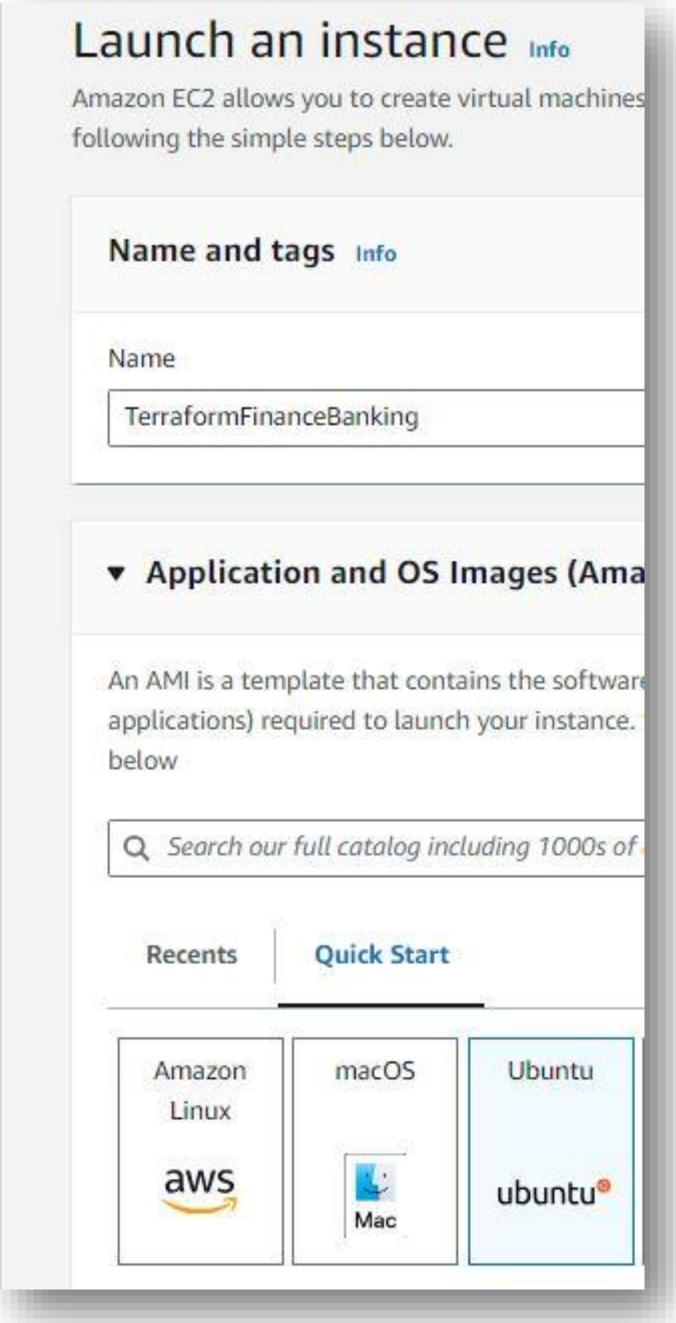
DevOps Certification Training
Certification Project – Finance Me
Banking and Finance Domain

by – Abhishek Jain

Launch an Terraform instance with in built command for insalling terraform into it.



Launch an instance Info

Amazon EC2 allows you to create virtual machines following the simple steps below.

