

DEVEOPS CAPSTONE : PROJECT II

You are hired as a DevOps Engineer for Analytics Pvt Ltd. This company is a product based organization which uses Docker for their containerization needs within the company. The final product received a lot of traction in the first few weeks of launch. Now with the increasing demand, the organization needs to have a platform for automating deployment, scaling and operations of application containers across clusters of hosts. As a DevOps Engineer, you need to implement a DevOps lifecycle such that all the requirements are implemented without any change in the Docker containers in the testing environment.

Up until now, this organization used to follow a monolithic architecture with just 2 developers. The product is present on: <https://github.com/hshar/website.git>

Following are the specifications of the lifecycle:

1. Git workflow should be implemented. Since the company follows a monolithic architecture of development, you need to take care of version control. The release should happen only on the 25th of every month.
2. CodeBuild should be triggered once the commits are made in the master branch.
3. The code should be containerized with the help of the Dockerfile. The Dockerfile should be built every time if there is a push to GitHub. Create a custom Docker image using a Dockerfile.
4. As per the requirement in the production server, you need to use the Kubernetes cluster and the containerized code from Docker Hub should be deployed with 2 replicas. Create a NodePort service and configure the same for port 30008.
5. Create a Jenkins Pipeline script to accomplish the above task.
6. For configuration management of the infrastructure, you need to deploy the configuration on the servers to install necessary software and configurations.
7. Using Terraform, accomplish the task of infrastructure creation in the AWS cloud provider.

Architectural Advice

Architectural Advice:

Softwares to be installed on the respective machines using configuration management.

Worker1: Jenkins, Java

Contact us: support@intellipaat.com / © Copyright Intellipaat / All rights reserved

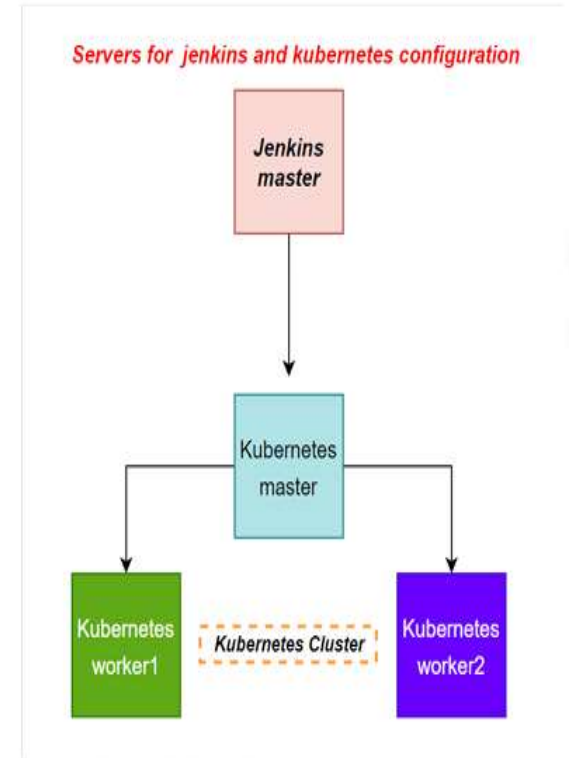
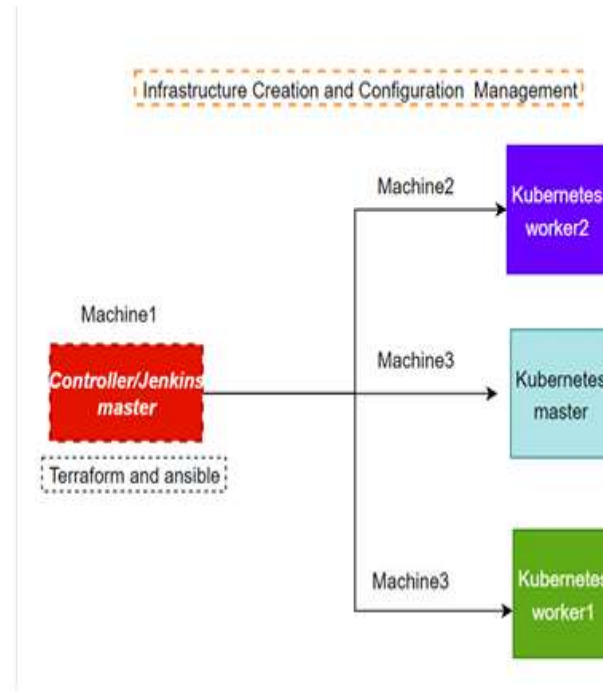
DevOps Certification Training



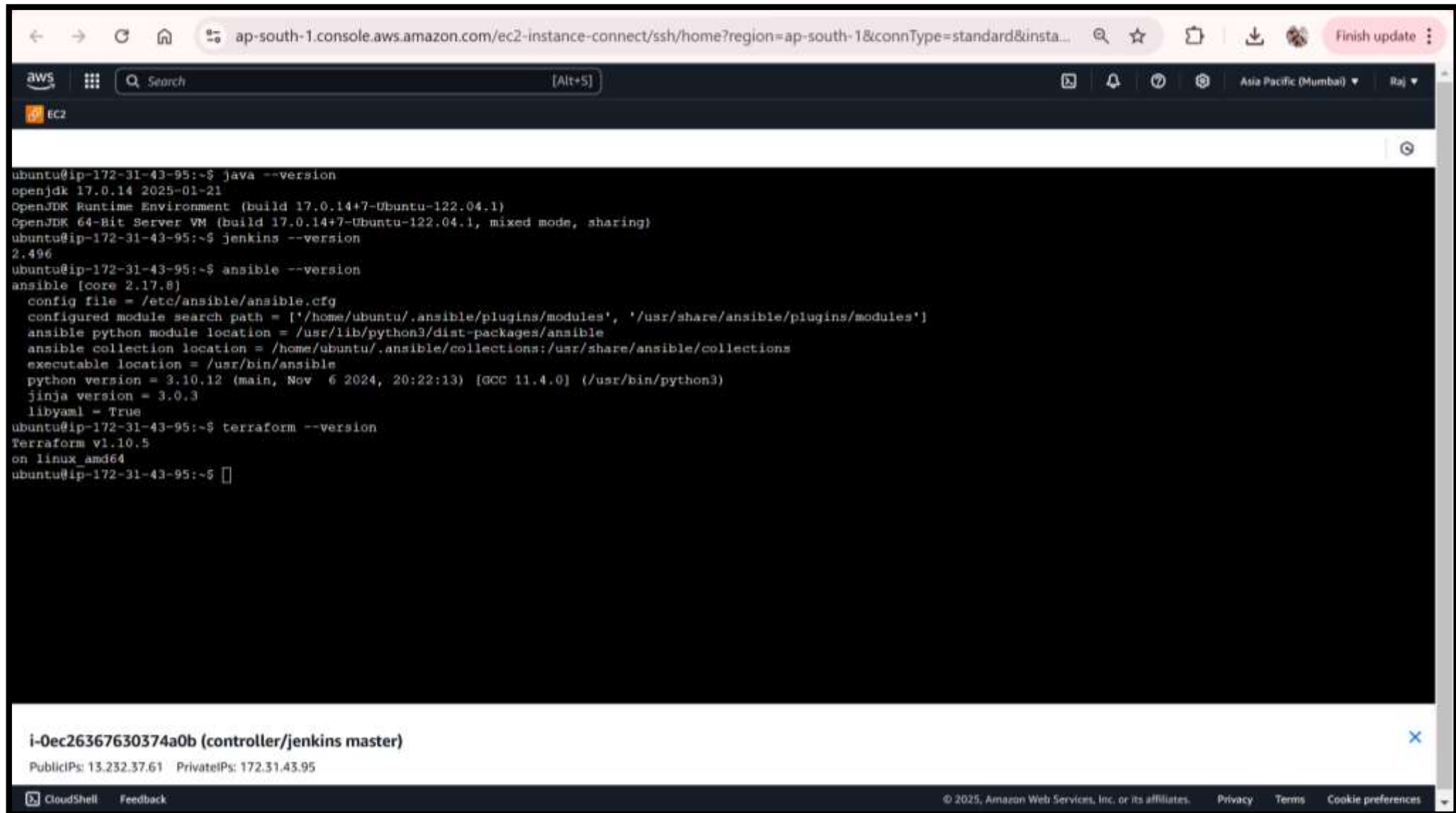
Worker2: Docker, Kubernetes

Worker3: Java, Docker, Kubernetes

Worker4: Docker, Kubernetes



In Master-Jenkins Instance Jenkins, Java, Ansible, Terraform Install



The screenshot shows an AWS CloudShell terminal window with the following content:

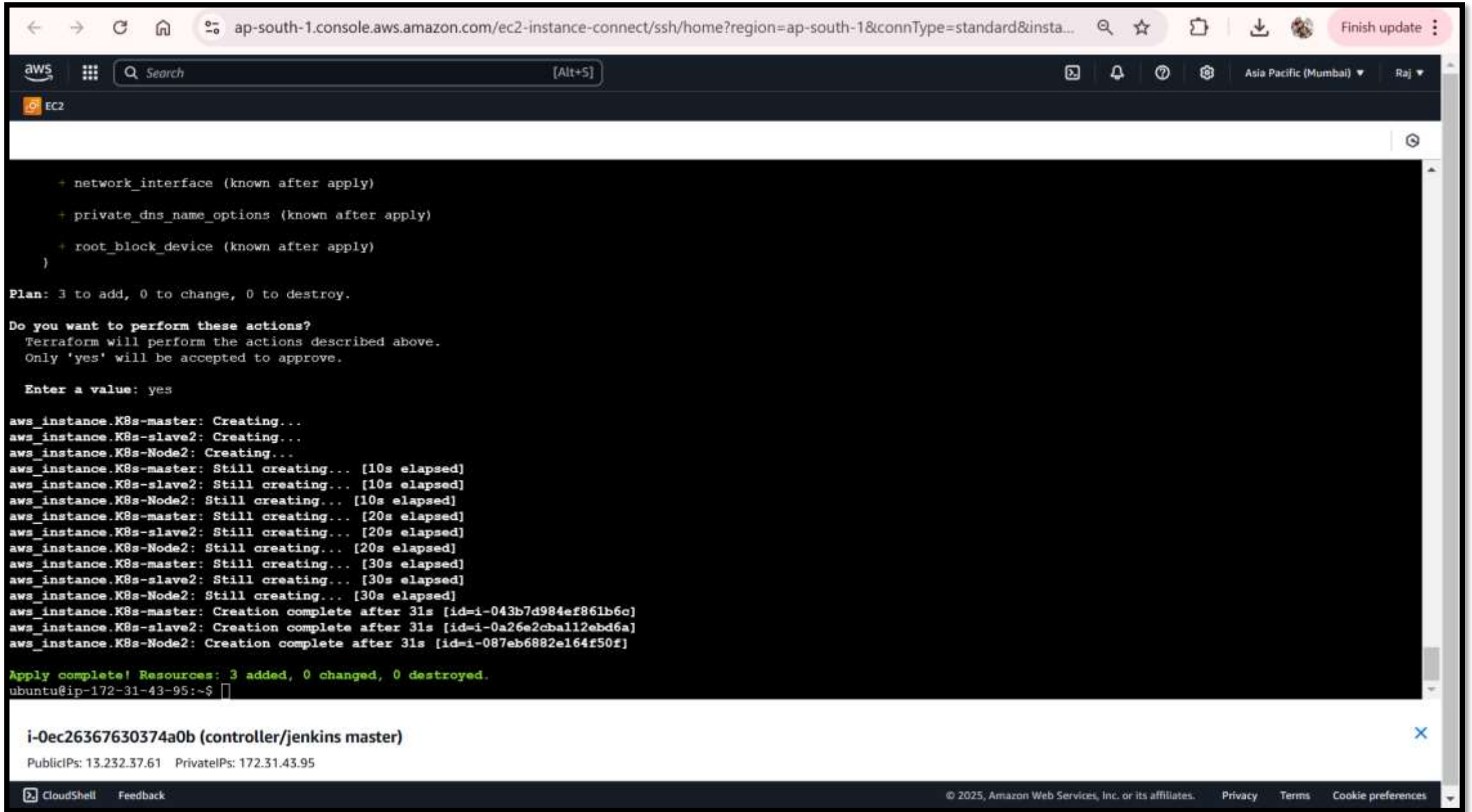
```
ubuntu@ip-172-31-43-95:~$ java --version
openjdk 17.0.14 2025-01-21
OpenJDK Runtime Environment (build 17.0.14+7-Ubuntu-122.04.1)
OpenJDK 64-Bit Server VM (build 17.0.14+7-Ubuntu-122.04.1, mixed mode, sharing)
ubuntu@ip-172-31-43-95:~$ jenkins --version
2.496
ubuntu@ip-172-31-43-95:~$ ansible --version
ansible [core 2.17.8]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.10.12 (main, Nov  6 2024, 20:22:13) [GCC 11.4.0] (/usr/bin/python3)
  jinja version = 3.0.3
  libyaml = True
ubuntu@ip-172-31-43-95:~$ terraform --version
Terraform v1.10.5
on linux amd64
ubuntu@ip-172-31-43-95:~$
```

At the bottom of the terminal window, there is a summary box for the instance **i-0ec26367630374a0b (controller/jenkins master)** with the following details:

- Public IPs: 13.232.37.61
- Private IPs: 172.31.43.95

The footer of the CloudShell window includes the text "© 2025, Amazon Web Services, Inc. or its affiliates." and links for "Privacy", "Terms", and "Cookie preferences".

