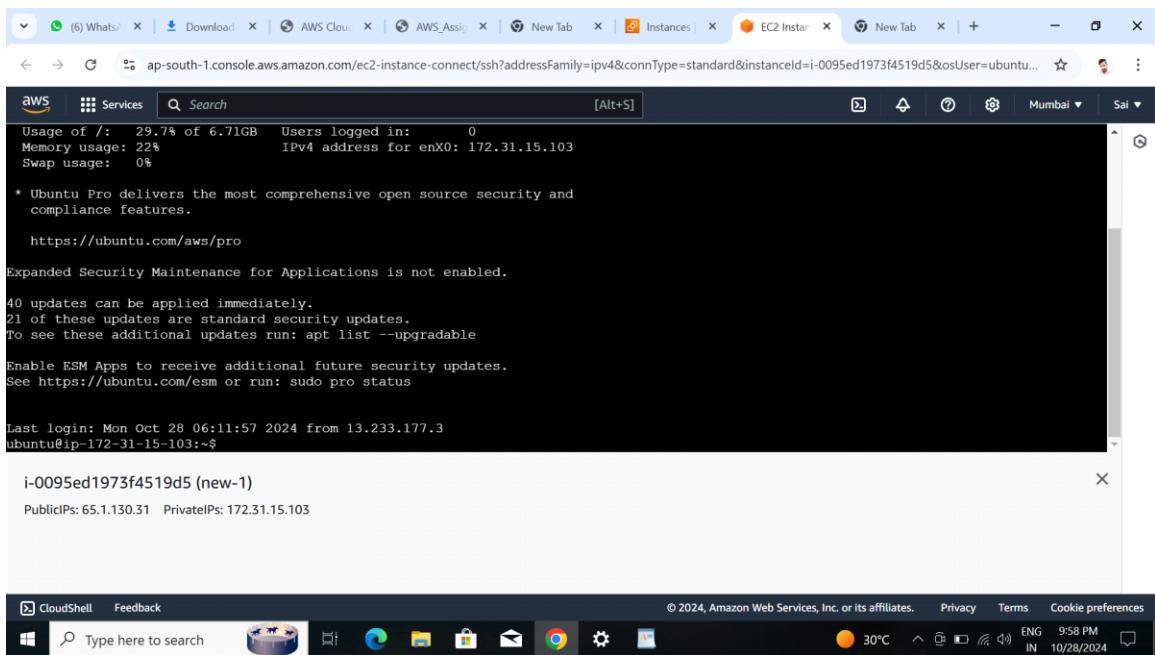
- Switch the Role account by add STS policy to role account ,role login details



Successfully login role account

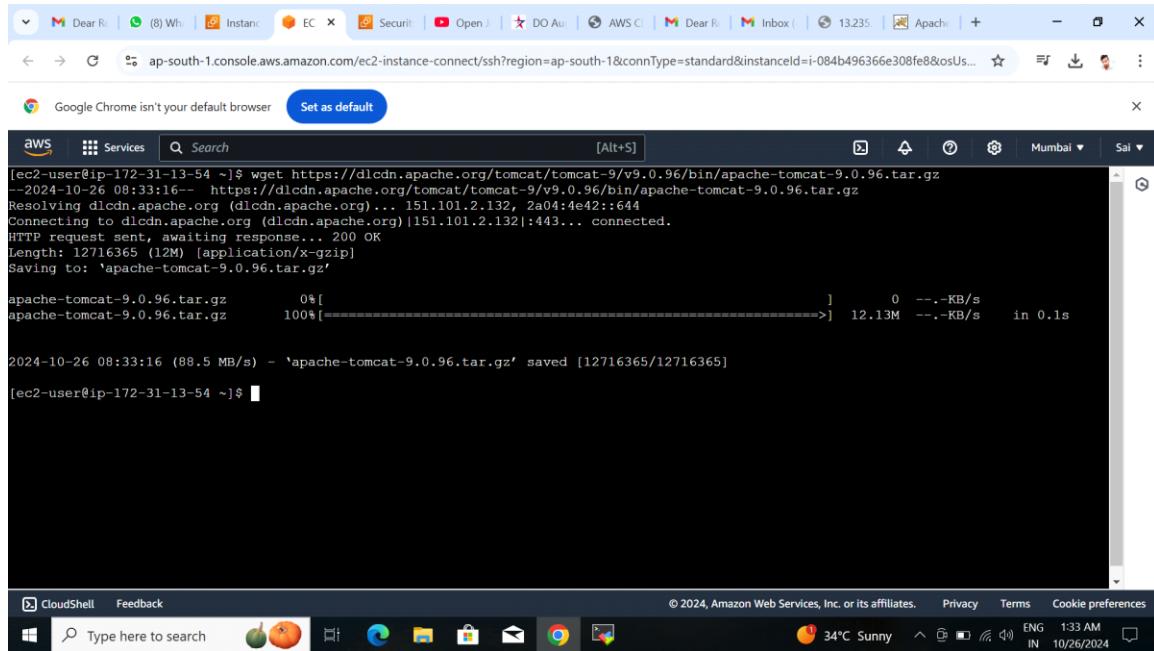
3. L3 - Launch AWS EC2 Ubuntu Instance and configure the Security Group - Inbound Rule: 8080. Justify the usage of Inbound Rules

1. Launch Ec2 - Instance Successfully



launch the ec2 - ubuntu Instance

2. With the help of 8080 port to login Tomcat in web-server



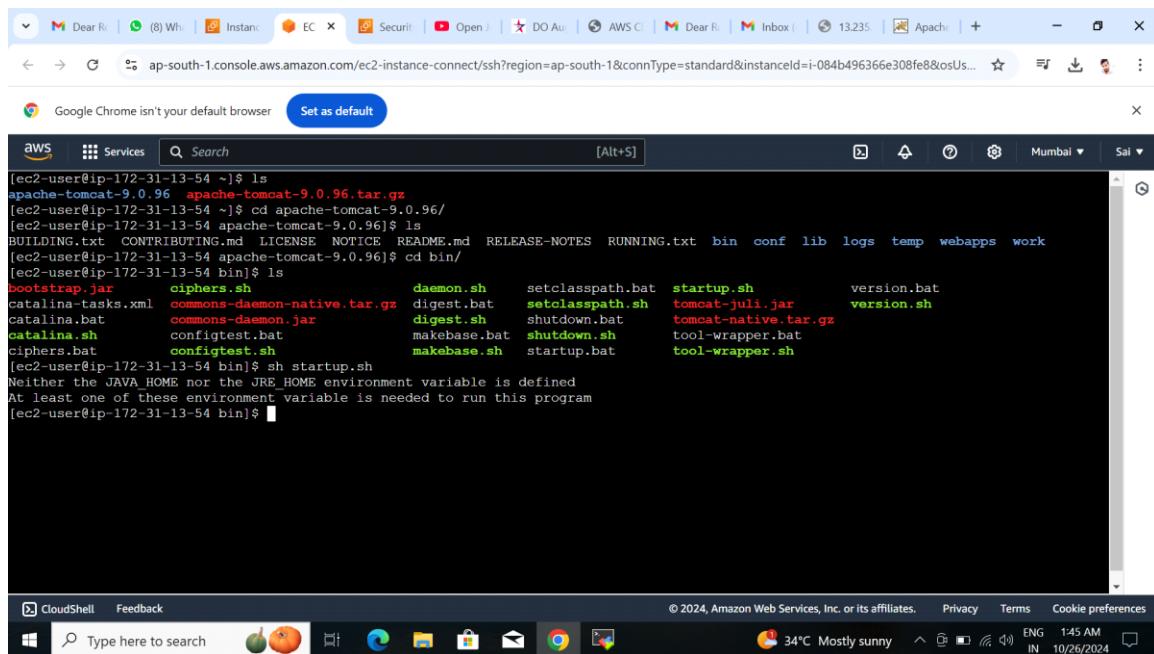
```
[ec2-user@ip-172-31-13-54 ~]$ wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.96/bin/apache-tomcat-9.0.96.tar.gz
--2024-10-26 08:33:16-- https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.96/bin/apache-tomcat-9.0.96.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org) ... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 12716365 (12M) [application/x-gzip]
Saving to: 'apache-tomcat-9.0.96.tar.gz'

apache-tomcat-9.0.96.tar.gz      0%[=====] 12.13M --.-KB/s   in 0.1s

2024-10-26 08:33:16 (88.5 MB/s) - 'apache-tomcat-9.0.96.tar.gz' saved [12716365/12716365]

[ec2-user@ip-172-31-13-54 ~]$
```