Name and tags Info

Name

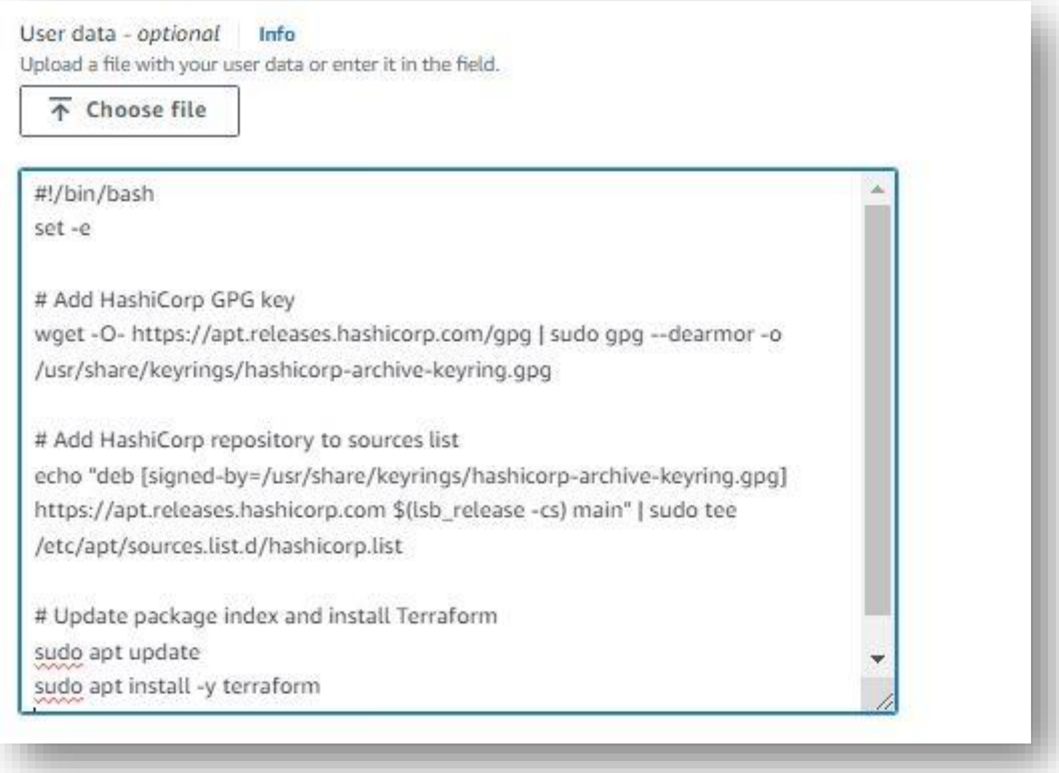
TerraformFinanceBanking

▼ Application and OS Images (Ama

An AMI is a template that contains the software applications) required to launch your instance. below

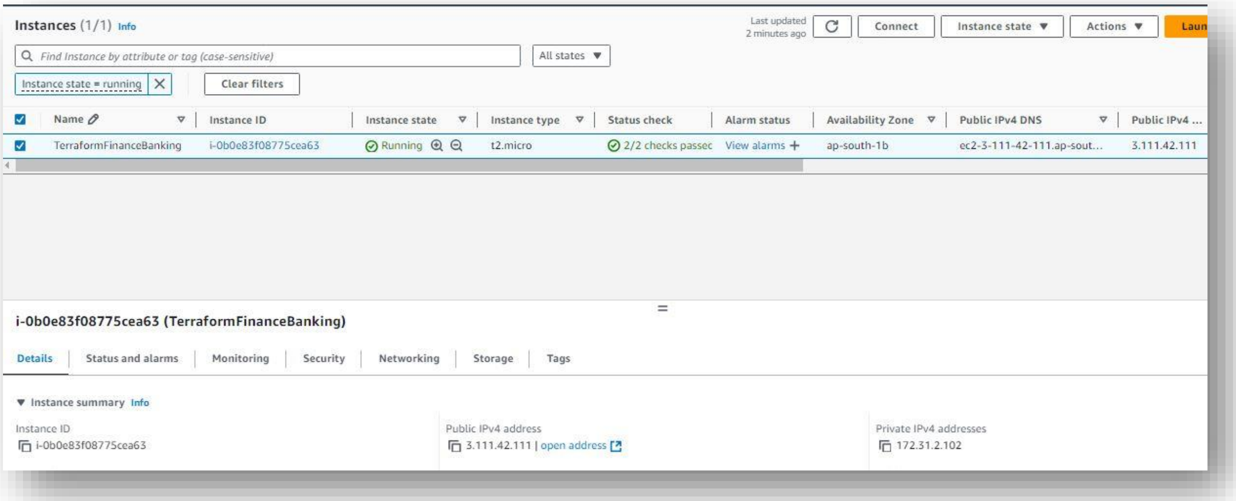
Search our full catalog including 1000s of

User Data Script :-



```
#!/bin/bash
set -e

# Add HashiCorp GPG key
wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o
/usr/share/keyrings/hashicorp-archive-keyring.gpg

# Add HashiCorp repository to sources list
echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg]
https://apt.releases.hashicorp.com $(lsb_release -cs) main" | sudo tee
/etc/apt/sources.list.d/hashicorp.list

# Update package index and install Terraform
sudo apt update
sudo apt install -y terraform
```

Terraform Instance Created :-



☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
☑	TerraformFinanceBanking	i-0b0e83f08775cea63	Running	t2.micro	2/2 checks passec	View alarms +	ap-south-1b	ec2-3-111-42-111.ap-sout...	3.111.42.111

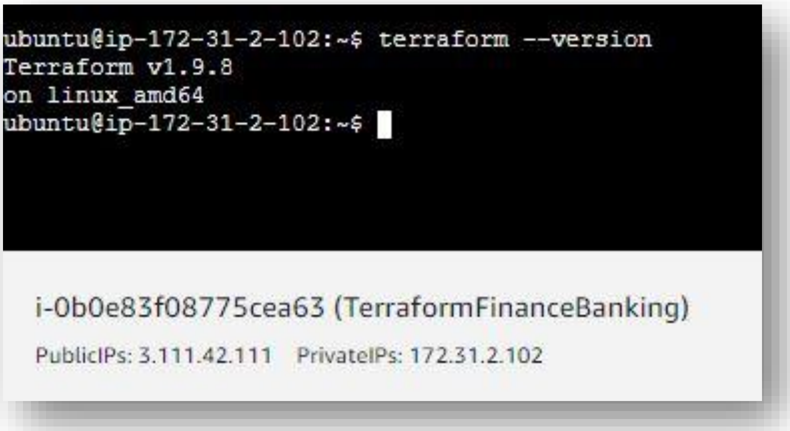
i-0b0e83f08775cea63 (TerraformFinanceBanking)