Terraform apply



```
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&insta...
aws
[Alt+S]
EC2

+ network_interface (known after apply)
+ private_dns_name_options (known after apply)
+ root_block_device (known after apply)
)

Plan: 3 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_instance.K8s-master: Creating...
aws_instance.K8s-slave2: Creating...
aws_instance.K8s-Node2: Creating...
aws_instance.K8s-master: Still creating... [10s elapsed]
aws_instance.K8s-slave2: Still creating... [10s elapsed]
aws_instance.K8s-Node2: Still creating... [10s elapsed]
aws_instance.K8s-master: Still creating... [20s elapsed]
aws_instance.K8s-slave2: Still creating... [20s elapsed]
aws_instance.K8s-Node2: Still creating... [20s elapsed]
aws_instance.K8s-master: Still creating... [30s elapsed]
aws_instance.K8s-slave2: Still creating... [30s elapsed]
aws_instance.K8s-Node2: Still creating... [30s elapsed]
aws_instance.K8s-master: Creation complete after 31s [id=i-043b7d984ef861b6c]
aws_instance.K8s-slave2: Creation complete after 31s [id=i-0a26e2c112ebd6a]
aws_instance.K8s-Node2: Creation complete after 31s [id=i-087eb6882e164f50f]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-43-95:~$
```

i-0ec26367630374a0b (controller/jenkins master)

PublicIPs: 13.232.37.61 PrivateIPs: 172.31.43.95

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Master & worker Machine

The screenshot displays the AWS Management Console for the 'ap-south-1' region. The left sidebar shows the navigation menu with categories like EC2, Images, Elastic Block Store, and Network & Security. The main content area shows a list of EC2 instances with the filter 'instance state = running'. The instance 'controller/jenkins master' is selected, and its details are shown below the list.

Instances (1/4) Info

Find Instance by attribute or tag (case-sensitive) All states

instance state = running Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
controller/jenkins master	i-0ec26367630374a0b	Running	t2.medium	2/2 checks passed	View alarms +	ap-south
m3-master	i-043b7d984ef861b6c	Running	t2.medium	Initializing	View alarms +	ap-south
m2-slave	i-0a26e2cba112ebd6a	Running	t2.medium	Initializing	View alarms +	ap-south
m4-slave	i-087eb6882e164f50f	Running	t2.medium	Initializing	View alarms +	ap-south

i-0ec26367630374a0b (controller/jenkins master)

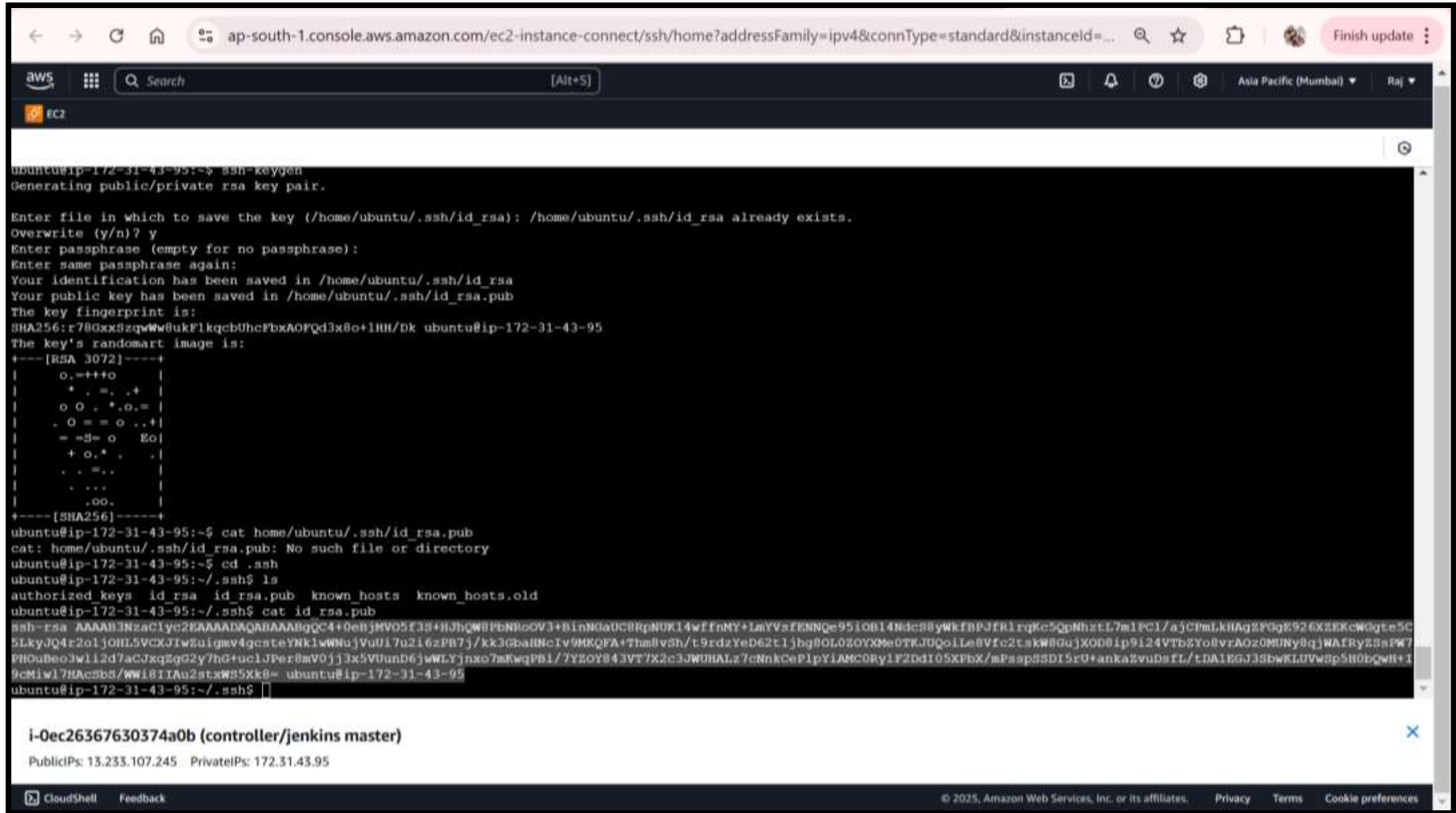
Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary Info

Instance ID i-0ec26367630374a0b	Public IPv4 address 13.232.37.61 open address	Private IPv4 addresses 172.31.43.95
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-13-232-37-61.ap-south-1.compute.amazonaws.com open address
Hostname type IP name: ip-172-31-43-95.ap-south-1.compute.internal	Private IP DNS name (IPv4 only) ip-172-31-43-95.ap-south-1.compute.internal	Elastic IP addresses
Answer private resource DNS name	Instance type	

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Generating Ssh public key



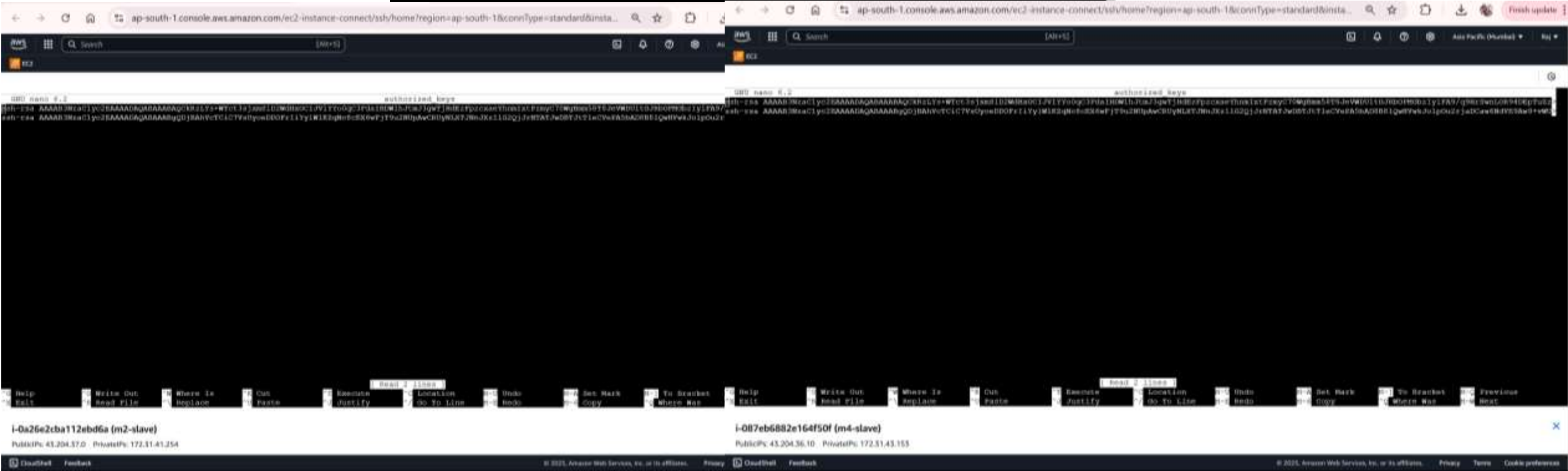
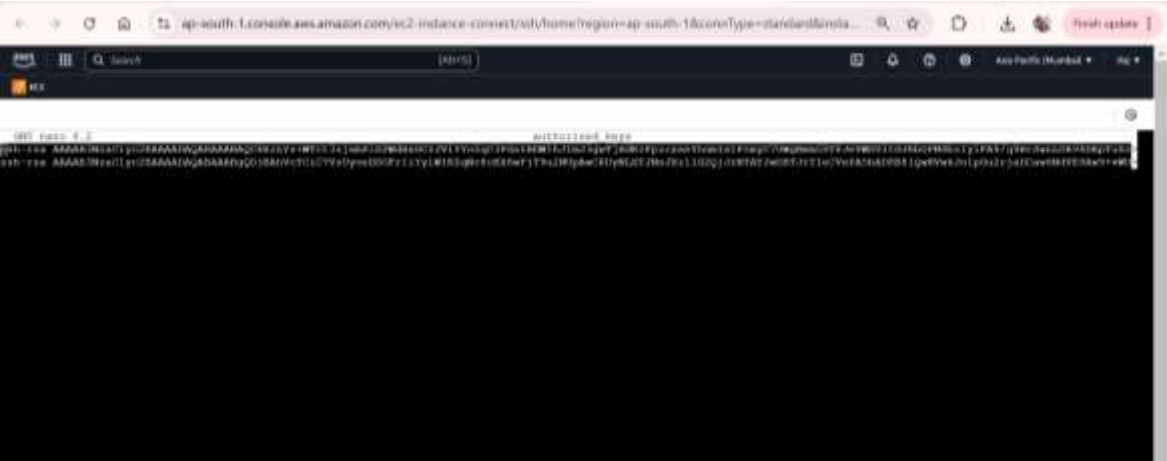
The screenshot shows the AWS CloudShell interface. The terminal window displays the following commands and output:

```
ubuntu@ip-172-31-43-95:~$ ssh-keygen
Generating public/private rsa key pair.

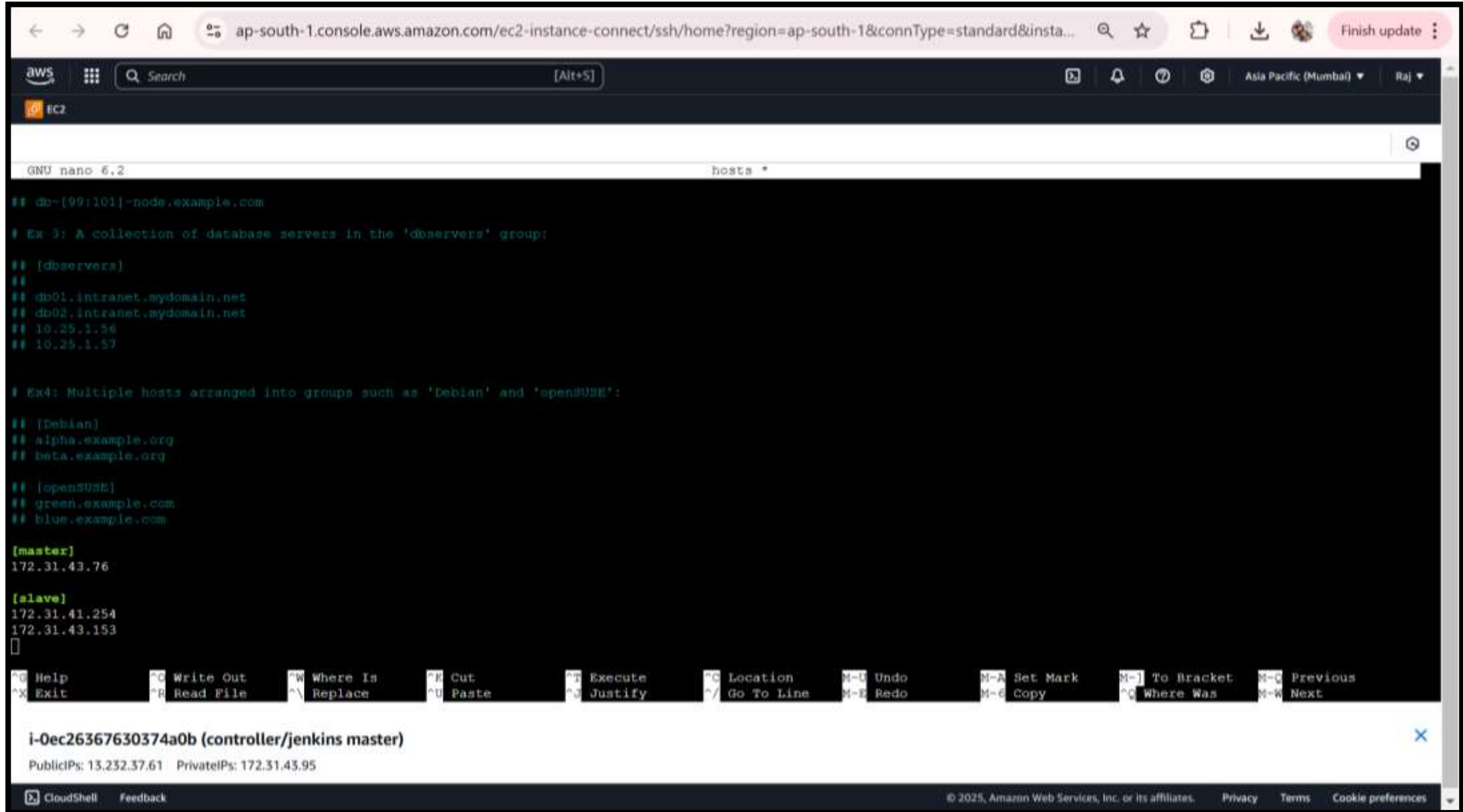
Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa): /home/ubuntu/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_rsa
Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:r78GxxS3zqWw0ukFlkqcb0hcFbxAOQd3x8o+1HH/Dk ubuntu@ip-172-31-43-95
The key's randomart image is:
+---[RSA 3072]-----+
|  o.=+++o  |
|  * . =. .+ |
|  o o . *.o.= |
|  . O = = o ..+ |
|  = -S= o  Eo |
|  + o.* .  . |
|  . . =..  |
|  . ...  |
|  .oo.  |
+---[SHA256]-----+
ubuntu@ip-172-31-43-95:~$ cat /home/ubuntu/.ssh/id_rsa.pub
cat: /home/ubuntu/.ssh/id_rsa.pub: No such file or directory
ubuntu@ip-172-31-43-95:~$ cd .ssh
ubuntu@ip-172-31-43-95:~/ssh$ ls
authorized_keys  id_rsa  id_rsa.pub  known_hosts  known_hosts.old
ubuntu@ip-172-31-43-95:~/ssh$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGC4+0eBjHVO5f3S+HJhQW8FBNROOV3+8InNGaUC8RpN0K14wffnMY+ImYVsFENNqe95IOB14NdcS8yWkFBPJr1rQKc5QpNhZEL7m1PC1/aJCFmLkHAq2FGqE926XSEKcWdgtE5C
5LkyJQ4r2o1jOHL5VCXJiWzuigmv4qcsteYNkiWwNuJVuU17u216zPB7j/kk3GbaHMcIv9MKQFA+Thm8vSh/t9rdzYeD62t1jhg8OL0ZOYXMe0TKJUQoiLe8Vfc2takW8GuJXOD8ip9i24VTbZY08vrAOz0MUNy8qjWAFRy2SsIPW7
PHOuBeo3w1i2d7AcJxqZg02y7hg+uclJP8r8mV0jj3x5VUunD6jWwLYjnxo7mKwqPB1/7YZOY843VT7X2c3JWUHALz7cNnkCePlpYIAMC0Ry1F2DdI05XfbX/mPasp8SDI5rU+ankaZvuDafL/tDA1EGJ39bWKLUVWSp5H0bQwH+I
9Cmiw17HAcSbS/WWi8IAu2stxWS5Xk8= ubuntu@ip-172-31-43-95
ubuntu@ip-172-31-43-95:~/ssh$
```