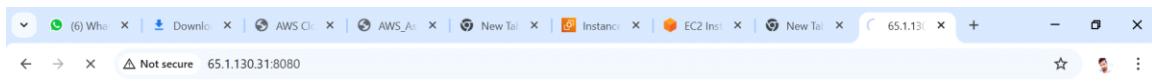
Successfully Download TOMCAT server



```
[ec2-user@ip-172-31-13-54 ~]$ ls
apache-tomcat-9.0.96 apache-tomcat-9.0.96.tar.gz
[ec2-user@ip-172-31-13-54 ~]$ cd apache-tomcat-9.0.96/
[ec2-user@ip-172-31-13-54 apache-tomcat-9.0.96]$ ls
BUILDING.txt CONTRIBUTING.md LICENSE NOTICE README.md RELEASE-NOTES RUNNING.txt bin conf lib logs temp webapps work
[ec2-user@ip-172-31-13-54 apache-tomcat-9.0.96]$ cd bin/
[ec2-user@ip-172-31-13-54 bin]$ ls
bootstrap.jar ciphers.sh daemon.sh setclasspath.bat startup.sh version.bat
catalina.bat commons-daemon-native.tar.gz digest.bat setclasspath.sh tomcat-juli.jar version.sh
catalina.sh configtest.bat makebase.bat shutdown.sh tomcat-native.tar.gz
ciphers.bat configtest.sh makebase.sh startup.sh tool-wrapper.bat
[ec2-user@ip-172-31-13-54 bin]$ sh startup.sh
Neither the JAVA_HOME nor the JRE_HOME environment variable is defined
At least one of these environment variable is needed to run this program
[ec2-user@ip-172-31-13-54 bin]$
```

To start tomcate server

4. Try to access web service with public IP



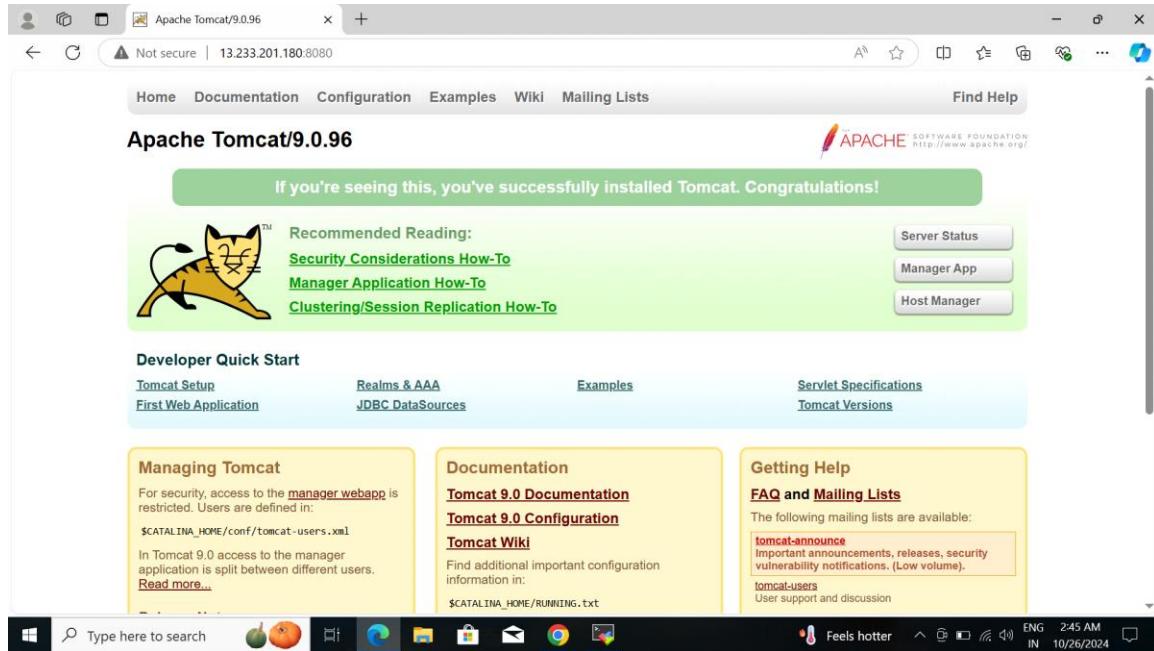
4. Change Security Group permission then try again to access web -server

The screenshot shows the AWS EC2 Instances page. A specific instance, 'i-055e049cff3e25869', is selected. The 'Security' tab is active, displaying security group details and inbound rules. An inbound rule allows traffic on port 8080 from 0.0.0.0/0 to the security group 'sg-024c71168dadcc2c09 (sg8080)'. The instance has an IMDSv2 IAM role and was launched on Saturday, October 26, 2024, at 00:07:59 GMT-0700.

Name	Security group rule ID	Port range	Protocol	Source	Security groups
-	sgr-05a57f4ac0116328c	443	TCP	0.0.0.0/0	sg8080
-	sgr-05d3fe5127a360d24	80	TCP	0.0.0.0/0	sg8080
-	sgr-086ba91684de60352	8080	TCP	0.0.0.0/0	sg8080

- Add 8080 port access to EC2 - instance for access of web-server with public ip

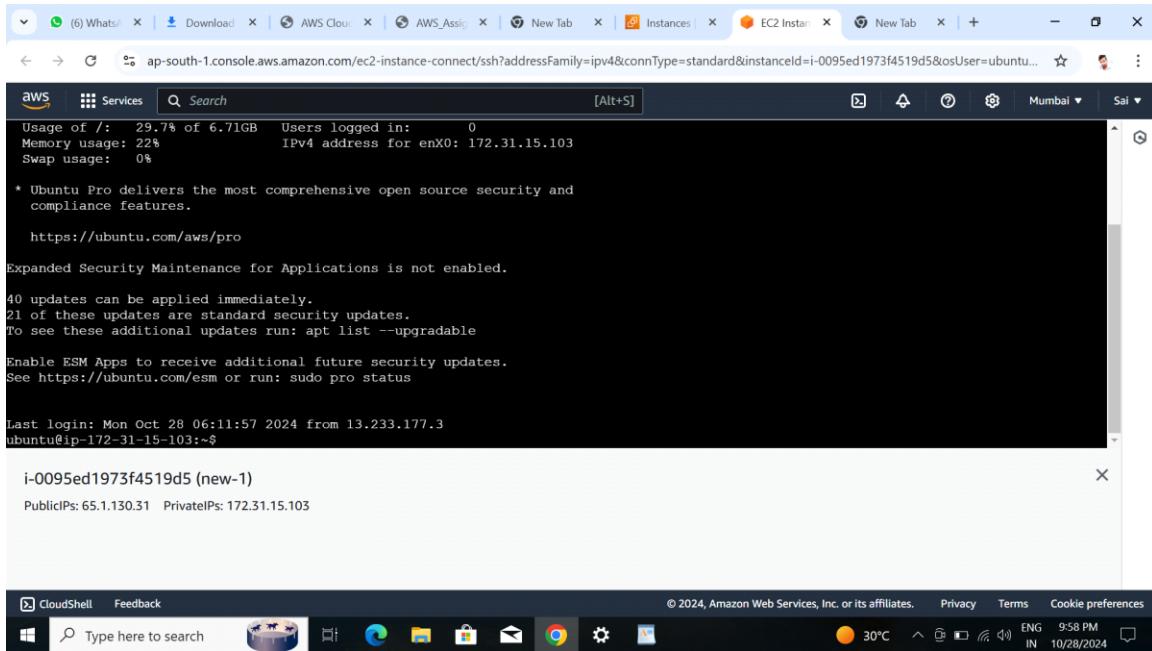
5. After changing the security group permission the again try to access web - server