Instance ID: i-0b0e83f08775cea63

Public IPv4 address: 3.111.42.111 | open address

Private IPv4 addresses: 172.31.2.102

Here, to check ,Terraform pre-installed in Terraform instance :-



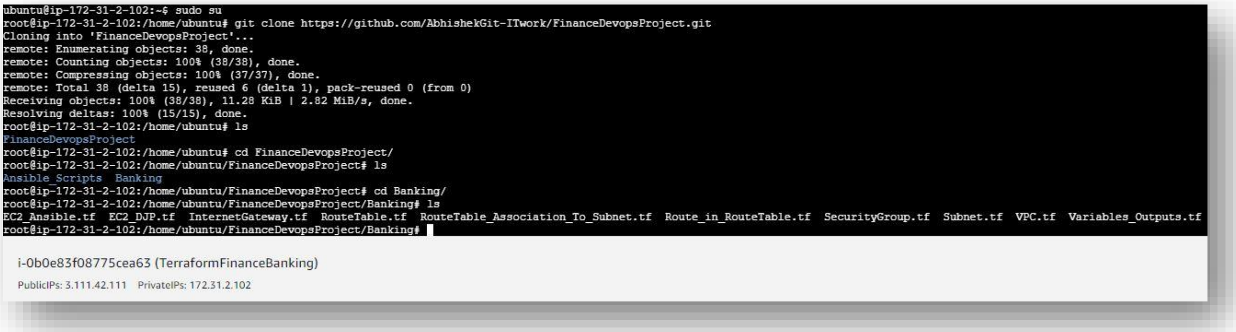
```
ubuntu@ip-172-31-2-102:~$ terraform --version
Terraform v1.9.8
on linux_amd64
ubuntu@ip-172-31-2-102:~$ 
```

i-0b0e83f08775cea63 (TerraformFinanceBanking)

PublicIPs: 3.111.42.111   PrivateIPs: 172.31.2.102

Now, using the git clone command, get the data of this git repository - **https://github.com/AbhishekGit-ITwork/FinanceDevopsProject.git**

Here, various terraform files are there **under Banking Directory** to create the 3 new EC2 machine for namely- **Docker_Jenkins_Prometheus(DJP), Ansible and Grafana** along with a set of 'new - vpc, subnet,RouteTable,Internet Gateway, Security Group' :-

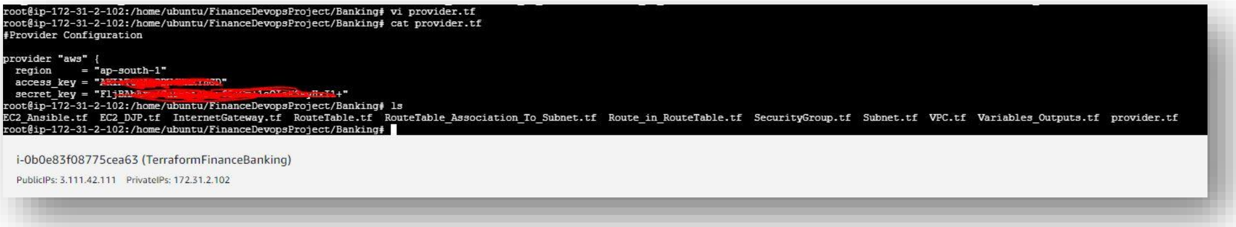


```
ubuntu@ip-172-31-2-102:~$ sudo su
root@ip-172-31-2-102:/home/ubuntu# git clone https://github.com/AbhishekGit-ITwork/FinanceDevopsProject.git
Cloning into 'FinanceDevopsProject'...
remote: Enumerating objects: 38, done.
remote: Counting objects: 100% (38/38), done.
remote: Compressing objects: 100% (37/37), done.
remote: Total 38 (delta 15), reused 6 (delta 1), pack-reused 0 (from 0)
Receiving objects: 100% (38/38), 11.28 KiB | 2.82 MiB/s, done.
Resolving deltas: 100% (15/15), done.
root@ip-172-31-2-102:/home/ubuntu# ls
FinanceDevopsProject
root@ip-172-31-2-102:/home/ubuntu# cd FinanceDevopsProject/
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject# ls
Ansible_Scripts  Banking
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject# cd Banking/
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking# ls
EC2_Ansible.tf  EC2_DJP.tf  InternetGateway.tf  RouteTable.tf  RouteTable_Association_To_Subnet.tf  Route_in_RouteTable.tf  SecurityGroup.tf  Subnet.tf  VPC.tf  Variables_Outputs.tf
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking# 
```

i-0b0e83f08775cea63 (TerraformFinanceBanking)

PublicIPs: 3.111.42.111   PrivateIPs: 172.31.2.102

Additionally , **manually** create a **provider.tf** file with necessary details (under this Banking directory.