Below the terminal window, a notification bar shows the instance ID **i-0ec26367630374a0b (controller/jenkins master)** and IP addresses: **Public IPs: 13.233.107.245** and **Private IPs: 172.31.43.95**.

Ssh connection master with slaves machine



Adding private Ips of masters and slaves machine in hosts



```
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&insta...
aws Search [Alt+S] Asia Pacific (Mumbai) Raj
EC2
GNU nano 6.2 hosts *
## db-[99:101]-node.example.com
# Ex 3: A collection of database servers in the 'dbservers' group:
## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Ex4: Multiple hosts arranged into groups such as 'Debian' and 'openSUSE':
## [Debian]
## alpha.example.org
## beta.example.org

## [openSUSE]
## green.example.com
## blue.example.com

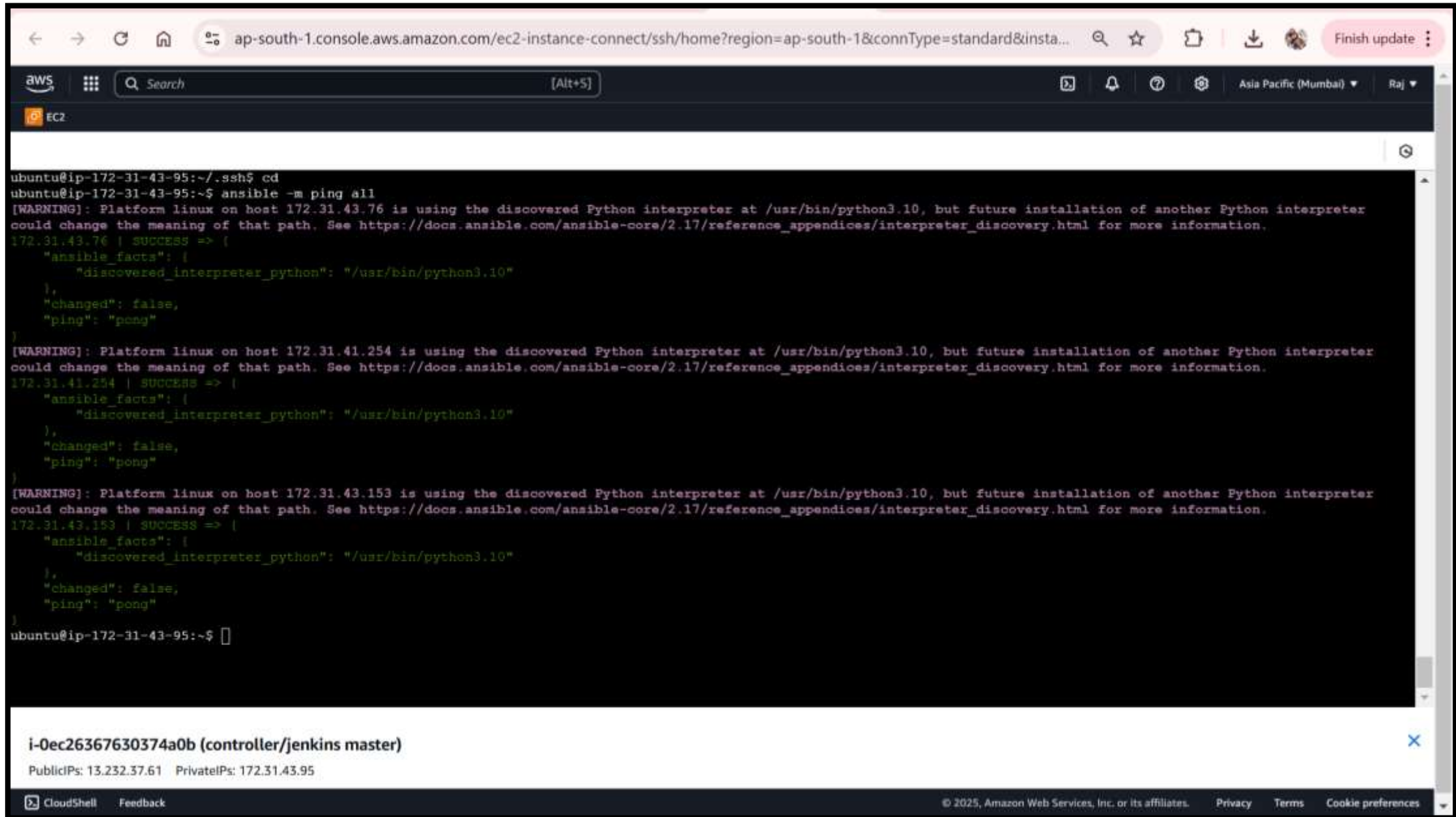
[master]
172.31.43.76

[slave]
172.31.41.254
172.31.43.153
[]

^C Help      ^C Write Out  ^W Where Is  ^K Cut       ^T Execute   ^C Location  ^U Undo      ^M-A Set Mark  ^M-] To Bracket ^M-C Previous
^X Exit      ^R Read File  ^\ Replace   ^U Paste     ^J Justify   ^/_ Go To Line ^M-E Redo    ^M-6 Copy     ^M-^ Where Was  ^M-W Next

i-0ec26367630374a0b (controller/jenkins master)
PublicIPs: 13.232.37.61 PrivateIPs: 172.31.43.95
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

All machine are up and running



The screenshot shows the AWS CloudShell interface. The terminal window displays the output of an Ansible command to ping three EC2 instances. The output shows that all three instances are successfully pinged, with the discovered Python interpreter path being `/usr/bin/python3.10` for each. The instances are identified by their private IP addresses: 172.31.43.76, 172.31.41.254, and 172.31.43.153. The terminal also shows a warning message about the Python interpreter path.

```
ubuntu@ip-172-31-43-95:~/.ssh$ cd
ubuntu@ip-172-31-43-95:~$ ansible -m ping all
[WARNING]: Platform linux on host 172.31.43.76 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter
could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
172.31.43.76 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Platform linux on host 172.31.41.254 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter
could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
172.31.41.254 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Platform linux on host 172.31.43.153 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter
could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
172.31.43.153 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "changed": false,
  "ping": "pong"
}
ubuntu@ip-172-31-43-95:~$
```

i-0ec26367630374a0b (controller/jenkins master)

PublicIPs: 13.232.37.61 PrivateIPs: 172.31.43.95

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Ansible playbook play.yaml

← → ↻ 🏠 🔍 ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=... 🔍 ☆ 📄 👤 Finish update ⋮

aws 🔍 Search [Alt+S] 📄 🔔 ? ⚙️ Asia Pacific (Mumbai) ▾ Raj ▾

EC2

GNU nano 6.2 play.yaml *