Successfully login tomcate server with 8080 port

4. L4 - Connect to the AWS EC2 Ubuntu Instance and Update default packages, install JDK, Maven, Git, and validate the versions

1.- Connect to the AWS EC2 Ubuntu Instance



```
Usage of /: 29.7% of 6.71GB  Users logged in: 0
Memory usage: 22%          IPv4 address for enX0: 172.31.15.103
Swap usage: 0%
* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.
  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

40 updates can be applied immediately.
21 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Mon Oct 28 06:11:57 2024 from 13.233.177.3
ubuntu@ip-172-31-15-103:~$
```

i-0095ed1973f4519d5 (new-1)
PublicIPs: 65.1.130.31 PrivateIPs: 172.31.15.103

Successfully launch AWS EC2 Ubuntu Instance

2 . Update default packages using ec2 -ubuntu Instance & MobaXterm

```

ubuntu@ip-172-31-15-103:~$ sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
ubuntu@ip-172-31-15-103:~$ 

```

i-0095ed1973f4519d5 (new-1)

PublicIPs: 65.1.130.31 PrivateIPs: 172.31.15.103

Update default packages using ec2 -ubuntu Instance

```

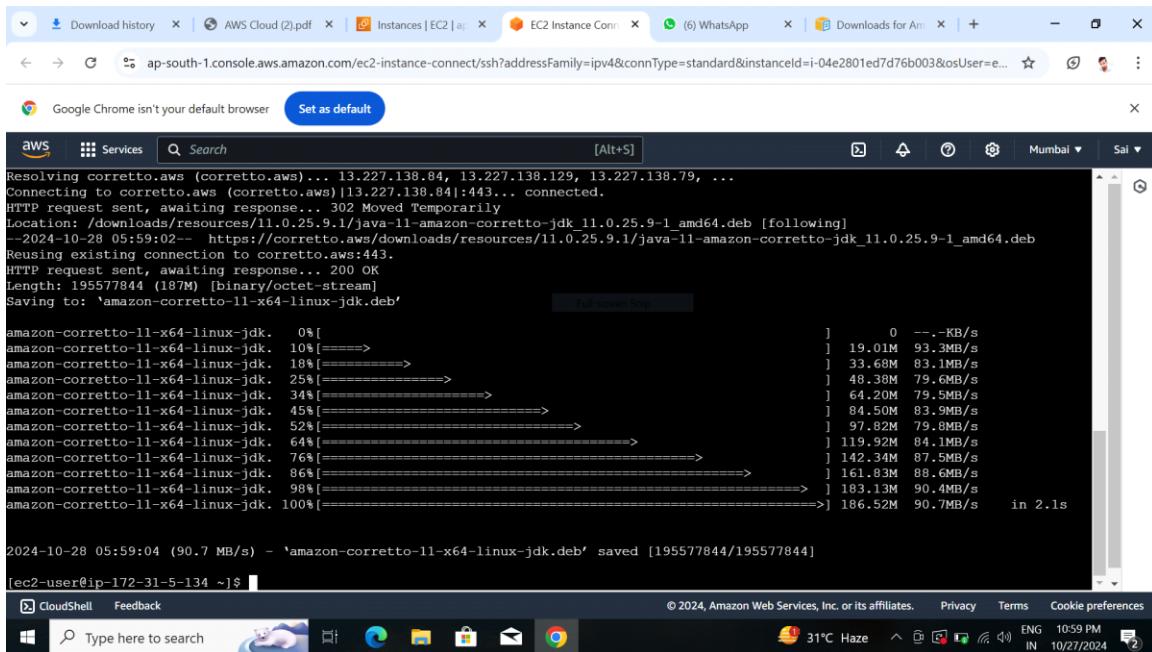
ubuntu@ip-172-31-15-103:~$ sudo apt-get update
Get:23 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 B]
Get:24 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [14.7 kB]
Get:25 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [3820 B]
Get:26 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [940 B]
Get:27 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [552 B]
Get:28 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [208 B]
Get:29 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [112 B]
Get:30 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [10.6 kB]
Get:31 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [10.8 kB]
Get:32 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [21.1 kB]
Get:33 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1104 B]
Get:34 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [216 B]
Get:35 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 c-n-f Metadata [116 B]
Get:36 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Get:37 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:38 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [433 kB]
Get:39 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [93.2 kB]
Get:40 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [7216 B]
Get:41 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [5820 B]
Get:42 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [556 kB]
Get:43 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [148 kB]
Get:44 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [51.9 kB]
Get:45 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [13.5 kB]
Get:46 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [388 kB]
Get:47 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [74.8 kB]
Get:48 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [212 B]
Get:49 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [10.9 kB]
Get:50 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [2808 B]
Get:51 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [212 B]
Get:52 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [344 B]
Fetched 30.4 MB in 12s (2637 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-15-103:~$ 

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Update default packages using MobaXterm -ubuntu Instance

3. Launch Ec2 - ubuntu Instance & Install JDK using corretto 11



```
Download history | AWS Cloud (2).pdf | Instances | EC2 | EC2 Instance Conn | (6) WhatsApp | Downloads for All | + | - | X
← → ⌂ ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?addressFamily=ipv4&connType=standard&instanceId=i-04e2801ed7d76b003&osUser=e... ☆ ⌂ : Set as default
Google Chrome isn't your default browser
Services Search [Alt+S] Mumbai Sai
Resolving corretto.aws (corretto.aws)... 13.227.138.84, 13.227.138.129, 13.227.138.79, ...
Connecting to corretto.aws (corretto.aws)|13.227.138.84|:443... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: /downloads/resources/11.0.25.9.1/java-11-amazon-corretto-jdk_11.0.25.9-1_amd64.deb [following]
--2024-10-28 05:59:02-- https://corretto.aws/downloads/resources/11.0.25.9.1/java-11-amazon-corretto-jdk_11.0.25.9-1_amd64.deb
Reusing existing connection to corretto.aws:443.
HTTP request sent, awaiting response... 200 OK
Length: 195577844 (187M) [binary/octet-stream]
Saving to: 'amazon-corretto-11-x64-linux-jdk.deb'

amazon-corretto-11-x64-linux-jdk. 0%[=====] 0 --.KB/s
amazon-corretto-11-x64-linux-jdk. 10%[=====] 19.01M 93.3MB/s
amazon-corretto-11-x64-linux-jdk. 18%[=====] 33.68M 83.1MB/s
amazon-corretto-11-x64-linux-jdk. 25%[=====] 48.38M 79.6MB/s
amazon-corretto-11-x64-linux-jdk. 34%[=====] 64.20M 79.5MB/s
amazon-corretto-11-x64-linux-jdk. 45%[=====] 84.50M 83.9MB/s
amazon-corretto-11-x64-linux-jdk. 52%[=====] 97.82M 79.6MB/s
amazon-corretto-11-x64-linux-jdk. 64%[=====] 119.92M 84.1MB/s
amazon-corretto-11-x64-linux-jdk. 76%[=====] 142.34M 87.5MB/s
amazon-corretto-11-x64-linux-jdk. 86%[=====] 161.83M 88.6MB/s
amazon-corretto-11-x64-linux-jdk. 98%[=====] 183.13M 90.4MB/s
amazon-corretto-11-x64-linux-jdk. 100%[=====] 186.52M 90.7MB/s in 2.1s

2024-10-28 05:59:04 (90.7 MB/s) - 'amazon-corretto-11-x64-linux-jdk.deb' saved [195577844/195577844]

[ec2-user@ip-172-31-5-134 ~]$
```

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 31°C Haze ENG 10:59 PM IN 10/27/2024

Successfully download JDK using Amazon -linux