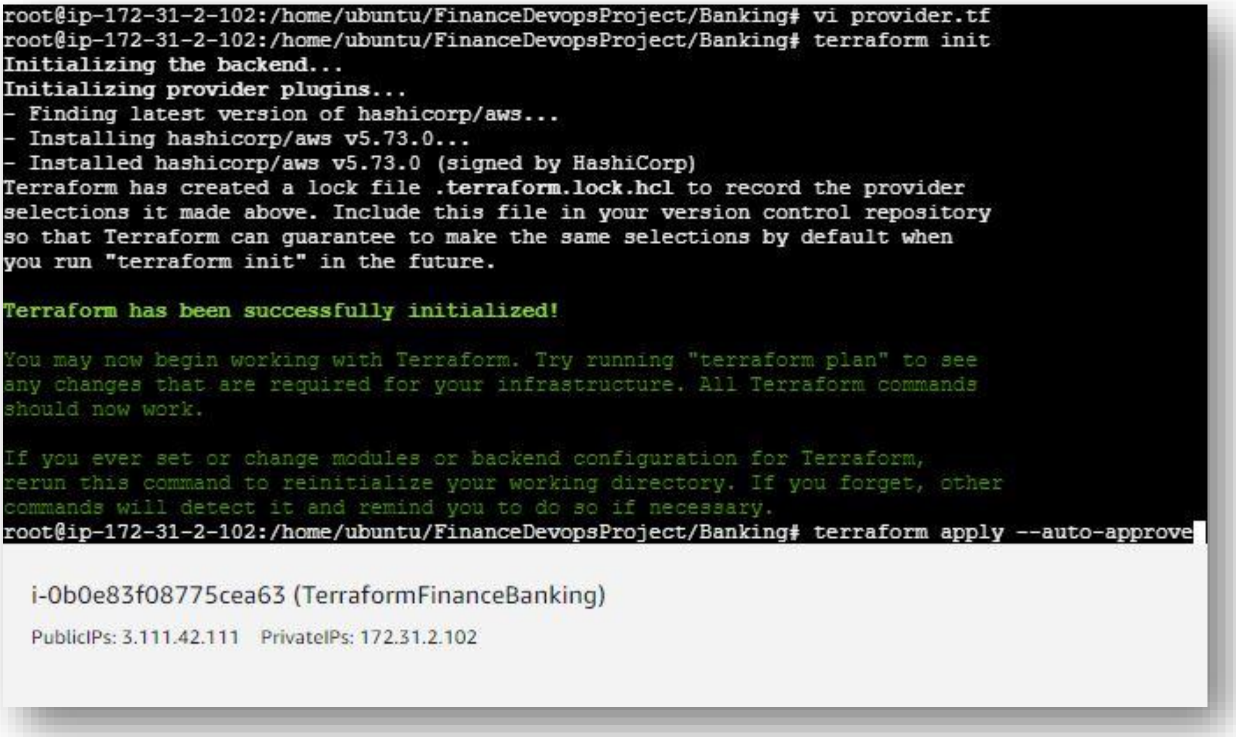


```
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking# vi provider.tf
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking# cat provider.tf
#Provider Configuration

provider "aws" {
  region     = "ap-south-1"
  access_key = "[redacted]"
  secret_key = "[redacted]"
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking# ls
EC2_Ansible.tf  EC2_DJP.tf  InternetGateway.tf  RouteTable.tf  RouteTable_Association_To_Subnet.tf  Route_in_RouteTable.tf  SecurityGroup.tf  Subnet.tf  VPC.tf  Variables_Outputs.tf  provider.tf
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking# 
```

i-0b0e83f08775cea63 (TerraformFinanceBanking)

PublicIPs: 3.111.42.111   PrivateIPs: 172.31.2.102

Now, run the **terraform init** command to initialize the terraform. After successful initiation, then run **terraform apply –auto-approve** command to apply changes to our infrastructure as defined in our Terraform configuration files without requiring manual confirmation.

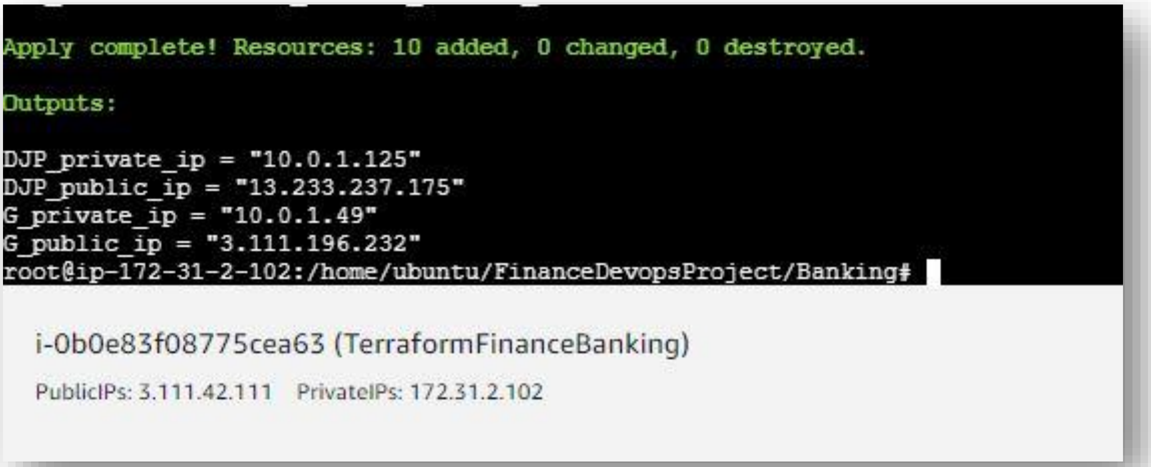


```
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking# vi provider.tf
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking# terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.73.0...
- Installed hashicorp/aws v5.73.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking# terraform apply --auto-approve
```