```
--
- name: install Jenkins and Java on host
  become: true
  hosts: localhost
  tasks:
    - name: running script to install tools on host
      script: script1.sh
- name: install docker, K8s and Java on main
  become: true
  hosts: master
  tasks:
    - name: running script to install tools on main
      script: script2.sh
- name: install docker, K8s on nodes
  become: true
  hosts: slaves
  tasks:
    - name: running script to install tools on node
      script: script3.sh
```

⌘ Help ⌘C Write Out ⌘W Where Is ⌘R Cut ⌘I Execute ⌘O Location ⌘U Undo ⌘A Set Mark ⌘J To Bracket ⌘Q Previous
⌘X Exit ⌘F Read File ⌘\ Replace ⌘P Paste ⌘J Justify ⌘/_ Go To Line ⌘E Redo ⌘G Copy ⌘Q Where Was ⌘N Next

i-0ec26367630374a0b (controller/jenkins master) ✕

PublicIPs: 13.232.37.61 PrivateIPs: 172.31.43.95

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

← → ↻ 🏠 🔍 ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=... 🔍 ☆ 📁 🧑 Finish update ⋮

aws 🔍 Search [Alt+S]

📌 EC2

TASK [running script to install tools on host] *****
skipping: [localhost]

PLAY [install docker, K8s and Java on main] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host 172.31.43.76 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [172.31.43.76]

TASK [running script to install tools on main] *****
skipping: [172.31.43.76]

PLAY [install docker, K8s on nodes] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host 172.31.41.254 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [172.31.41.254]
[WARNING]: Platform linux on host 172.31.43.153 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [172.31.43.153]

TASK [running script to install tools on node] *****
skipping: [172.31.41.254]
skipping: [172.31.43.153]

PLAY RECAP *****
172.31.41.254 : ok=1 changed=0 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0
172.31.43.153 : ok=1 changed=0 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0
172.31.43.76 : ok=1 changed=0 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0
localhost : ok=1 changed=0 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0

ubuntu@ip-172-31-43-95:/etc/ansible\$ ansible-playbook play.yaml

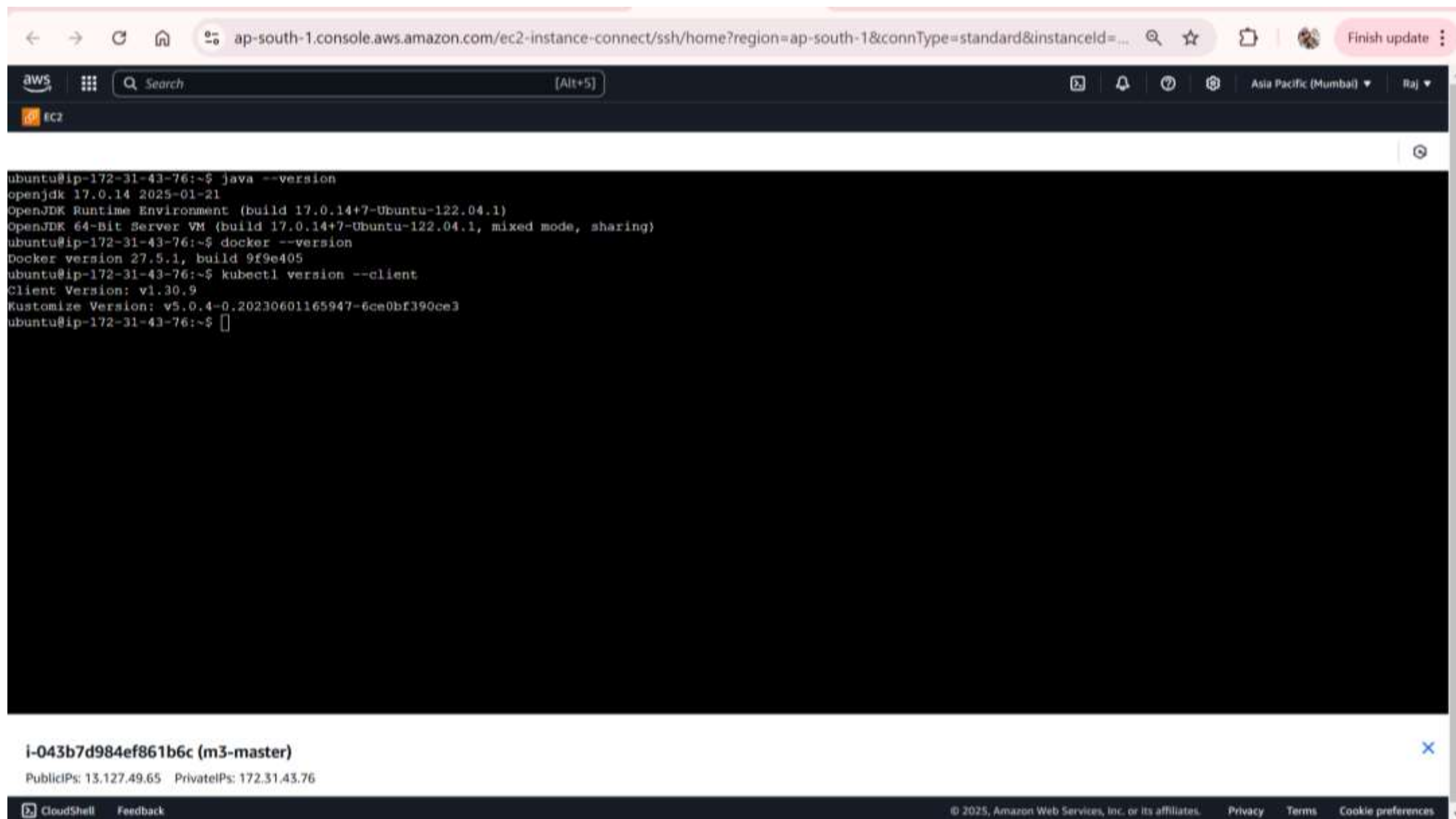
i-0ec26367630374a0b (controller/jenkins master)

PublicIPs: 13.232.37.61 PrivateIPs: 172.31.43.95

📄 CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

m3-k8s-master Kubernetes , docker & Java Installed



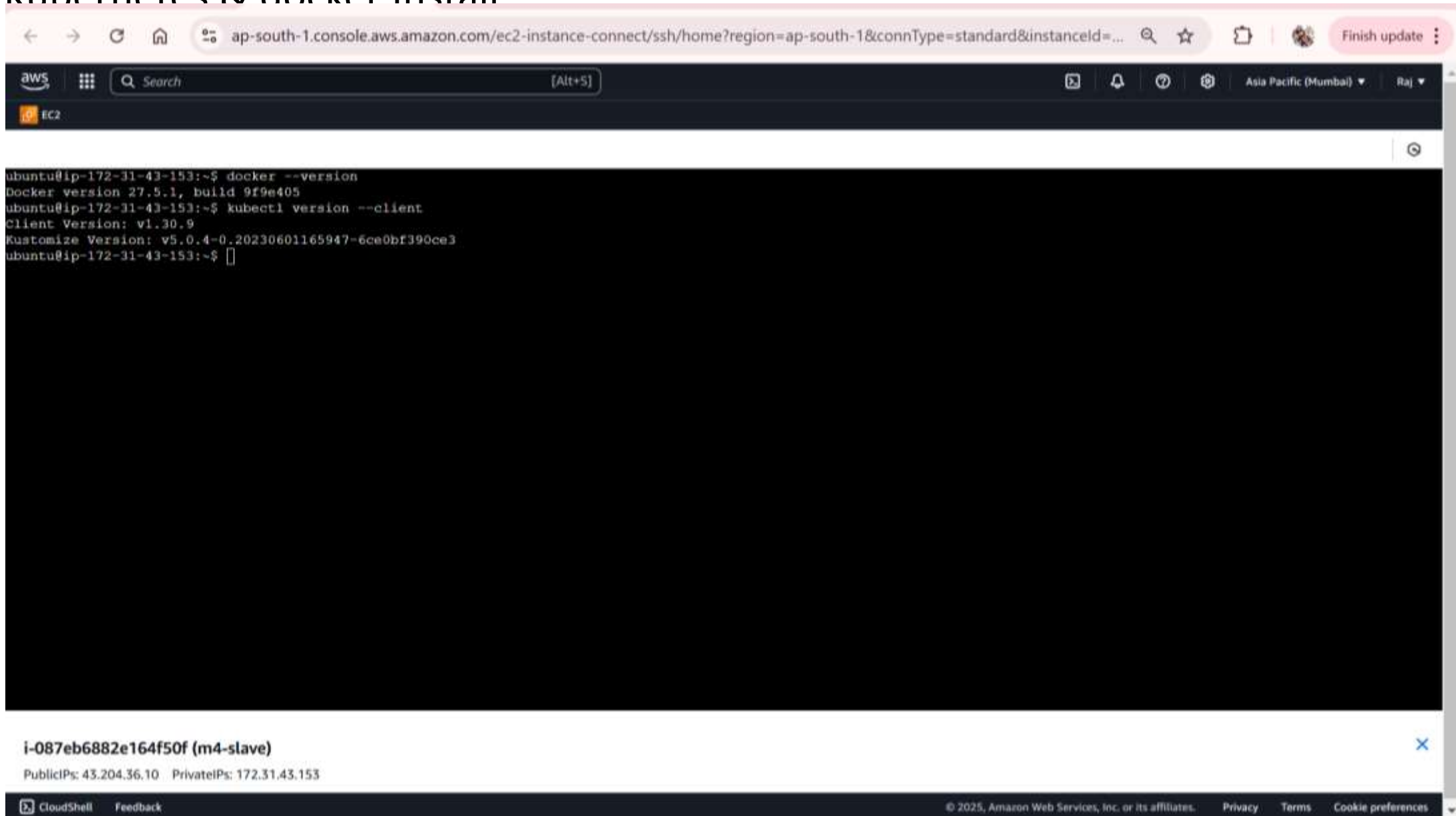
The screenshot displays the AWS Management Console interface. At the top, the browser address bar shows the URL: `ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=...`. The console header includes the AWS logo, a search bar, and navigation icons. The main content area features a terminal window with the following output:

```
ubuntu@ip-172-31-43-76:~$ java --version
openjdk 17.0.14 2025-01-21
OpenJDK Runtime Environment (build 17.0.14+7-Ubuntu-122.04.1)
OpenJDK 64-Bit Server VM (build 17.0.14+7-Ubuntu-122.04.1, mixed mode, sharing)
ubuntu@ip-172-31-43-76:~$ docker --version
Docker version 27.5.1, build 9f9e405
ubuntu@ip-172-31-43-76:~$ kubectl version --client
Client Version: v1.30.9
Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3
ubuntu@ip-172-31-43-76:~$
```

Below the terminal window, the instance details for `i-043b7d984ef861b6c (m3-master)` are shown, including Public IPs: `13.127.49.65` and Private IPs: `172.31.43.76`. The footer of the console contains the CloudShell logo, a feedback link, and copyright information for Amazon Web Services, Inc. or its affiliates, along with links for Privacy, Terms, and Cookie preferences.

m4-k8s-slave

Kubernetes & docker Install



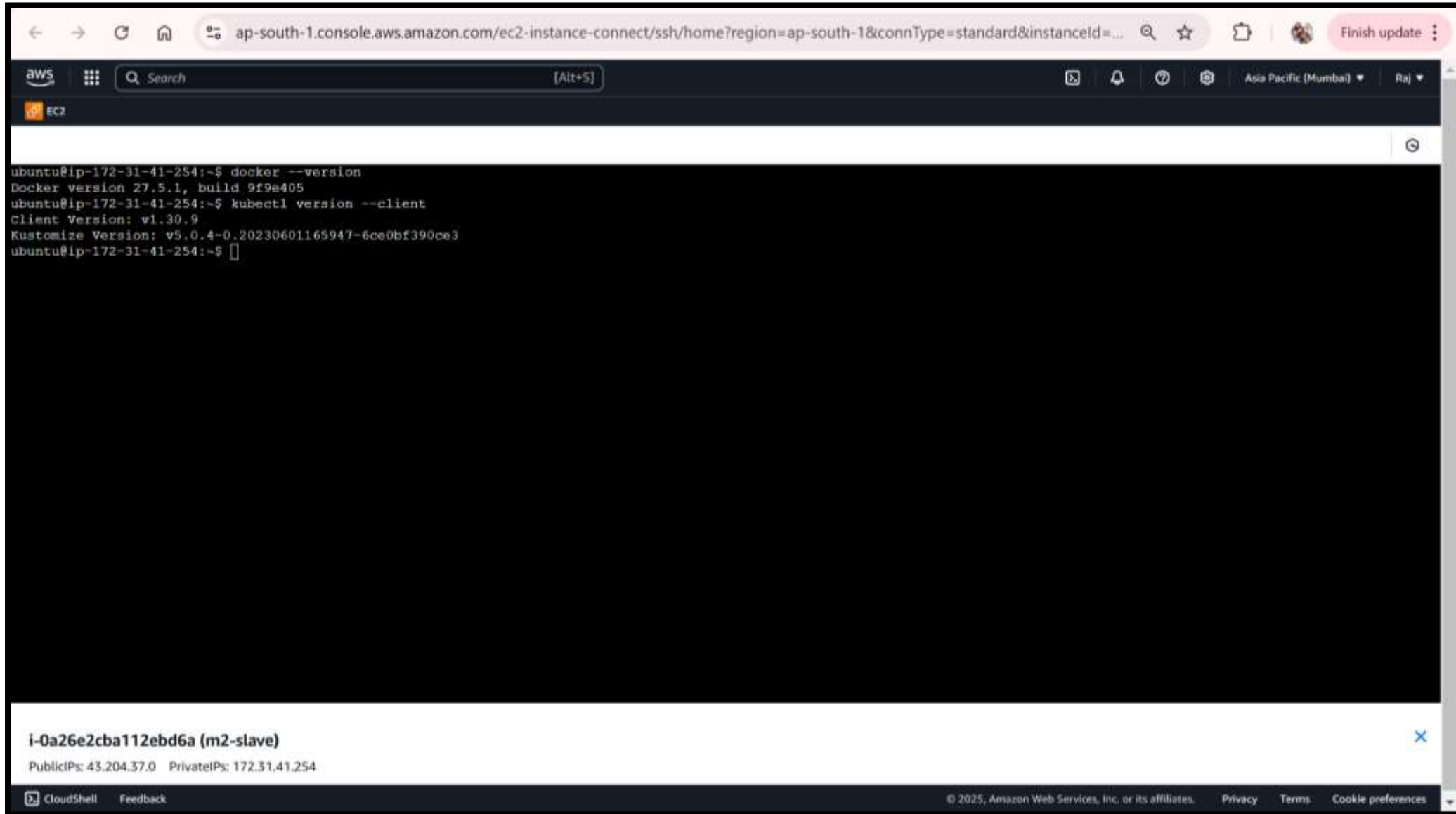
The screenshot shows the AWS CloudShell interface. The browser address bar displays the URL: `ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=...`. The AWS console header includes the AWS logo, a search bar, and navigation icons. The main terminal area shows the following commands and output:

```
ubuntu@ip-172-31-43-153:~$ docker --version
Docker version 27.5.1, build 9f9e405
ubuntu@ip-172-31-43-153:~$ kubectl version --client
Client Version: v1.30.9
Kustomize Version: v5.0.4-0.20230601165947-6cae0bf390ce3
ubuntu@ip-172-31-43-153:~$
```

At the bottom of the terminal window, the instance details are shown: `i-087eb6882e164f50f (m4-slave)` with PublicIPs: 43.204.36.10 and PrivateIPs: 172.31.43.153. The footer of the CloudShell interface includes the CloudShell logo, a feedback link, and copyright information: © 2025, Amazon Web Services, Inc. or its affiliates. It also includes links for Privacy, Terms, and Cookie preferences.

m2-k8s-slave

Kubernetes & docker Install



The screenshot shows the AWS CloudShell interface. The terminal window displays the following commands and output:

```
ubuntu@ip-172-31-41-254:~$ docker --version
Docker version 27.5.1, build 9f9e405
ubuntu@ip-172-31-41-254:~$ kubectl version --client
Client Version: v1.30.9
Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3
ubuntu@ip-172-31-41-254:~$
```

Below the terminal window, the instance details are shown:

i-0a26e2cba112ebd6a (m2-slave)
PublicIPs: 43.204.37.0 PrivateIPs: 172.31.41.254

The footer of the CloudShell interface includes the text "CloudShell Feedback" and "© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences".

Kubernetes master nodes

ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=i...

aws [Alt+S] Search

EC2

```
ubuntu@ip-172-31-46-65:~$ kubectl get nodes
NAME                STATUS    ROLES    AGE   VERSION
ip-172-31-42-41     Ready    <none>   5m24s v1.30.9
ip-172-31-46-65     Ready    control-plane  8m1s  v1.30.9
ip-172-31-47-112    Ready    <none>   5m35s v1.30.9
ubuntu@ip-172-31-46-65:~$
```

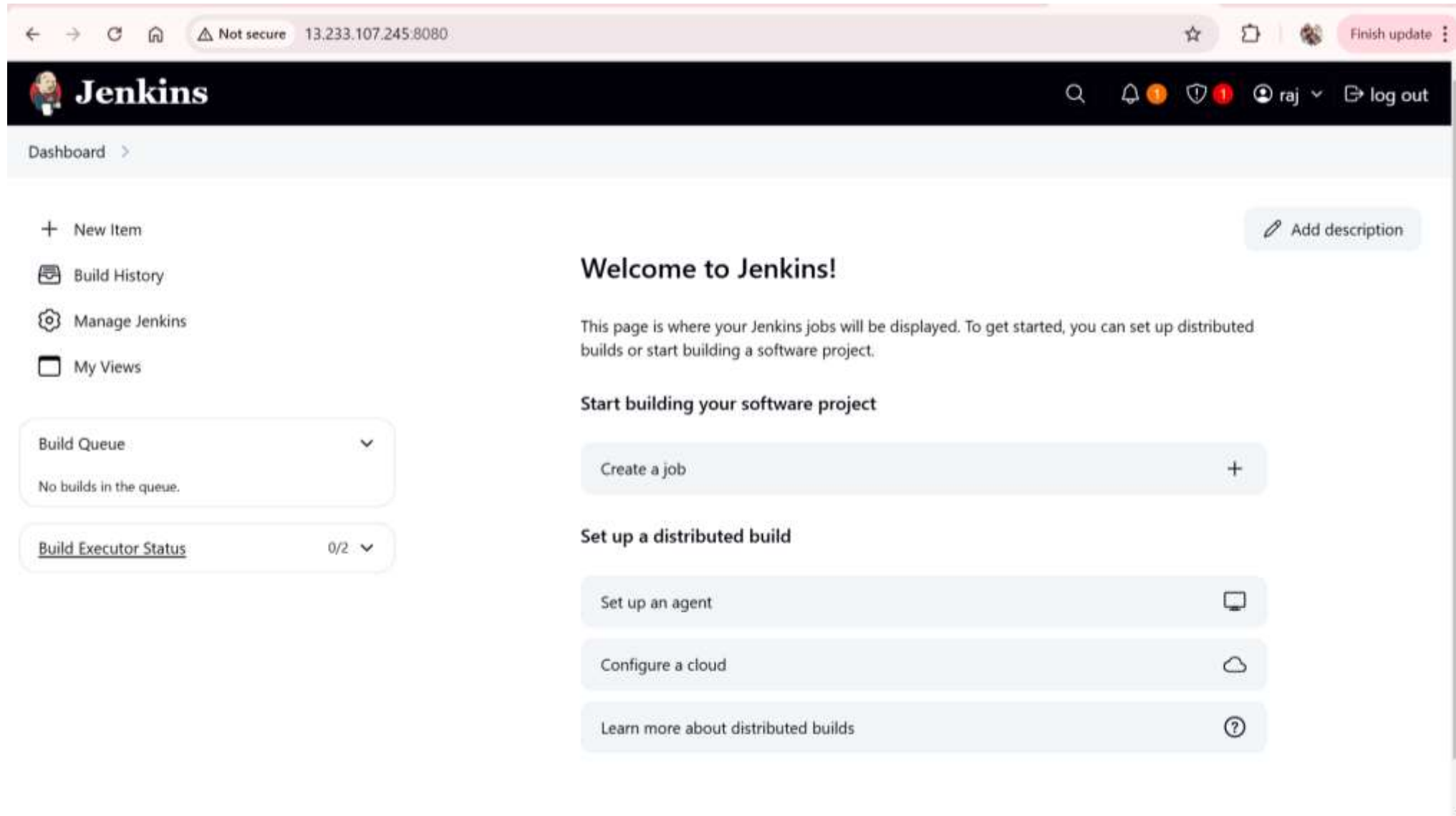
i-06b12002f11ac4ff9 (m3-master)

PublicIPs: 3.109.214.54 PrivateIPs: 172.31.46.65

CloudShell Feedback


© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Connection to Jenkins port using public Ips of master-Jenkins machine



The screenshot shows the Jenkins web interface in a browser. The address bar indicates a 'Not secure' connection to the IP address 13.233.107.245:8080. The Jenkins logo and name are in the top left, and a user profile 'raj' with a 'log out' button is in the top right. The main content area features a 'Welcome to Jenkins!' message, explaining that this is where jobs are displayed and can be set up. Below this, there are two main sections: 'Start building your software project' with a 'Create a job' button, and 'Set up a distributed build' with options to 'Set up an agent', 'Configure a cloud', and 'Learn more about distributed builds'. On the left sidebar, there are links for 'New Item', 'Build History', 'Manage Jenkins', and 'My Views', along with status boxes for 'Build Queue' (empty) and 'Build Executor Status' (0/2).

← → ↻ 🏠 ⚠ Not secure 13.233.107.245:8080 ☆ 📁 👤 Finish update ⋮

 **Jenkins** 🔍 🔔 🛡️ 🚫 👤 raj ▾ 🚪 log out

Dashboard >

+ New Item

📁 Build History

⚙️ Manage Jenkins

📋 My Views

Build Queue ▾
No builds in the queue.

Build Executor Status 0/2 ▾

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

[Add description](#)

Start building your software project

Create a job +

Set up a distributed build


Set up an agent 🖥️






Configure a cloud ☁️

Learn more about distributed builds ?


Created a node for **m3-k8s-master**
named as “k8node”


← ↻ ⚠ Not secure | 13.201.79.254:8080/manage/computer/

 **Jenkins**

 ras  log out

Dashboard > Manage Jenkins > Nodes >


 Nodes

 Clouds


Build Queue


No builds in the queue.

Build Executor Status

 Built-In Node

1 Idle

 2 Idle


 k8node







1 Idle

Nodes

+ New Node

Configure Monitors



S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-In Node	Linux (amd64)	In sync	4.06 GiB	 0 B	4.06 GiB	0ms 
	k8node	Linux (amd64)	In sync	3.03 GiB	 0 B	3.03 GiB	26ms 
Data obtained		1 min 48 sec	1 min 48 sec	1 min 48 sec	1 min 48 sec	1 min 48 sec	1 min 48 sec

Icon:

S

M

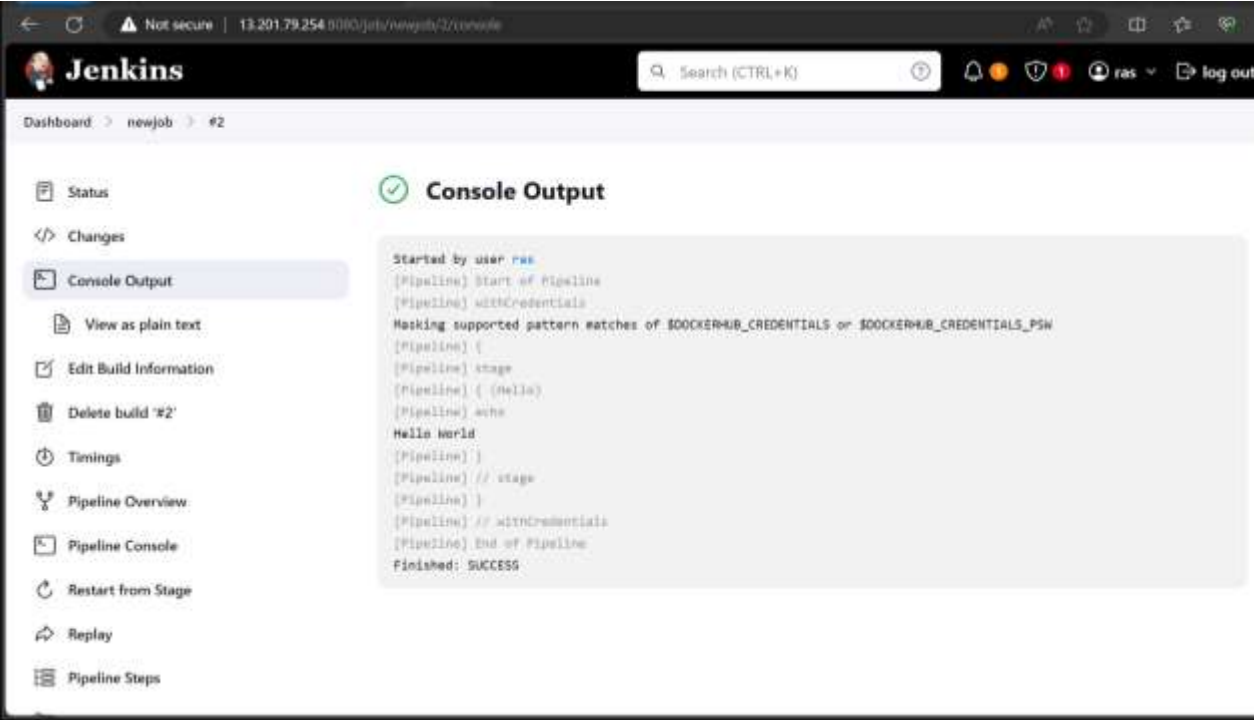
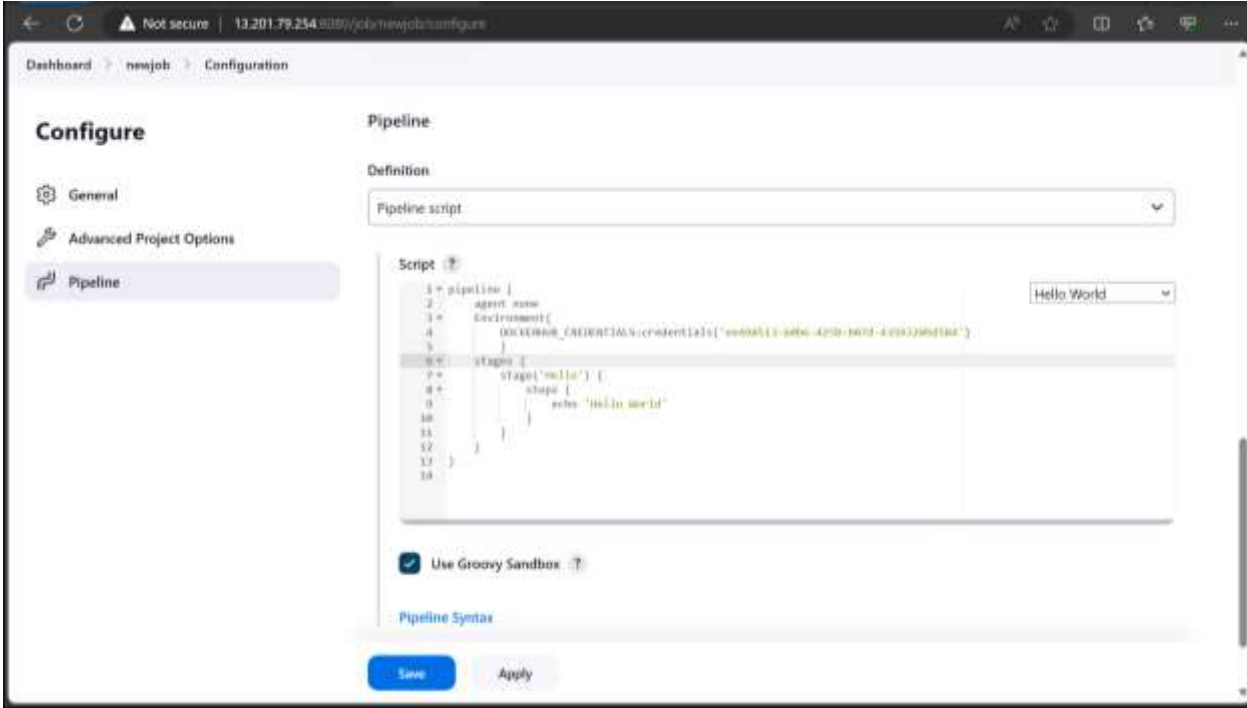
L

Legend

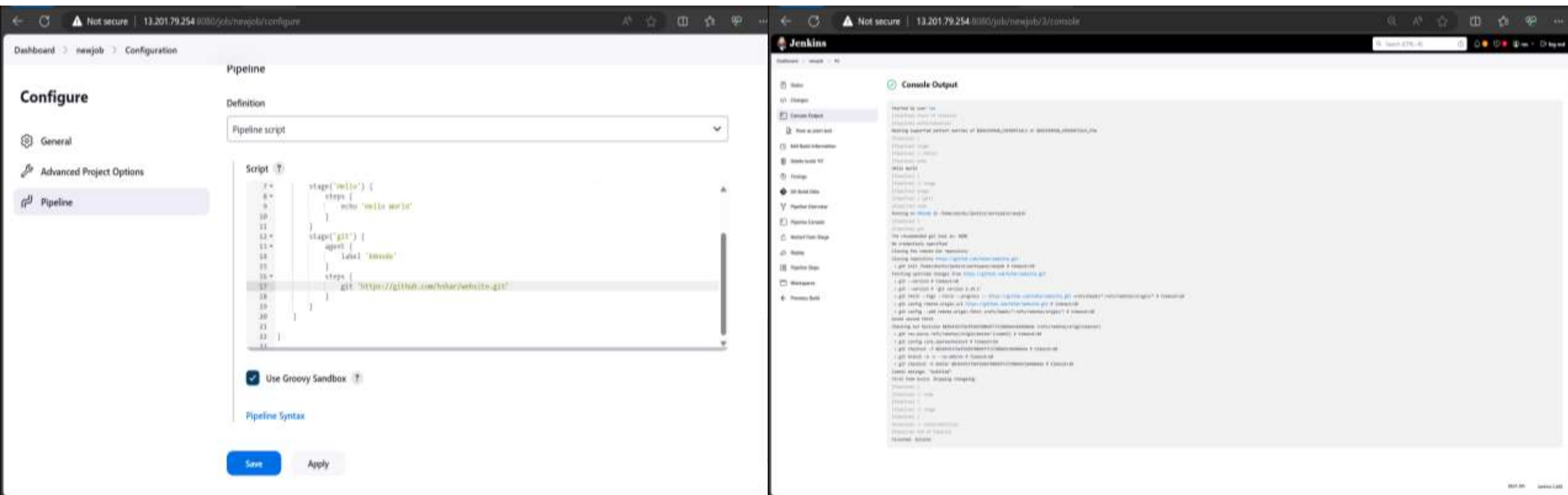
REST API

Jenkins 2.460

Created a newjob



Adding git stage



Reflecting all the file from Jenkins and git in m3-k8s-master machine