```
ubuntu@ip-172-31-15-103:~$ sudo apt install openjdk-8-jdk
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
alsa-icon-theme alsamixer-conf at-spi2-common at-spi2-core ca-certificates-java dconf-gsettings-backend
dconf-service fontconfig fontconfig-config fonts-dejavu-core fonts-dejavu-extra fonts-dejavu-mono gsettings-desktop-schemas
gtk-update-icon-cache hicolor-icon-theme humanity-icon-theme java-common libasound2-data libasound2t64 libasynccns0
libatk-bridge2.0-0t64 libatk-wrapper-java libatk-wrapper-java-jni libatk1.0-0t64 libatspi2.0-0t64 libavahi-client3
libavahi-common-data libavahi-common3 libcairo-gobject3 libcairo2 libcurl3t64 libdatrie libdconf1 libdeflate0 libdrm-amdgpu
libdrm-intel libdrm-nouveau2 libdrm-radeon1 libflac12t64 libfontconfig1 libgail-common libgail11t64 libgd-pixbuf-2.0-0
libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgif7 libgl1 libgl1-amber-dri libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0
libglx0 libgrahpite2-3 libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libharfbuzz0 libice-dev libice6 libjbig0 libjpeg-turbo8
libjpegt libicms2-2 liblerc4 liblwm17t64 libmp3lame0 libmpg123-0t64 libogg libopus0 libpango-1.0-0 libpangocairo-1.0-0
libpangoft2-1.0-0 libpciaccess0 libpcslite1 libpixman-1-0 libpthread-stubs0-dev libpulse0 librsvg2-2 librsvg2-common libsharpuy0
libbam-dev libsm6 libsndfile1 libthai-data libthai0 libtiff6 libvorbis0a libvorbisenc2 libvulkan1 libwayland-client0 libwevp7
libxml-dev libxml1 libxau-dev libxaw7 libxcb-dri2-0 libxcb-glx0 libxcb-present0 libxcb-randr0 libxcb-render0
libxcb-shape0 libxcb-shm0 libxcb-sync1 libxcb-xfixes0 libxcb1-dev libxcompositel libxcursor1 libxdamage1 libxdmcp-dev libxfixes3
libxft2 libx16 libxinerama1 libxkbfile1 libxmu6 libxpmp4 libxrandr2 libxrender1 libxshmfence1 libxt-dev libxt6t64 libxtst6 libxv1
libxf86gal libxxfb6vml mesa-vulkan-drivers openjdk-8-jdk-headless openjdk-8-jre openjdk-8-jre-headless session-migration
ubuntu-mono x11-common x11-utils x11proto-dev xorg-sgml-doctools xtrans-dev
```

i-0095ed1973f4519d5 (new-1)
PublicIPs: 65.1.130.31 PrivateIPs: 172.31.15.103

Successfully download JDK using Ubuntu -linux

4 . Lauch Ec2 - ubuntu Instance & Install maven

```
[ec2-user@ip-172-31-5-134 ~]$ sudo yum install maven -y
Last metadata expiration check: 0:03:09 ago on Sat Oct 26 09:11:15 2024.
Dependencies resolved.

=====
 Package          Architecture Version      Repository   Size
 e
=====
Installing:
 maven           noarch      1:3.8.4-3.amzn2023.0.5    amazonlinux  18
 k
Installing dependencies:
 alsa-lib         x86_64      1.2.7.2-1.amzn2023.0.2    amazonlinux  504
 k
 apache-commons-cli x86_64      1.5.0-3.amzn2023.0.3    amazonlinux  76
 k
 apache-commons-codec x86_64      1.15-6.amzn2023.0.3    amazonlinux  303
 k
 apache-commons-io  noarch      1:2.8.0-7.amzn2023.0.4    amazonlinux  284
 k
 apache-commons-lang3 x86_64      3.12.0-7.amzn2023.0.3    amazonlinux  559
 k
 atinject         noarch      1.0.5-3.amzn2023.0.3    amazonlinux  23
 k
 cairo            x86_64      1.17.6-2.amzn2023.0.1    amazonlinux  684
 k

CloudShell Feedback
```

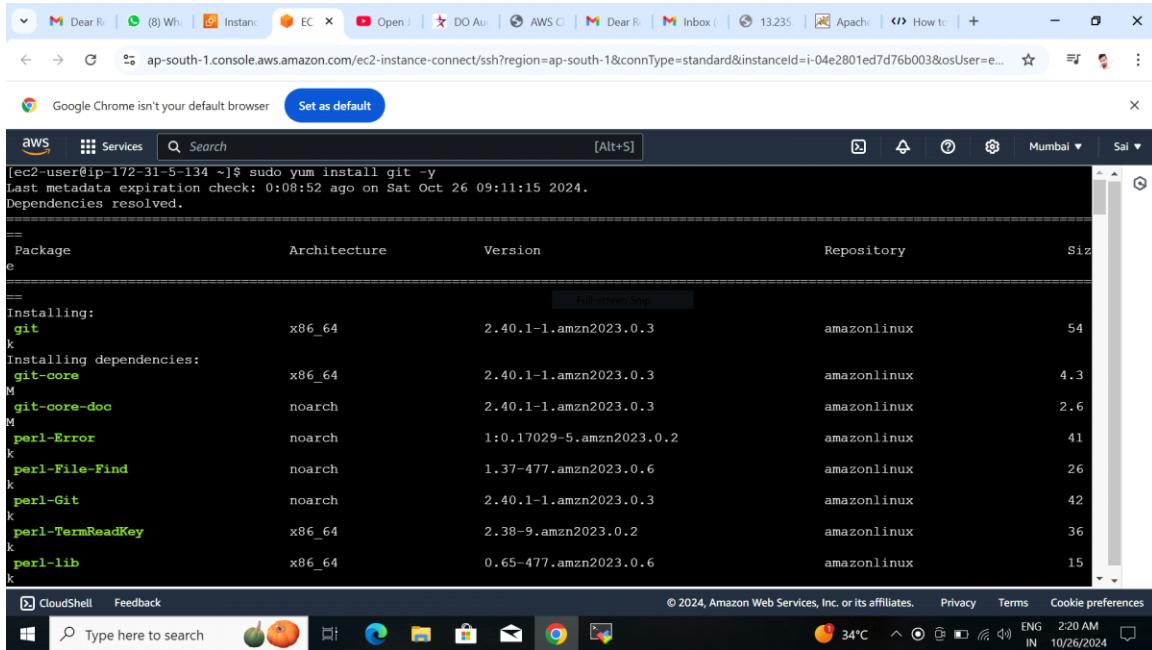
Successfully install maven using Amazon - linux

```
ubuntu@ip-172-31-15-103:~$ sudo apt install maven
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
 libapache-pom-java libatbatinjection-jsr330-api-java libcdi-api-java libcommons-cli-java libcommons-io-java
 libcommons-lang3-java libcommons-parent-java liberror-prone-java liberonimo-annotation-1.3-spec-java
 liberonimo-interceptor-3.0-spec-java libguava-java libguice-java libjansi-java libjsr305-java libmaven-parent-java
 libmaven-resolver-java libmaven-shared-utils-java libmaven3-core-java libplexus-cipher-java libplexus-classworlds-java
 libplexus-component-annotations-java libplexus-interpolation-java libplexus-sec-dispatcher-java libplexus-util2-java
 libsisu-inject-java libsisu-plexus-java libsisu4j-java libwagon-file-java libwagon-http-shaded-java libwagon-provider-api-java
Suggested packages:
 libatinject-jsr330-api-java-doc libel-api-java libcommons-io-java-doc libasm-java libcglib-java libjsr305-java-doc
 libmaven-shared-utils-java-doc liblogback-java libplexus-utils2-java-doc junit4 testing libcommons-logging-java liblog4j1.2-java
The following NEW packages will be installed:
 libapache-pom-java libatbatinjection-jsr330-api-java libcdi-api-java libcommons-cli-java libcommons-io-java
 libcommons-lang3-java libcommons-parent-java liberror-prone-java liberonimo-annotation-1.3-spec-java
 liberonimo-interceptor-3.0-spec-java libguava-java libguice-java libjansi-java libjsr305-java libmaven-parent-java
 libmaven-resolver-java libmaven-shared-utils-java libmaven3-core-java libplexus-cipher-java libplexus-classworlds-java
 libplexus-component-annotations-java libplexus-interpolation-java libplexus-sec-dispatcher-java libplexus-util2-java
 libsisu-inject-java libsisu-plexus-java libsisu4j-java libwagon-file-java libwagon-http-shaded-java libwagon-provider-api-java maven