i-0b0e83f08775cea63 (TerraformFinanceBanking)

PublicIPs: 3.111.42.111   PrivateIPs: 172.31.2.102

Applied Complete message :-

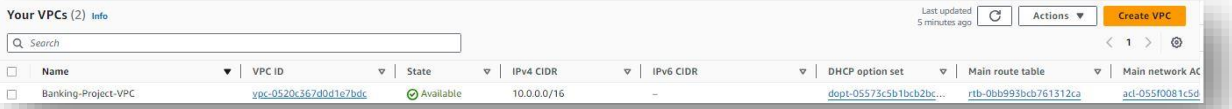


```
Apply complete! Resources: 10 added, 0 changed, 0 destroyed.

Outputs:

DJP_private_ip = "10.0.1.125"
DJP_public_ip = "13.233.237.175"
G_private_ip = "10.0.1.49"
G_public_ip = "3.111.196.232"
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking#
```

i-0b0e83f08775cea63 (TerraformFinanceBanking)

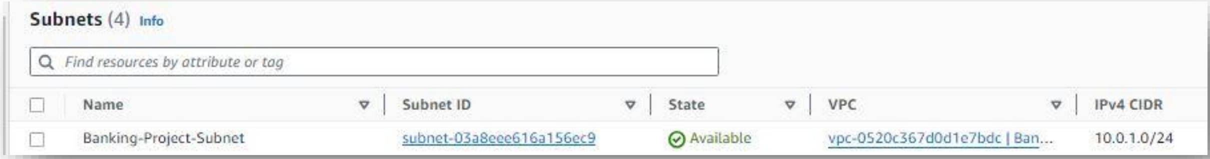
PublicIPs: 3.111.42.111   PrivateIPs: 172.31.2.102

We can see now, Created <u>Infrastructure</u> with – **'VPC, Subnet,RouteTable,Internet Gateway, Security Group'**



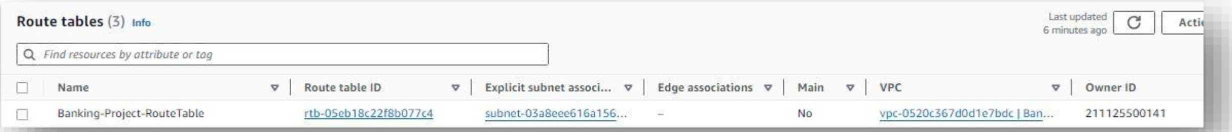
Your VPCs (2) Info

□	Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP option set	Main route table	Main network AC
□	Banking-Project-VPC	vpc-0520c367d0d1e7bdc	✓ Available	10.0.0.0/16	–	dopt-05573c5b1bcb2bc...	rtb-0bb993bcb761312ca	acl-055f0081c5d



Subnets (4) Info

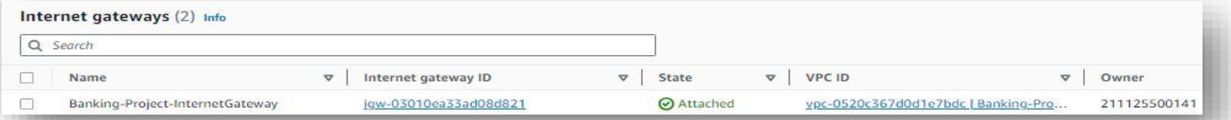
□	Name	Subnet ID	State	VPC	IPv4 CIDR
□	Banking-Project-Subnet	subnet-03a8eee616a156ec9	✓ Available	vpc-0520c367d0d1e7bdc | Ban...	10.0.1.0/24