```
ubuntu@ip-172-31-44-22:~$ ls
jenkins
ubuntu@ip-172-31-44-22:~$ cd jenkins
ubuntu@ip-172-31-44-22:~/jenkins$ ls
remoting  remoting.jar  workspace
ubuntu@ip-172-31-44-22:~/jenkins$ cd workspace
ubuntu@ip-172-31-44-22:~/jenkins/workspace$ ls
newjob
ubuntu@ip-172-31-44-22:~/jenkins/workspace$ cd newjob
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
images  index.html
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$
```

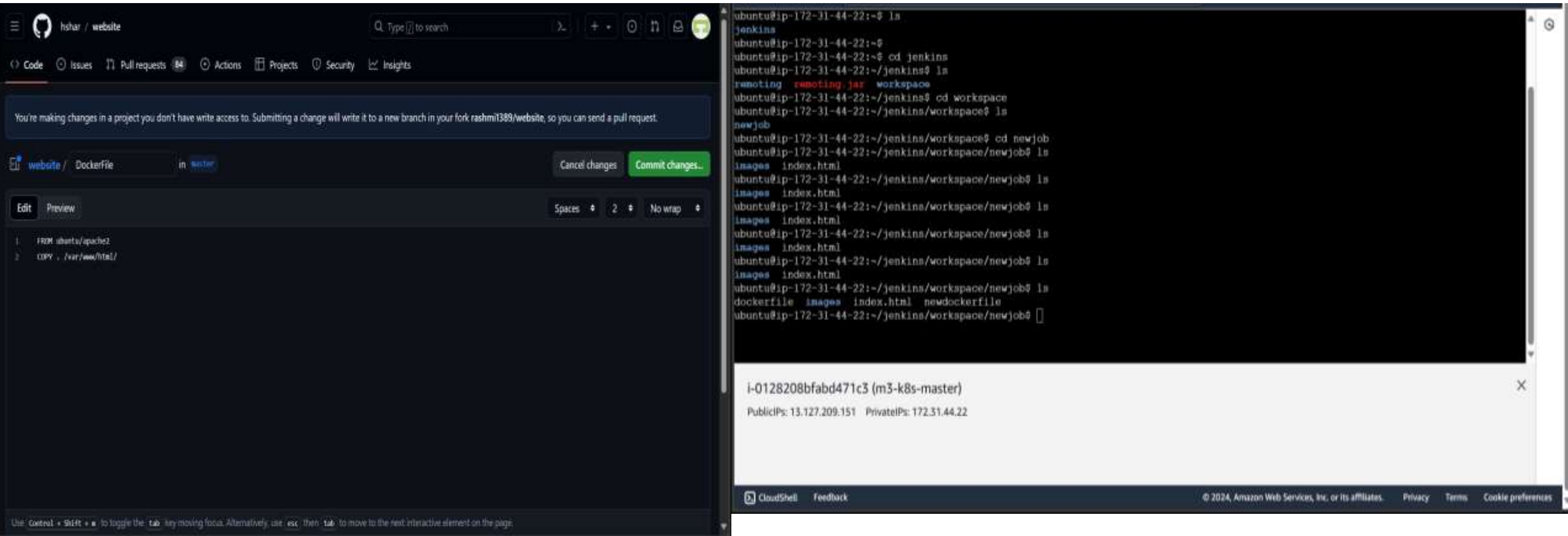
i-0128208bfabd471c3 (m3-k8s-master)

PublicIPs: 13.127.209.151 PrivateIPs: 172.31.44.22

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Create a Dockerfile in git and commit changes



The image shows a GitHub repository editor interface on the left and a CloudShell terminal window on the right.

GitHub Repository Editor (Left):

- Repository: hshar / website
- File: website / DockerFile
- Buttons: Edit, Preview, Spaces: 2, No wrap
- Code content:

```
1 FROM ubuntu:quay2
2 COPY . /var/www/html/
```
- Buttons: Cancel changes, Commit changes..

CloudShell Terminal (Right):