i-0095ed1973f4519d5 (new-1)
PublicIPs: 65.1.130.31 PrivateIPs: 172.31.15.103
```

Successfully install maven using Ubuntu - linux

5. Launch Ec2 - ubuntu Instance & Install Git

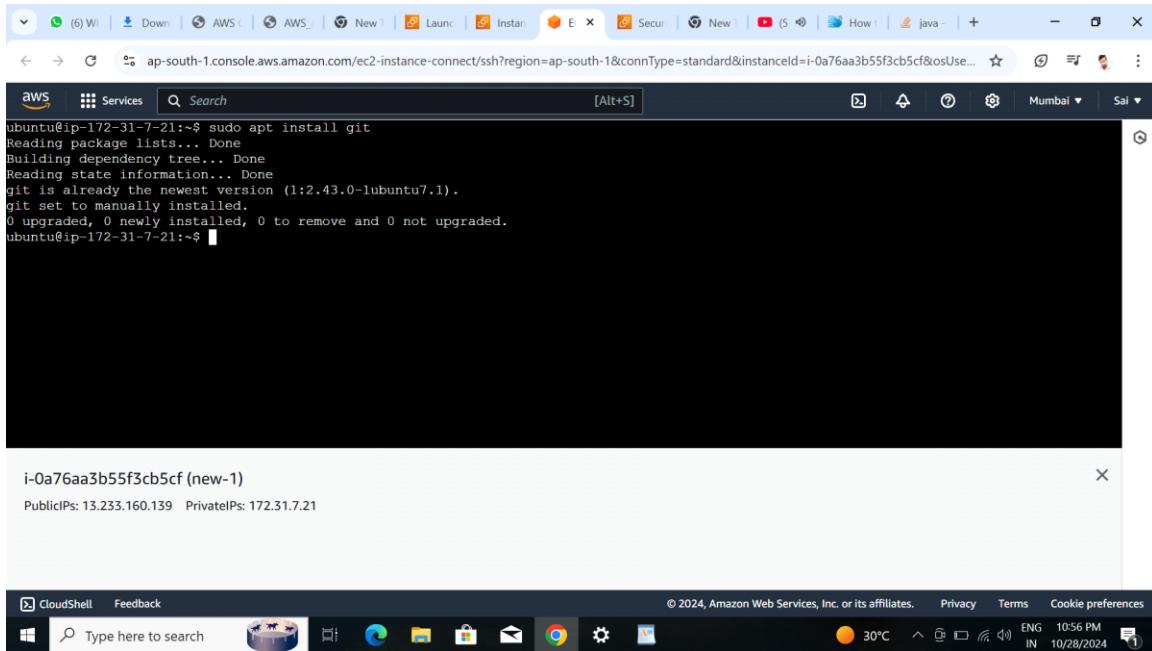


The screenshot shows a Google Chrome window with a terminal session running on an Amazon Linux EC2 instance. The terminal output is as follows:

```
[ec2-user@ip-172-31-5-134 ~]$ sudo yum install git -y
Last metadata expiration check: 0:08:52 ago on Sat Oct 26 09:11:15 2024.
Dependencies resolved.
=====
== Package           Architecture      Version       Repository   Size
e
=====
Installing:
git                  x86_64          2.40.1-1.amzn2023.0.3    amazonlinux  54
k
Installing dependencies:
git-core              x86_64          2.40.1-1.amzn2023.0.3    amazonlinux  4.3
M
git-core-doc          noarch          2.40.1-1.amzn2023.0.3    amazonlinux  2.6
M
perl-Error            noarch          1:0.17029-5.amzn2023.0.2  amazonlinux  41
k
perl-File-Find        noarch          1.37-477.amzn2023.0.6    amazonlinux  26
k
perl-Git               noarch          2.40.1-1.amzn2023.0.3    amazonlinux  42
k
perl-TermReadKey     x86_64          2.38-9.amzn2023.0.2     amazonlinux  36
k
perl-lib               x86_64          0.65-477.amzn2023.0.6    amazonlinux  15
k
```

The terminal prompt shows the user is root (ec2-user) on an IP address 172.31.5.134. The command run was `sudo yum install git -y`. The output indicates that the package `git` and its dependencies were successfully installed from the `amazonlinux` repository.

Successfully install Git using Amazon - linux

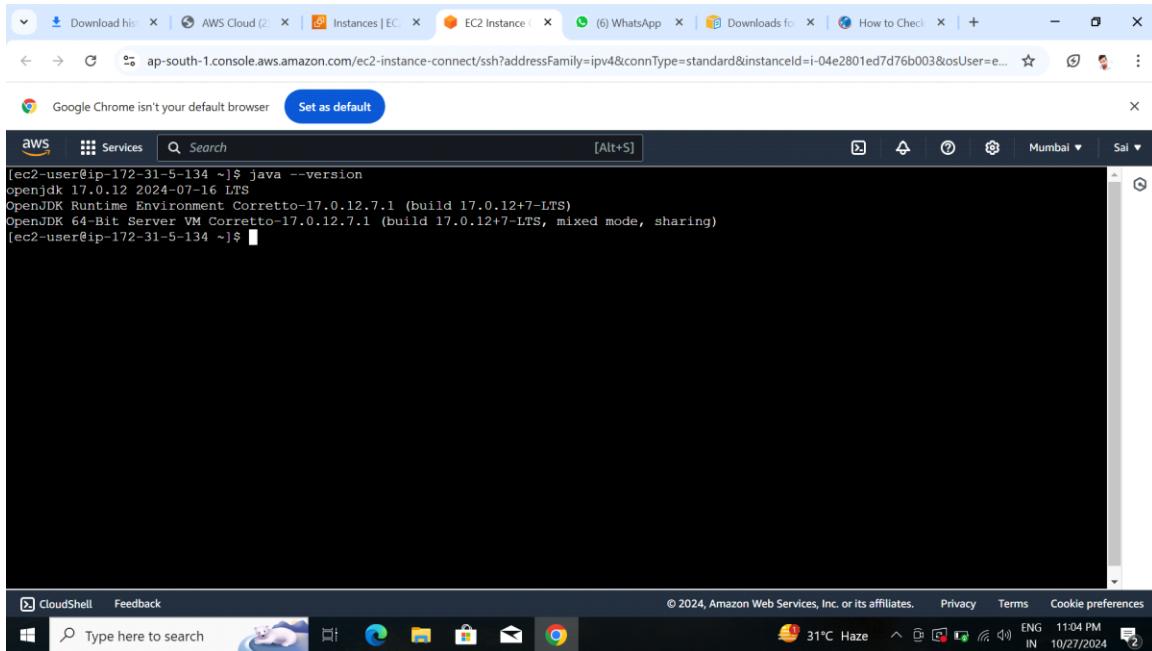


```
ubuntu@ip-172-31-7-21:~$ sudo apt install git
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.43.0-1ubuntu7.1).
git set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-7-21:~$
```

i-0a76aa3b55f3cb5cf (new-1)
PublicIPs: 13.233.160.139 PrivateIPs: 172.31.7.21

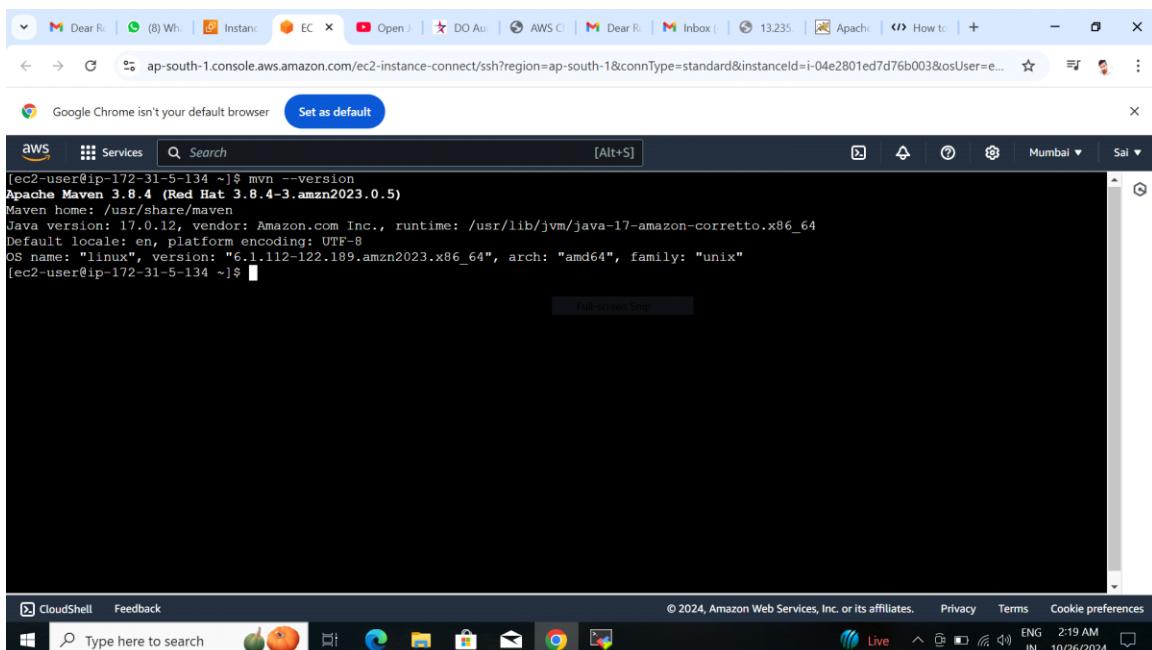
Successfully install Git using Ubuntu - linux

6 . Launch Ec2 - ubuntu Instance & validate the versions