Route tables (3) Info

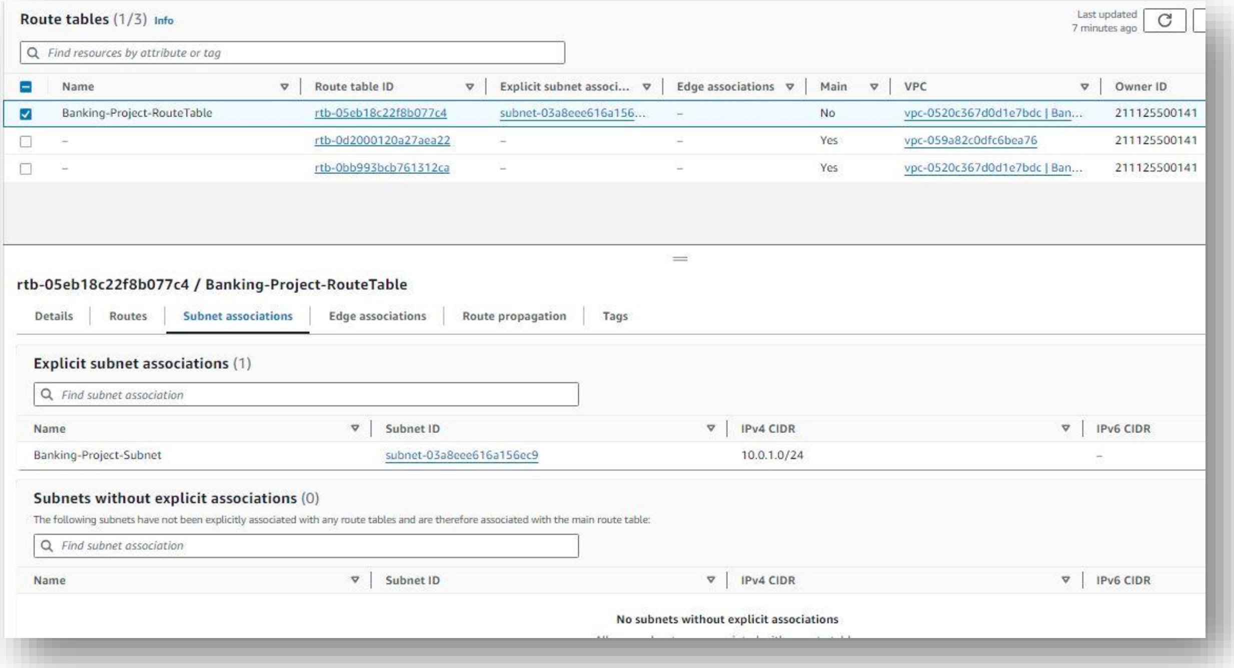
Last updated 6 minutes ago
Acti
Find resources by attribute or tag

□	Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	Owner ID
□	Banking-Project-RouteTable	rtb-05eb18c22f8b077c4	subnet-03a8eee616a156...	–	No	vpc-0520c367d0d1e7bdc | Ban...	211125500141



Internet gateways (2) Info

□	Name	Internet gateway ID	State	VPC ID	Owner
□	Banking-Project-InternetGateway	igw-03010ea33ad08d821	✓ Attached	vpc-0520c367d0d1e7bdc | Banking-Pro...	211125500141



Route tables (1/3) Info

Find resources by attribute or tag

	Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	Owner ID
☑	Banking-Project-RouteTable	rtb-05eb18c22f8b077c4	subnet-03a8eee616a156...	–	No	vpc-0520c367d0d1e7bdc | Ban...	211125500141
□	–	rtb-0d2000120a27aea22	–	–	Yes	vpc-059a82c0dfc6bea76	211125500141
□	–	rtb-0bb993bcb761312ca	–	–	Yes	vpc-0520c367d0d1e7bdc | Ban...	211125500141

rtb-05eb18c22f8b077c4 / Banking-Project-RouteTable

Explicit subnet associations (1)

Find subnet association

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
Banking-Project-Subnet	subnet-03a8eee616a156ec9	10.0.1.0/24	–

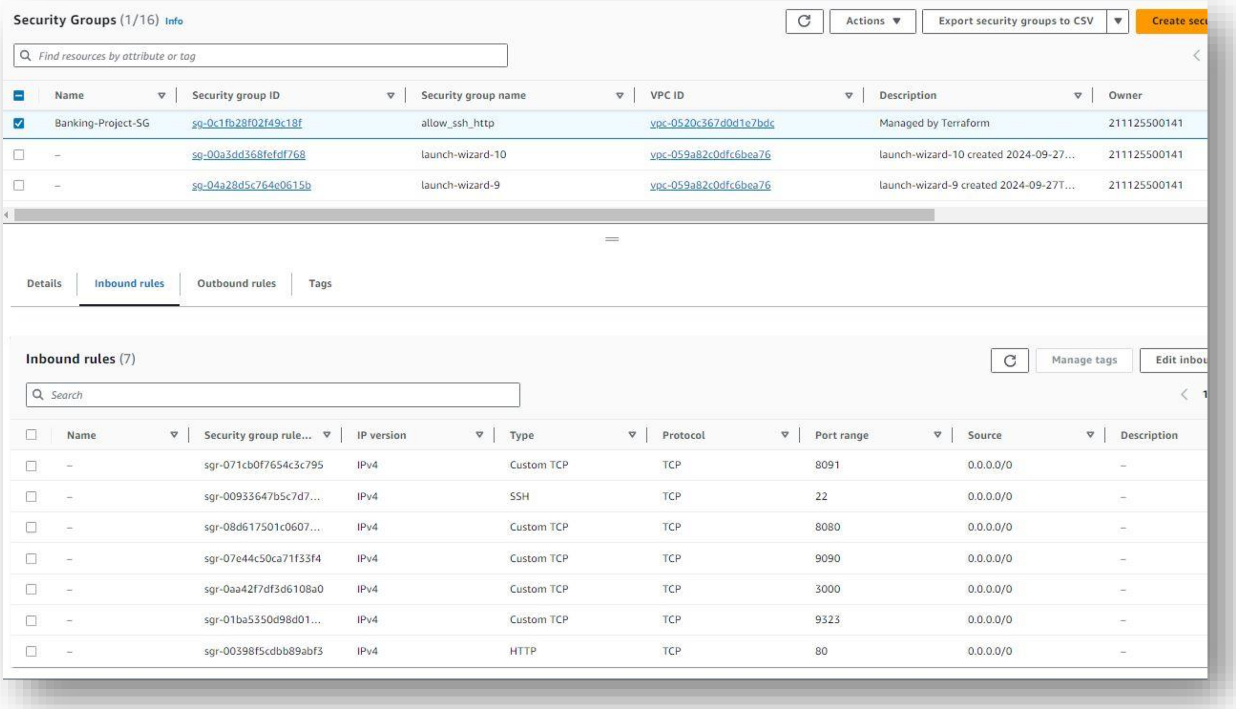
Subnets without explicit associations (0)