```
ubuntu@ip-172-31-44-22:~$ ls
jenkins
ubuntu@ip-172-31-44-22:~$ cd jenkins
ubuntu@ip-172-31-44-22:~/jenkins$ ls
remoting  remoting.jar  workspace
ubuntu@ip-172-31-44-22:~/jenkins$ cd workspace
ubuntu@ip-172-31-44-22:~/jenkins/workspace$ ls
newjob
ubuntu@ip-172-31-44-22:~/jenkins/workspace$ cd newjob
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
images  index.html
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
images  index.html
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
images  index.html
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
images  index.html
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
dockerfile  images  index.html  newdockerfile
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$
```

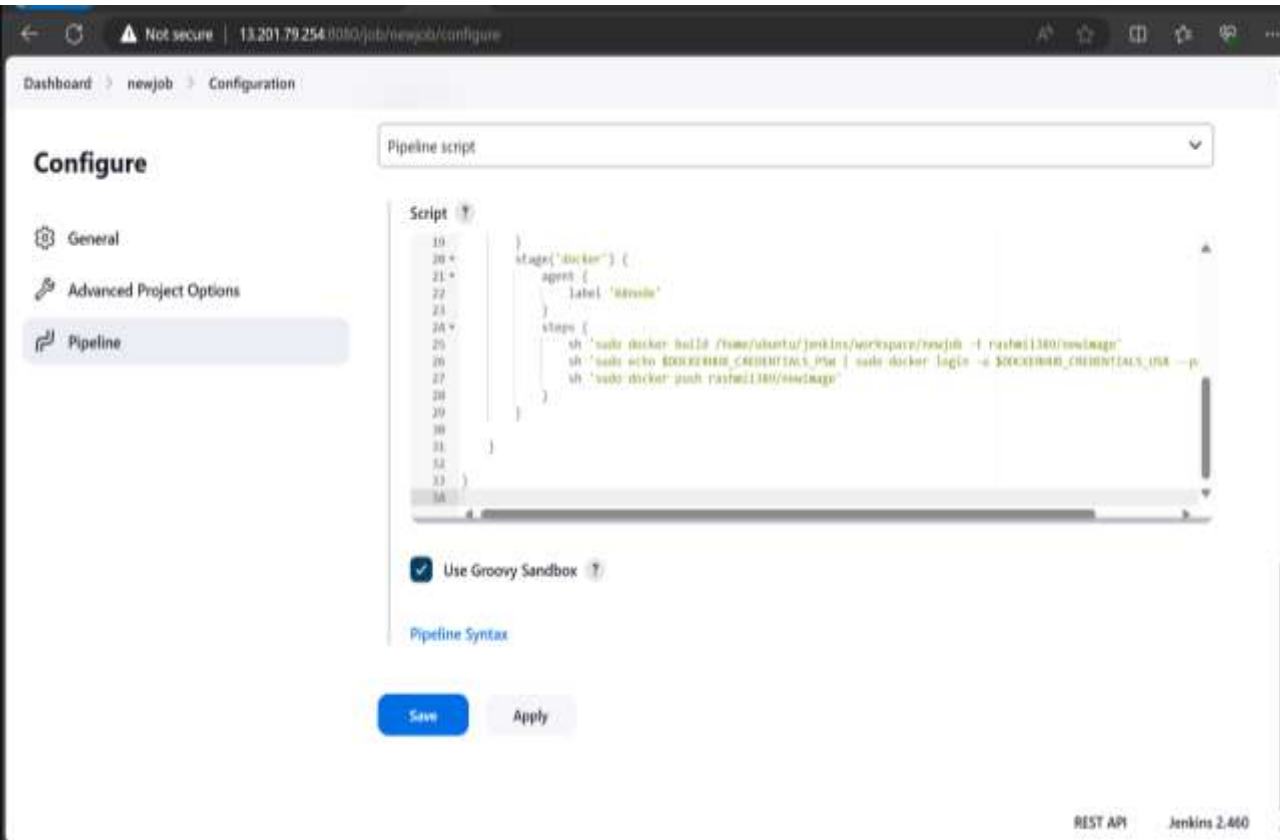
CloudShell Metadata:

- Instance: i-0128208bfabd471c3 (m3-k8s-master)
- Public IPs: 13.127.209.151 Private IPs: 172.31.44.22

Footer:

- CloudShell Feedback
- © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Configuration for building a docker image and pushing to docker hub



The screenshot shows the Jenkins 'Configure' page for a new job. The 'Pipeline script' tab is selected, displaying a Groovy script for building and pushing a Docker image. The script defines a stage named 'docker' with a parallel block containing two steps: building the image and pushing it to Docker Hub. The 'Use Groovy Sandbox' checkbox is checked. The 'Save' button is highlighted.

Dashboard > newjob > Configuration

Configure

Pipeline script

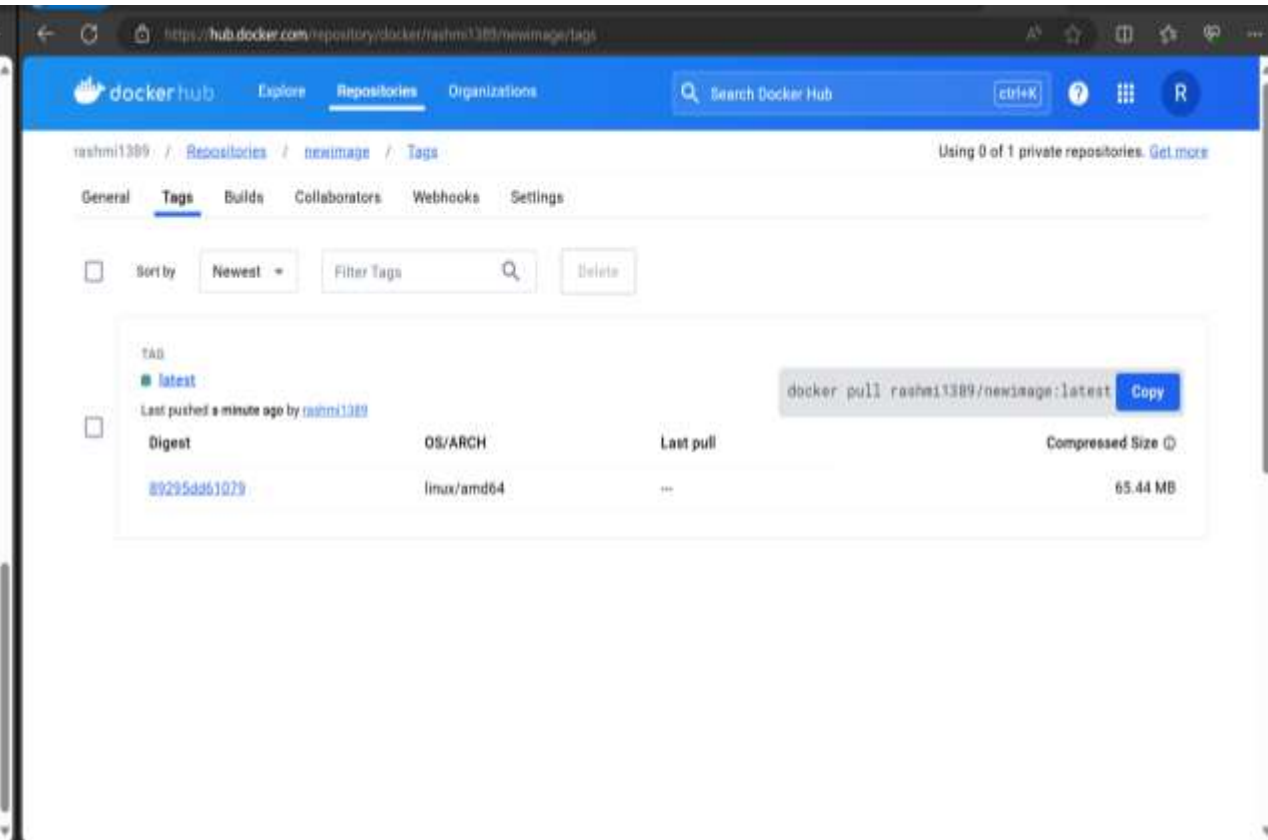
```
19
20 *
21 *
22 *
23 *
24 *
25 sh "sudo docker build -f /home/ubuntu/jenkins/workspace/newjob -t rashmi1389/newimage"
26 sh "sudo echo $DOCKERHUB_CREDENTIALS_PSW | sudo docker login -u $DOCKERHUB_ORIENTALS_USER -p"
27 sh "sudo docker push rashmi1389/newimage"
28
29
30
31
32
33
34
```

☒ Use Groovy Sandbox

[Pipeline Syntax](#)

Save Apply

REST API Jenkins 2.460



The screenshot shows the Docker Hub repository page for 'rashmi1389/newimage'. The 'Tags' tab is selected, showing a table of tags. The 'latest' tag is highlighted, with a 'Copy' button next to the pull command 'docker pull rashmi1389/newimage:latest'. The table lists the digest, OS/ARCH, last pull, and compressed size for the 'latest' tag.

docker hub Explore Repositories Organizations

Search Docker Hub

rashmi1389 / Repositories / newimage / Tags

Using 0 of 1 private repositories. [Get more](#)

General Tags Builds Collaborators Webhooks Settings

Sort by Newest Filter Tags Delete

Tag

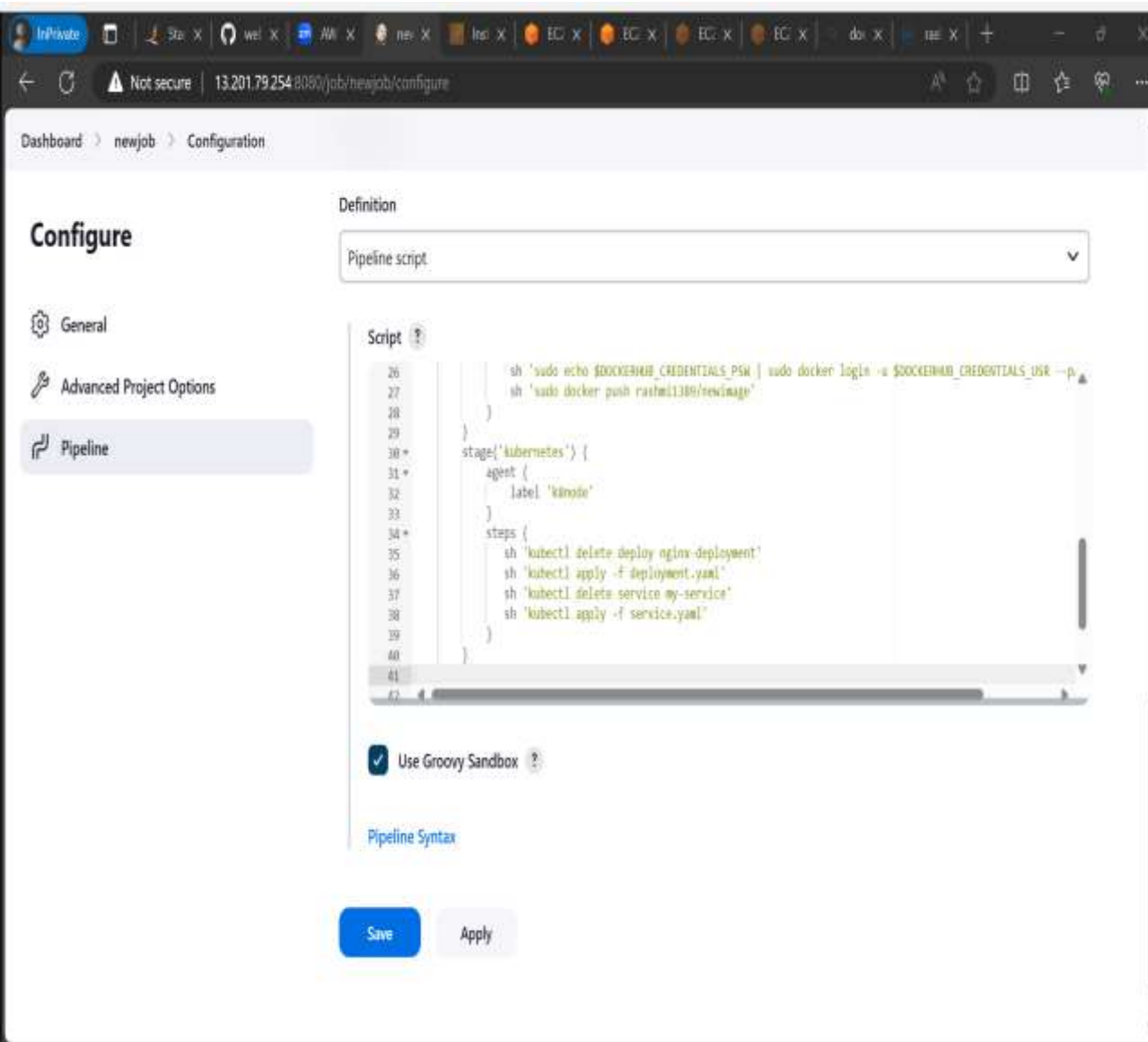
latest

Last pushed a minute ago by rashmi1389

docker pull rashmi1389/newimage:latest **Copy**

Digest	OS/ARCH	Last pull	Compressed Size
89295d861079	linux/amd64	...	65.44 MB

Configuring Kubernetes in jenkins



The screenshot shows the Jenkins 'Configure' page for a new job named 'newjob'. The 'Definition' dropdown is set to 'Pipeline script'. The 'Script' section contains a pipeline script that defines a stage named 'kubernetes' with a 'kubernetes' agent and a 'steps' block containing four shell commands: deleting the 'nginx-deployment', applying 'deployment.yaml', deleting the 'my-service', and applying 'service.yaml'. The 'Use Groovy Sandbox' checkbox is checked. The 'Pipeline Syntax' link is visible at the bottom.

Dashboard > newjob > Configuration

Configure

General
Advanced Project Options
Pipeline

Definition

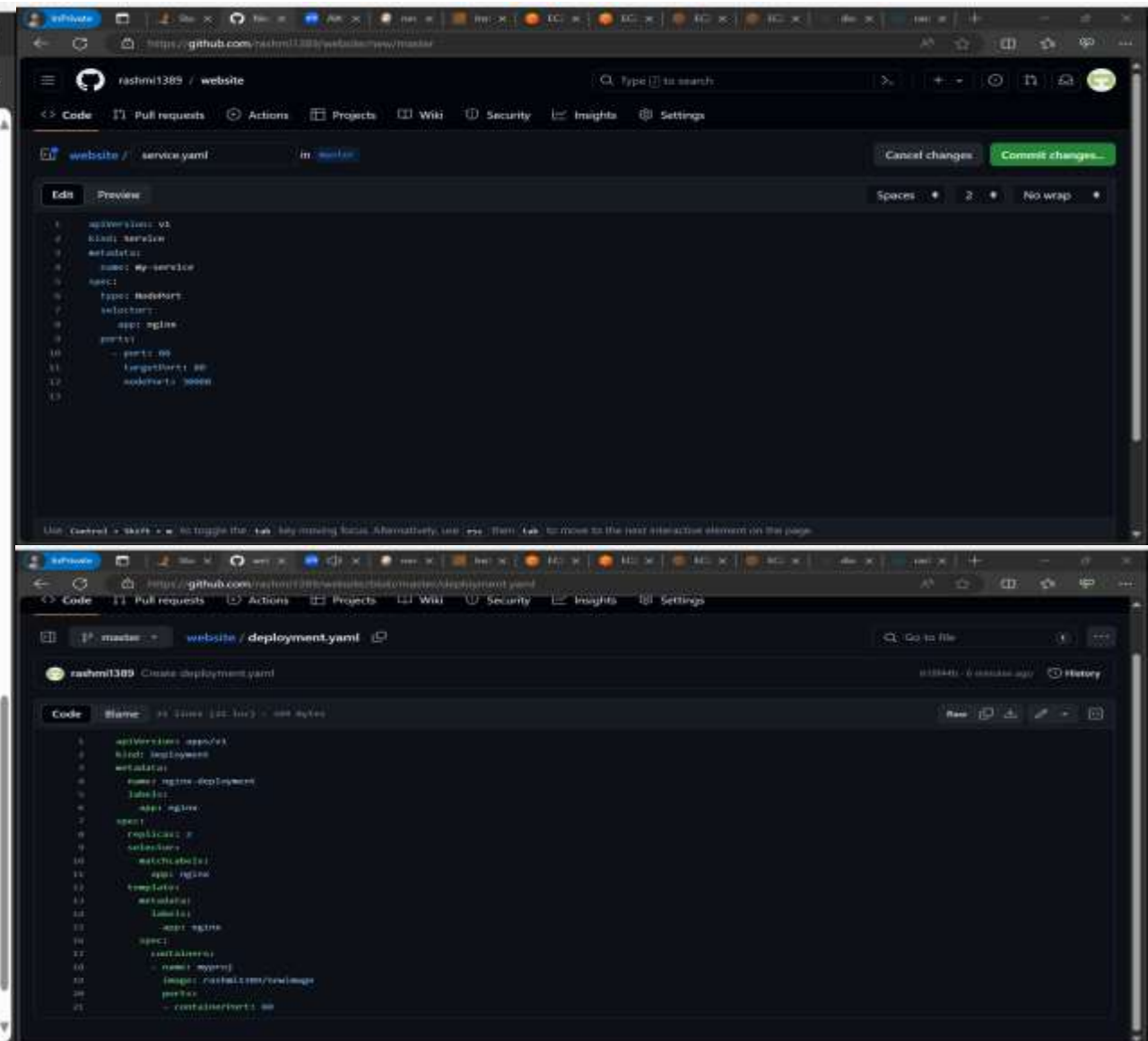
Pipeline script

```
26      sh 'sudo echo $DOCKERHUB_CREDENTIALS_PSW | sudo docker login -u $DOCKERHUB_CREDENTIALS_USR --p
27      sh 'sudo docker push rashmi1389/newimage'
28  }
29  stage('kubernetes') {
30    agent {
31      label 'kubernetes'
32    }
33    steps {
34      sh 'kubectl delete deploy nginx-deployment'
35      sh 'kubectl apply -f deployment.yaml'
36      sh 'kubectl delete service my-service'
37      sh 'kubectl apply -f service.yaml'
38    }
39  }
40
41
42
```

☒ Use Groovy Sandbox

[Pipeline Syntax](#)

Save Apply



The top screenshot shows the 'service.yaml' file in the 'rashmi1389 / website' repository. The file defines a Kubernetes service named 'my-service' of type 'NodePort', exposing port 80 on the node port 30080.

```
1 apiVersion: v1
2 kind: Service
3 metadata:
4   name: my-service
5 spec:
6   type: NodePort
7   selector:
8     app: nginx
9   ports:
10     - port: 80
11       targetPort: 80
12       nodePort: 30080
13
```

The bottom screenshot shows the 'deployment.yaml' file in the same repository. The file defines a Kubernetes deployment named 'nginx-deployment' with 2 replicas, using the 'nginx' image from 'rashmi1389/newimage'.