```
[ec2-user@ip-172-31-5-134 ~]$ java --version
openjdk 17.0.12 2024-07-16 LTS
OpenJDK Runtime Environment Corretto-17.0.12.7.1 (build 17.0.12+7-LTS)
OpenJDK 64-Bit Server VM Corretto-17.0.12.7.1 (build 17.0.12+7-LTS, mixed mode, sharing)
[ec2-user@ip-172-31-5-134 ~]$
```

Successfully validate Java(Jdk) version



```
[ec2-user@ip-172-31-5-134 ~]$ mvn --version
Apache Maven 3.8.4 (Red Hat 3.8.4-3.amzn2023.0.5)
Maven home: /usr/share/maven
Java version: 17.0.12, vendor: Amazon.com Inc., runtime: /usr/lib/jvm/java-17-amazon-corretto.x86_64
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "6.1.112-122.189.amzn2023.x86_64", arch: "amd64", family: "unix"
[ec2-user@ip-172-31-5-134 ~]$
```

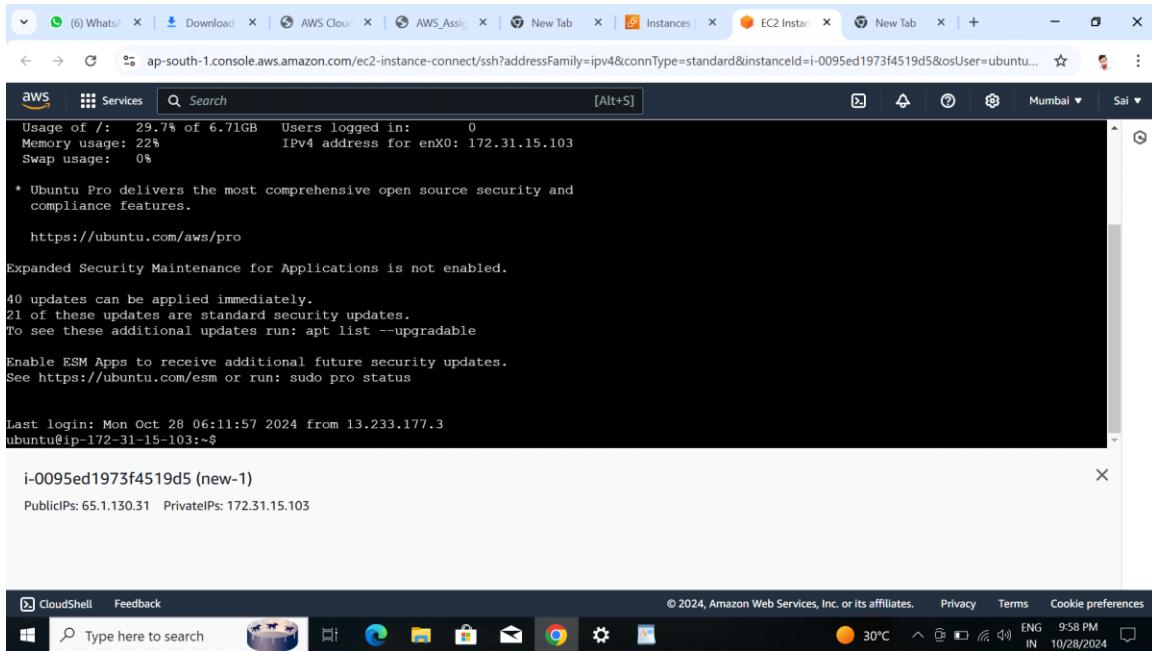
Successfully validate maven version

```
[ec2-user@ip-172-31-5-134 ~]$ ls
[ec2-user@ip-172-31-5-134 ~]$ sudo yum install git -y
Last metadata expiration check: 0:11:58 ago on Sat Oct 26 09:11:15 2024.
Package git-2.40.1-1.amzn2023.0.3.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-5-134 ~]$ git --version
git version 2.40.1
[ec2-user@ip-172-31-5-134 ~]$
```

Successfully validate Git version

5. L5 - Install Tomcat web application server in AWS EC2 Ubuntu Instance and access Tomcat using a web browser

1. Lauch Ec2 - ubuntu Instance



```
Usage of /: 29.7% of 6.71GB  Users logged in: 0
Memory usage: 22%          IPv4 address for enX0: 172.31.15.103
Swap usage:  0%
* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.
https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

40 updates can be applied immediately.
21 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Mon Oct 28 06:11:57 2024 from 13.233.177.3
ubuntu@ip-172-31-15-103:~$
```

i-0095ed1973f4519d5 (new-1)
PublicIPs: 65.1.130.31 PrivateIPs: 172.31.15.103

Successfully launch Ec2 - Ubuntu Instance

2. Install Tomcat web application server in AWS EC2 Ubuntu Instance

```
[ec2-user@ip-172-31-13-54 ~]$ wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.96/bin/apache-tomcat-9.0.96.tar.gz
--2024-10-26 08:33:16-- https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.96/bin/apache-tomcat-9.0.96.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org) ... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 12716365 (12M) [application/x-gzip]
Saving to: 'apache-tomcat-9.0.96.tar.gz'

apache-tomcat-9.0.96.tar.gz      0%
apache-tomcat-9.0.96.tar.gz    100%[=====] 12.13M --.KB/s   in 0.1s

2024-10-26 08:33:16 (88.5 MB/s) - 'apache-tomcat-9.0.96.tar.gz' saved [12716365/12716365]

[ec2-user@ip-172-31-13-54 ~]$
```

Successfully Install Tomcat web application server in Ec2 instance

3. Extract the exist tomcate web application file

The screenshot shows a terminal window in an AWS CloudShell session. The user has run the command `tar -xvzf apache-tomcat-9.0.96.tar.gz`. The output lists numerous files and directories extracted from the archive, including configuration files like `catalina.policy`, `catalina.properties`, and `context.xml`, as well as web-related files like `index.html` and `web.xml`. The terminal interface includes standard AWS navigation buttons at the top and a taskbar at the bottom with icons for CloudShell, Feedback, and various services.