The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

Find subnet association

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR

No subnets without explicit associations



Security Groups (1/16) Info

Find resources by attribute or tag

	Name	Security group ID	Security group name	VPC ID	Description	Owner
☑	Banking-Project-SG	sg-0c1fb28f02f49c18f	allow_ssh_http	vpc-0520c367d0d1e7bdc	Managed by Terraform	211125500141
□	–	sg-00a3dd368fefdf768	launch-wizard-10	vpc-059a82c0dfc6bea76	launch-wizard-10 created 2024-09-27...	211125500141
□	–	sg-04a28d5c764e0615b	launch-wizard-9	vpc-059a82c0dfc6bea76	launch-wizard-9 created 2024-09-27T...	211125500141

Inbound rules (7)

Search

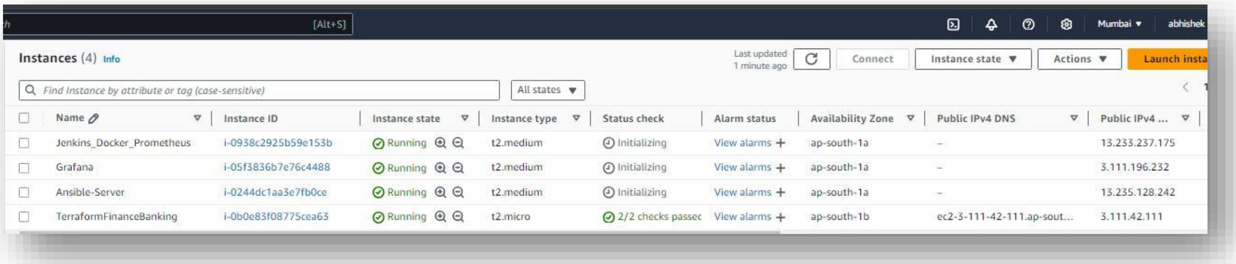
□	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
□	–	sgr-071cb0f7654c3c795	IPv4	Custom TCP	TCP	8091	0.0.0.0/0	–
□	–	sgr-00933647b5c7d7...	IPv4	SSH	TCP	22	0.0.0.0/0	–
□	–	sgr-08d617501c0607...	IPv4	Custom TCP	TCP	8080	0.0.0.0/0	–
□	–	sgr-07e44c50ca71f33f4	IPv4	Custom TCP	TCP	9090	0.0.0.0/0	–
□	–	sgr-0aa42f7df3d6108a0	IPv4	Custom TCP	TCP	3000	0.0.0.0/0	–
□	–	sgr-01ba5350d98d01...	IPv4	Custom TCP	TCP	9323	0.0.0.0/0	–
□	–	sgr-00398f5cdbb89abf3	IPv4	HTTP	TCP	80	0.0.0.0/0	–

We can also see now, **3 new Instances Created** namely :–

1.**Jenkins_Docker_Prometheus**

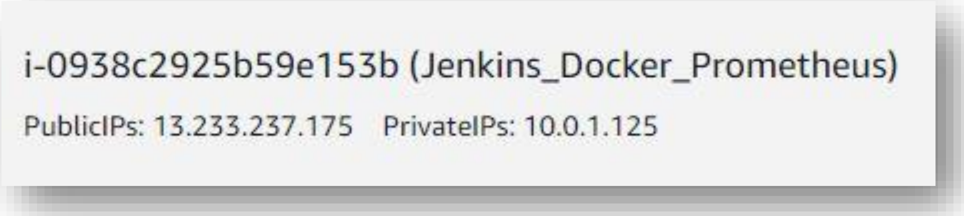
2. Grafana

3. Ansible



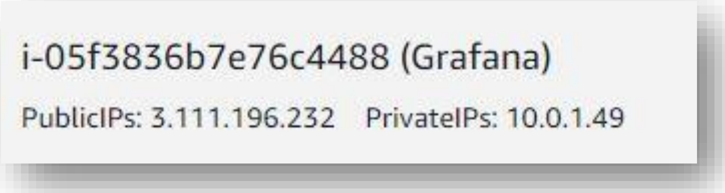
Instances (4)