```
1 apiVersion: apps/v1
2 kind: Deployment
3 metadata:
4   name: nginx-deployment
5 labels:
6   app: nginx
7 spec:
8   replicas: 2
9   selector:
10     matchLabels:
11       app: nginx
12   template:
13     metadata:
14       labels:
15         app: nginx
16     spec:
17       containers:
18         - name: nginx
19           image: rashmi1389/newimage
20           ports:
21             - containerPort: 80

```

Reflecting deployment.yaml and service.yaml file in machine m3-k8s-master

```
aws
Services
Search [Alt+S]
Mumbai ritu toppo

ubuntu@ip-172-31-44-22:~$
ubuntu@ip-172-31-44-22:~$ cd jenkins
ubuntu@ip-172-31-44-22:~/jenkins$ ls
remoting  remoting.jar  workspace
ubuntu@ip-172-31-44-22:~/jenkins$ cd workspace
ubuntu@ip-172-31-44-22:~/jenkins/workspace$ ls
newjob
ubuntu@ip-172-31-44-22:~/jenkins/workspace$ cd newjob
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
images  index.html
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
images  index.html
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
images  index.html
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
images  index.html
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
images  index.html
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
dockerfile  images  index.html  newdockerfile
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
Dockerfile  dockerfile  images  index.html
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$ ls
Dockerfile  deployment.yaml  dockerfile  images  index.html  service.yaml
ubuntu@ip-172-31-44-22:~/jenkins/workspace/newjob$

i-0128208bfabd471c3 (m3-k8s-master)
PublicIPs: 13.127.209.151 PrivateIPs: 172.31.44.22

CloudShell Feedback
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

```
Not secure | 13.201.79.254:8080/job/newjob/1/console
Dashboard newjob #11

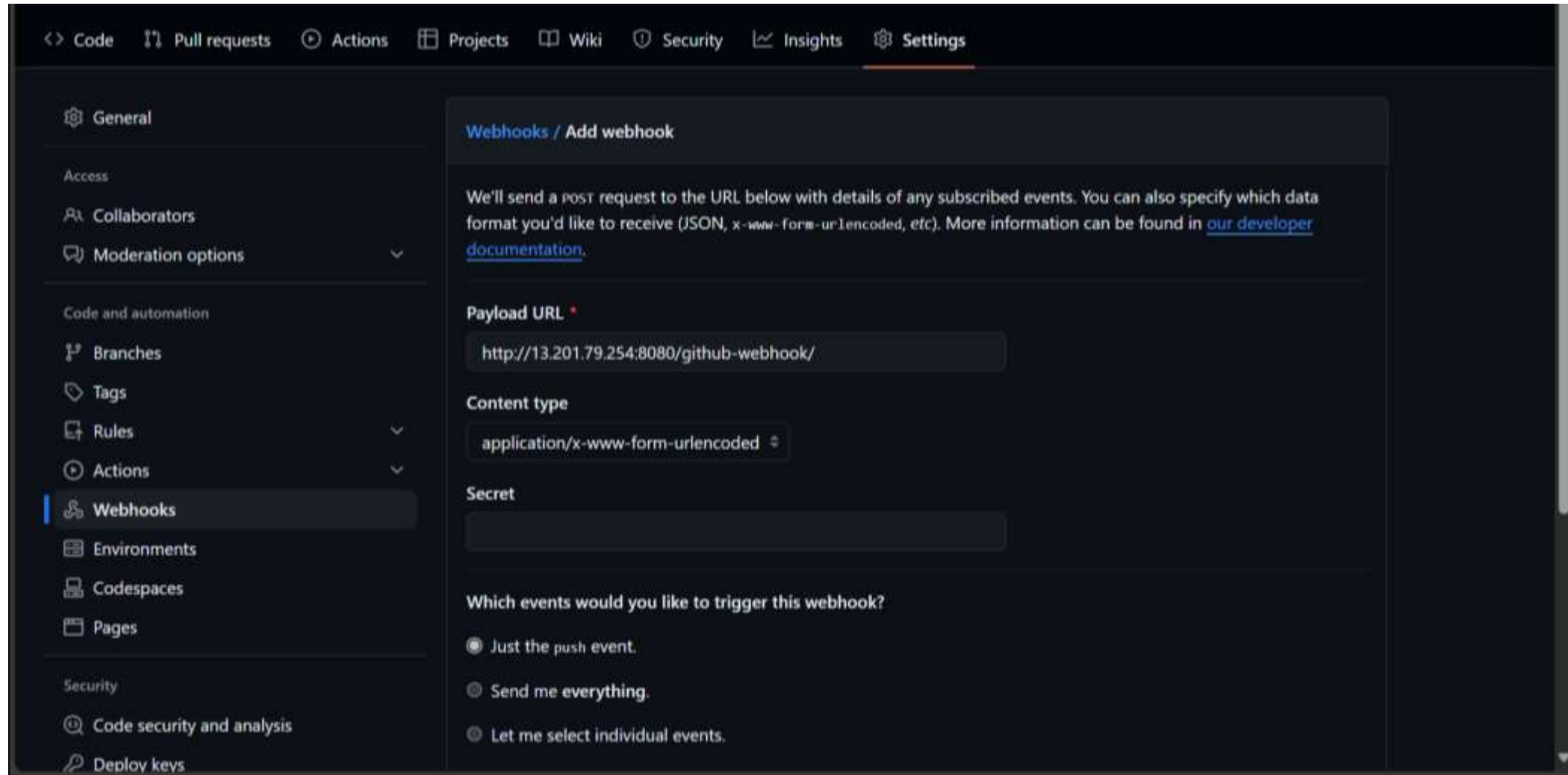
[Pipeline] }
[Pipeline] // node
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (kubernetes)
[Pipeline] node
Running on k8snode in /home/ubuntu/jenkins/workspace/newjob
[Pipeline] {
[Pipeline] sh
+ kubectl apply -f deployment.yaml
deployment.apps/nginx-deployment created
[Pipeline] sh
+ kubectl apply -f service.yaml
service/my-service created
[Pipeline] }
[Pipeline] // node
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] End of Pipeline
Finished: SUCCESS

REST API Jenkins 2.460
```


Able to access the web page using slave machine in port 30008










Connecting Jenkins to github webhook to automate workflow



Build stage



 **Jenkins**




Search (CTRL+K) ?

     ras   log out

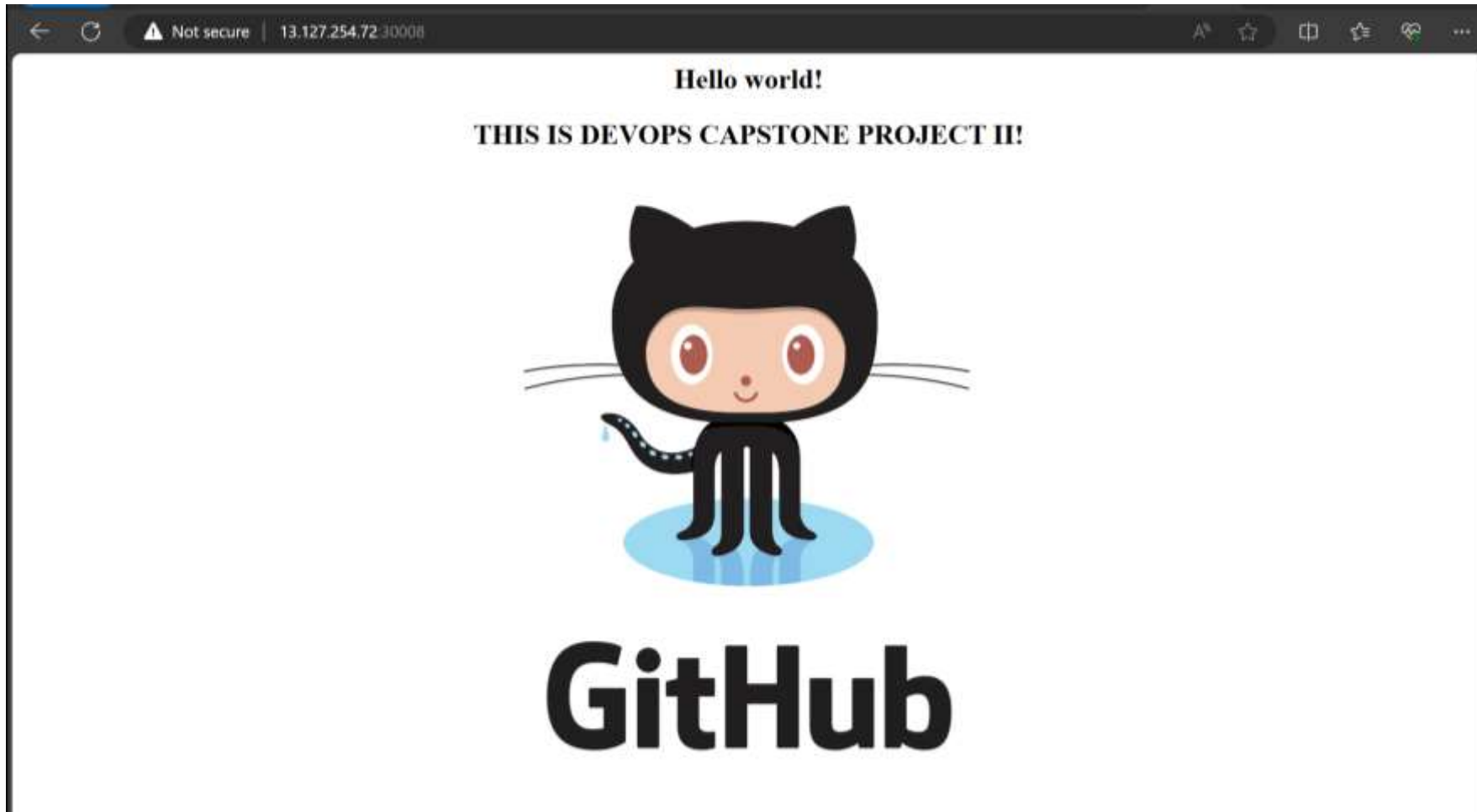
Dashboard > newjob > Stages

Build newjob

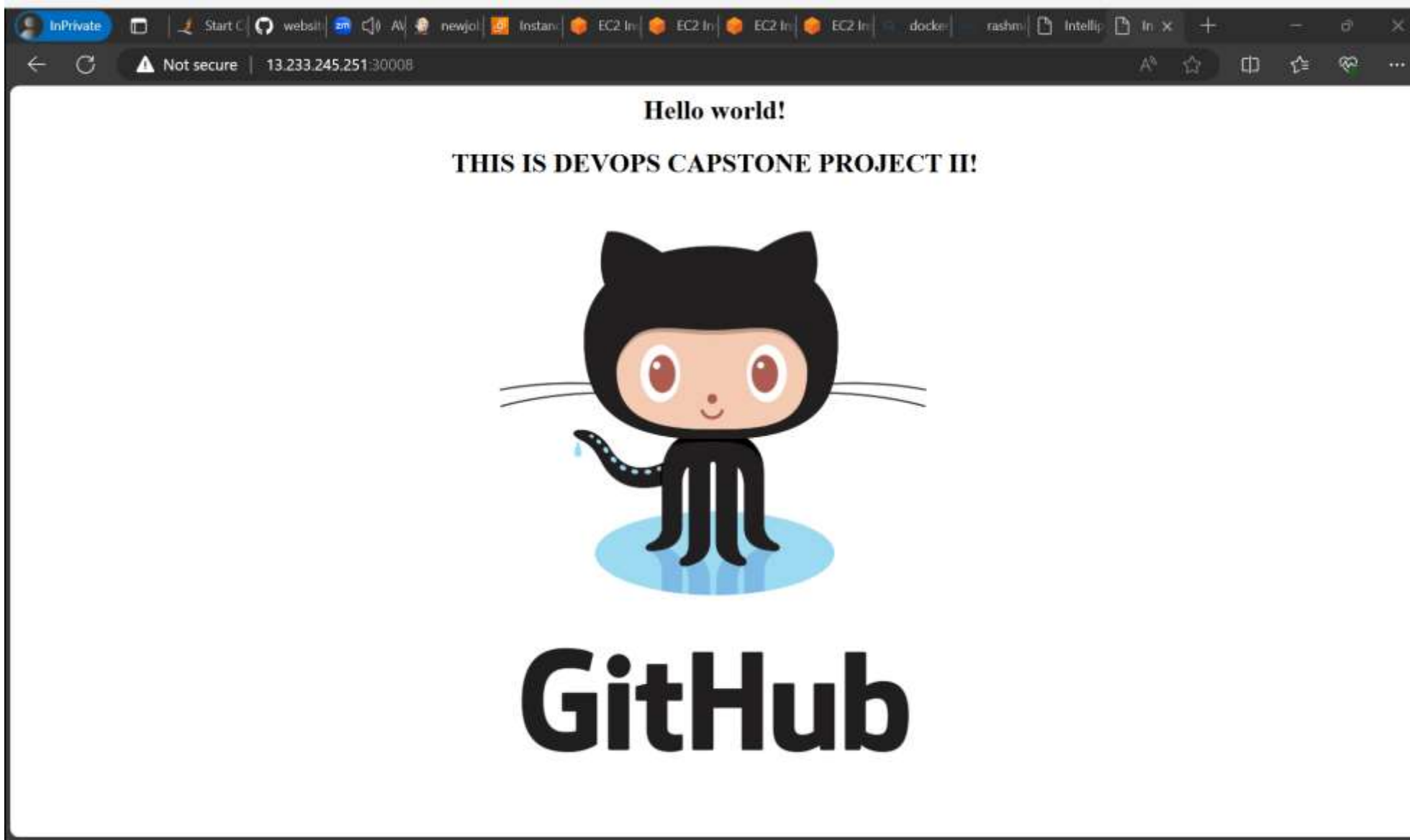
 Build  Configure

id	pipeline
#13	<div>Start Hello git docker kubernetes End</div> 
#12	<div>Start Hello git docker kubernetes End</div> 
#11	<div>Start Hello git docker kubernetes End</div> 

Jenkins Auto build job is running Successfully in m2-k8s-slave)









Jenkins Auto build job is running Successfully in m4-k8s-slave




Pipeline Console

← ↻ ⚠ Not secure | 13.201.79.254:8080/job/newjob/13/pipeline-console/


 **Jenkins**


    ras  log out


Dashboard > newjob > #13 > Pipeline Console


 < **Build #13**


Success 8 min 4 sec ago in 17 sec


 Hello


 git


 docker

 **kubernetes**




 Took 1.3 sec


 Success

 [View as plain text](#)




 **kubectl delete deploy nginx-deployment**


Shell Script

0.29 sec   




 **kubectl apply -f deployment.yaml**


Shell Script

0.29 sec   




 **kubectl delete service my-service**

Shell Script

0.29 sec   

 **kubectl apply -f service.yaml**

Shell Script

0.29 sec   

0 + kubectl apply -f service.yaml

1 service/my-service created

Jenkins 2.460