```
aws Services Search [Alt+S] Mumbai Sai
apache-tomcat-9.0.96.tar.gz
[ec2-user@ip-172-31-13-54 ~]$ tar -xvzf ^C
[ec2-user@ip-172-31-13-54 ~]$ tar -xvzf apache-tomcat-9.0.96.tar.gz
apache-tomcat-9.0.96/conf/
apache-tomcat-9.0.96/conf/catalina.policy
apache-tomcat-9.0.96/conf/catalina.properties
apache-tomcat-9.0.96/conf/context.xml
apache-tomcat-9.0.96/conf/jaspic-providers.xml
apache-tomcat-9.0.96/conf/jaspic-providers.xsd
apache-tomcat-9.0.96/conf/logging.properties
apache-tomcat-9.0.96/conf/server.xml
apache-tomcat-9.0.96/conf/tomcat-users.xml
apache-tomcat-9.0.96/conf/tomcat-users.xsd
apache-tomcat-9.0.96/conf/web.xml
apache-tomcat-9.0.96/bin/
apache-tomcat-9.0.96/lib/
apache-tomcat-9.0.96/logs/
apache-tomcat-9.0.96/temp/
apache-tomcat-9.0.96/webapps/
apache-tomcat-9.0.96/webapps/ROOT/
apache-tomcat-9.0.96/webapps/ROOT/WEB-INF/
apache-tomcat-9.0.96/webapps/docs/
apache-tomcat-9.0.96/webapps/docs/META-INF/
apache-tomcat-9.0.96/webapps/docs/WEB-INF/
apache-tomcat-9.0.96/webapps/docs/WEB-INF/jsp/
apache-tomcat-9.0.96/webapps/docs/annotationapi/
apache-tomcat-9.0.96/webapps/docs/api/
```

Successfully extract the exist file

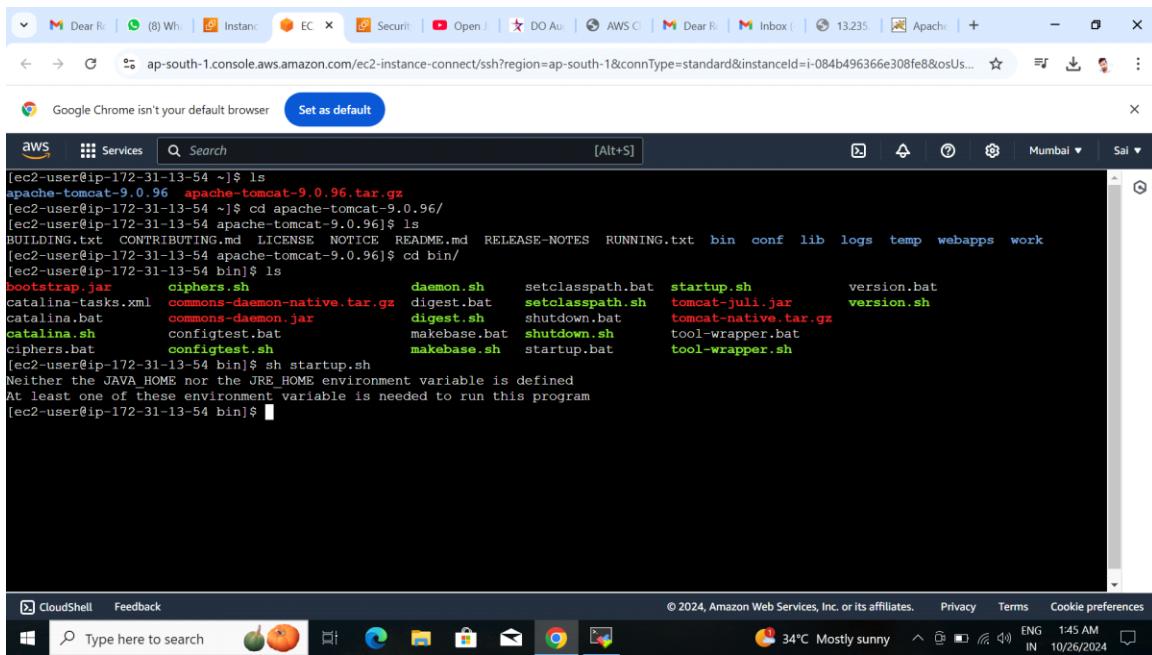
4. After exist file we get new file

The screenshot shows a terminal window in an AWS CloudShell session. The user has run the command `ls` to list the contents of the directory. The output shows a single file named `apache-tomcat-9.0.96.tar.gz`, which is highlighted in red. The terminal interface includes standard AWS navigation buttons at the top and a taskbar at the bottom with icons for CloudShell, Feedback, and various services.

```
aws Services Search [Alt+S] Mumbai Sai
[ec2-user@ip-172-31-13-54 ~]$ ls
apache-tomcat-9.0.96 apache-tomcat-9.0.96.tar.gz
[ec2-user@ip-172-31-13-54 ~]$
```

apache-tomcat-9.0.96

5. Start apache Tomcat web application server

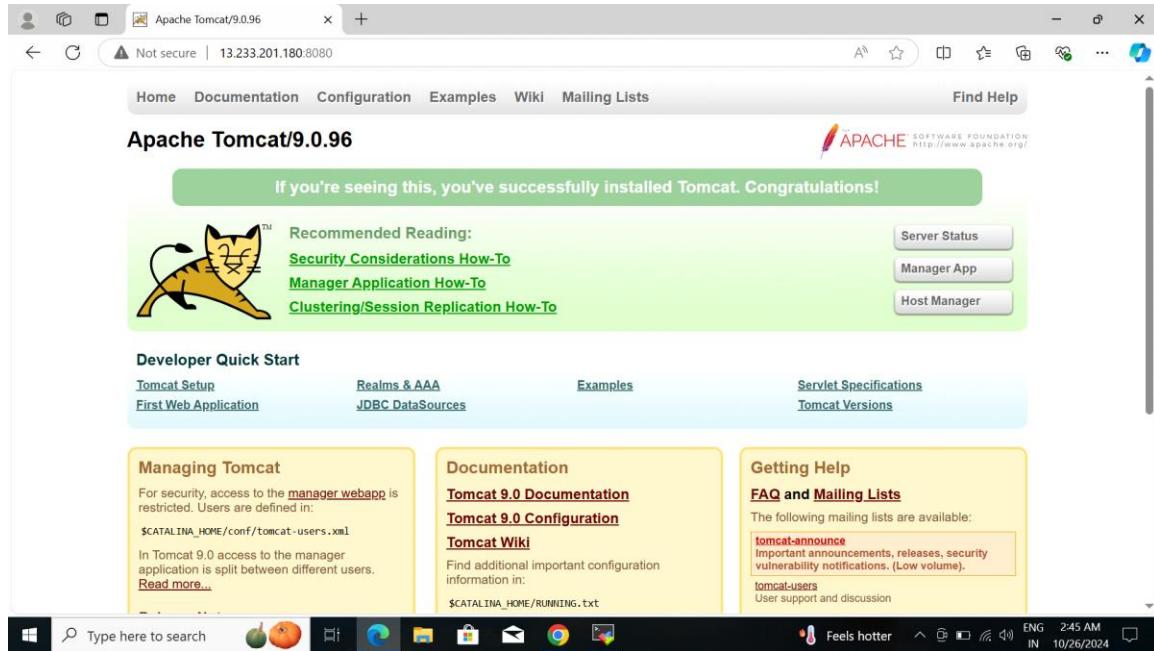


The screenshot shows a terminal window within an AWS CloudShell interface. The user has navigated to the directory containing the Tomcat 9.0.96 distribution files. They run the command `sh startup.sh`. The terminal output shows several environment variables being set, such as `JAVA_HOME` and `JRE_HOME`, and then the Tomcat service starts successfully, indicating it is now running.

```
[ec2-user@ip-172-31-13-54 ~]$ ls
apache-tomcat-9.0.96 apache-tomcat-9.0.96.tar.gz
[ec2-user@ip-172-31-13-54 ~]$ cd apache-tomcat-9.0.96/
[ec2-user@ip-172-31-13-54 apache-tomcat-9.0.96]$ ls
BUILDING.txt CONTRIBUTING.md LICENSE NOTICE README.md RELEASE-NOTES RUNNING.txt bin conf lib logs temp webapps work
[ec2-user@ip-172-31-13-54 apache-tomcat-9.0.96]$ cd bin/
[ec2-user@ip-172-31-13-54 bin]$ ls
bootstrap.jar ciphers.sh daemon.sh setclasspath.bat startup.sh version.bat
catalina-tasks.xml commons-daemon-native.tar.gz digest.bat setclasspath.sh tomcat-juli.jar version.sh
catalina.bat commons-daemon.jar digest.sh shutdown.bat tomcat-native.tar.gz
catalina.sh configtest.bat makebase.bat shutdown.sh tool-wrapper.bat
ciphers.bat configtest.sh makebase.sh startup.bat tool-wrapper.sh
[ec2-user@ip-172-31-13-54 bin]$ sh startup.sh
Neither the JAVA_HOME nor the JRE_HOME environment variable is defined
At least one of these environment variable is needed to run this program
[ec2-user@ip-172-31-13-54 bin]$
```

Successfully start tomcat-web-application server

6. Access Tomcat using a web browser with public ip & To allow SG permission are 8080 port



Successfully Access Tomcat using a web browser

6. L6 - Create S3 Bucket and add folders and files

1. Login into S3 service dashborad

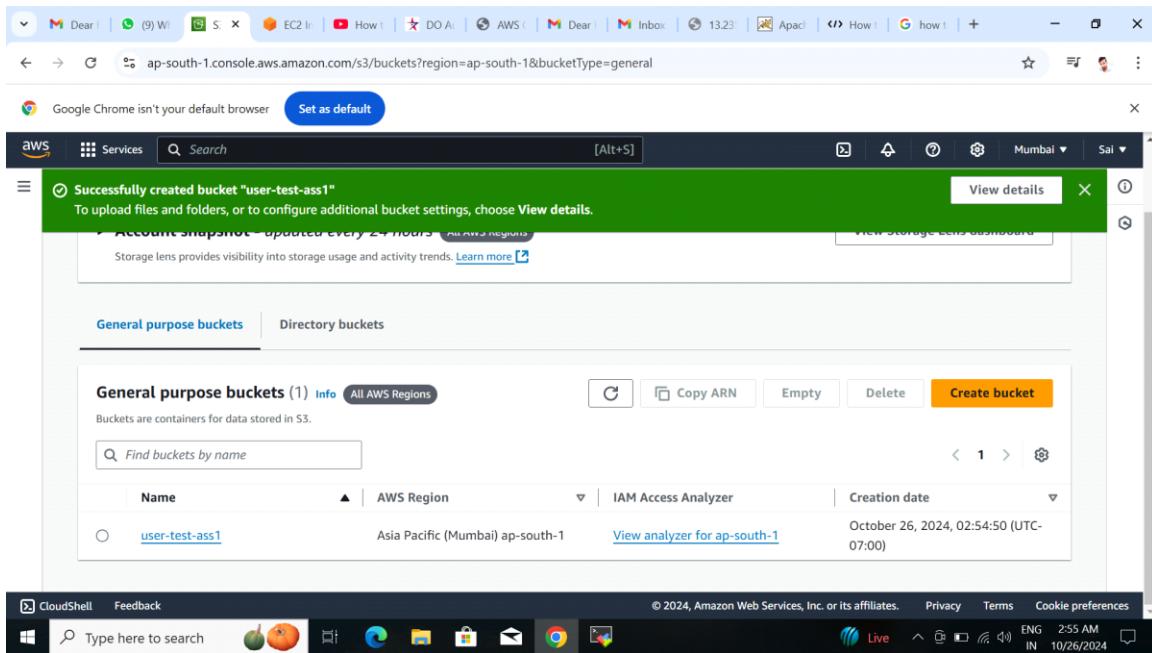
The screenshot shows the AWS S3 service page. At the top right, there is a modal window titled "Create a bucket". The modal contains the text: "Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored." Below this is a prominent orange "Create bucket" button. The main S3 page features the heading "Amazon S3" and the sub-headline "Store and retrieve any amount of data from anywhere". A descriptive paragraph below states: "Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance." On the left, there's a sidebar with "Storage" and a "How it works" section. The browser toolbar at the bottom includes icons for CloudShell, Feedback, and various system functions like Live, battery, and network.

2. Create a S3 bucket with unique name

The screenshot shows the "Create bucket" configuration page. The title bar says "Create bucket" and "Info". Below it, a sub-header states: "Buckets are containers for data stored in S3." The main section is titled "General configuration". It includes fields for "AWS Region" (set to "Asia Pacific (Mumbai) ap-south-1") and "Bucket name" (set to "user-test-ass1"). A note below the bucket name field says: "Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)". There is also a "Copy settings from existing bucket - optional" section with a "Choose bucket" button and a note: "Only the bucket settings in the following configuration are copied." The browser toolbar at the bottom is identical to the one in the previous screenshot.

Bucket name : user-test-ass1

3. Created S3 bucket with required configuration like(enable versioning)



The screenshot shows the AWS S3 console interface. At the top, there is a green success message: "Successfully created bucket 'user-test-ass1'. To upload files and folders, or to configure additional bucket settings, choose View details." Below this, there is a section titled "Account snapshot - updated every 24 hours" with a link to "View Storage Lens dashboard". The main area displays "General purpose buckets" with one item listed: "user-test-ass1" (Asia Pacific (Mumbai) ap-south-1). The creation date is "October 26, 2024, 02:54:50 (UTC-07:00)". Action buttons for the bucket include "Copy ARN", "Empty", "Delete", and "Create bucket". The bottom of the screen shows the Windows taskbar with various icons and the system tray.

Successfully created bucket

4. add folders and file in created S3 bucket with enabling versioning for retrieve the object

Objects (0) [Info](#)

No objects
You don't have any objects in this bucket.

[Upload](#)



- object dashboard for upload file and also create folders here

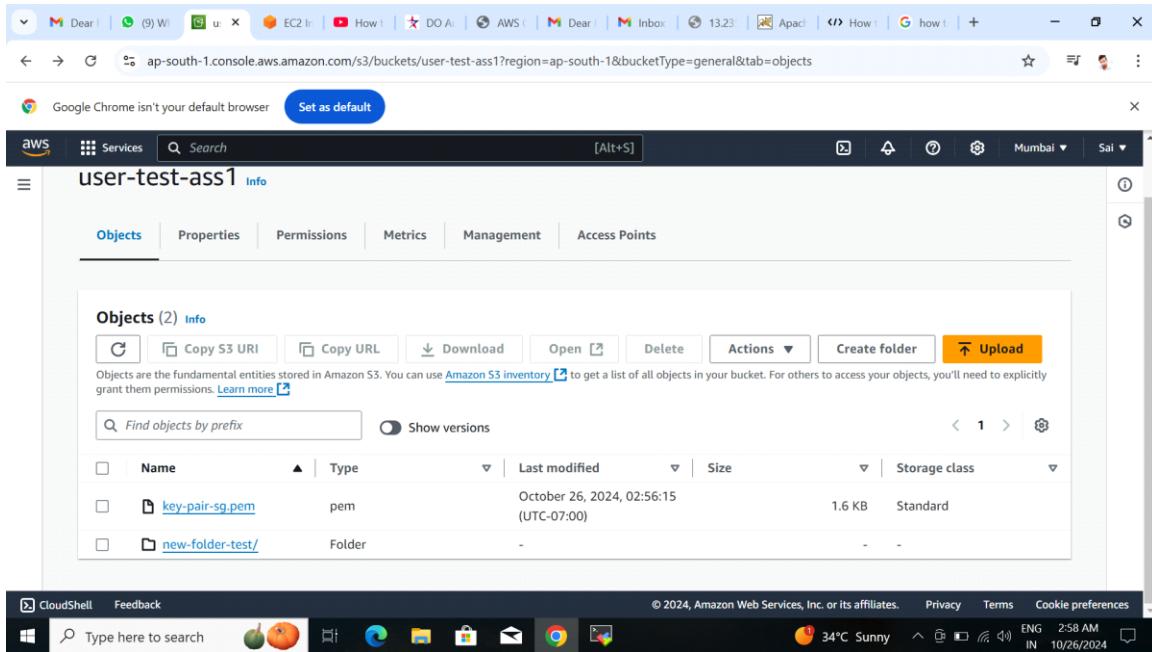
Destination	Succeeded	Failed
s3://user-test-ass1/new-folder-test/	1 file, 1.6 KB (100.00%)	0 files, 0 B (0%)

Files and folders (1 Total, 1.6 KB)

Name	Folder	Type	Size	Status	Error
key-pair-sg.p...	-	-	1.6 KB	Succeeded	-



Successfully add files in S3 Bucket



- Successfully add folders and files in bucket with secure versioning for retrieve data
- if we gave temporary access to some we use presigned URL for that

THANK YOU