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
□	Jenkins_Docker_Prometheus	i-0938c2925b59e153b	✓ Running	t2.medium	Initializing	View alarms +	ap-south-1a	–	13.233.237.175
□	Grafana	i-05f3836b7e76c4488	✓ Running	t2.medium	Initializing	View alarms +	ap-south-1a	–	3.111.196.232
□	Ansible-Server	i-0244dc1aa3e7fb0ce	✓ Running	t2.medium	Initializing	View alarms +	ap-south-1a	–	13.235.128.242
□	TerraformFinanceBanking	i-0b0e83f08775cea63	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	ap-south-1b	ec2-3-111-42-111.ap-sout...	3.111.42.111



i-0938c2925b59e153b (Jenkins_Docker_Prometheus)

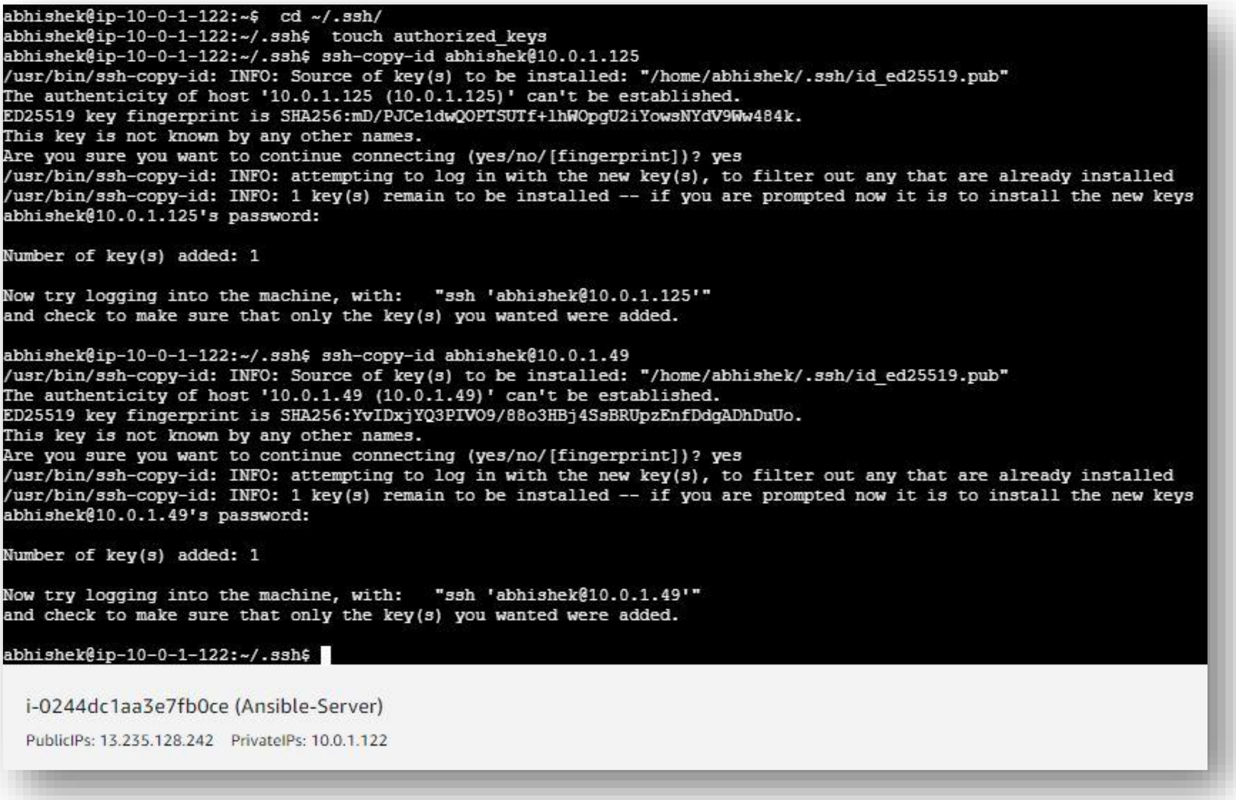
PublicIPs: 13.233.237.175   PrivateIPs: 10.0.1.125



i-05f3836b7e76c4488 (Grafana)

PublicIPs: 3.111.196.232   PrivateIPs: 10.0.1.49

Now, do **ssh** to **connect the Ansible server with - Jenkins_Docker_Prometheus and Grafana Machine**, this is to install our SSH public key on a remote server's authorized keys list. This allows us to log in to the server using SSH without needing to enter a password each time.



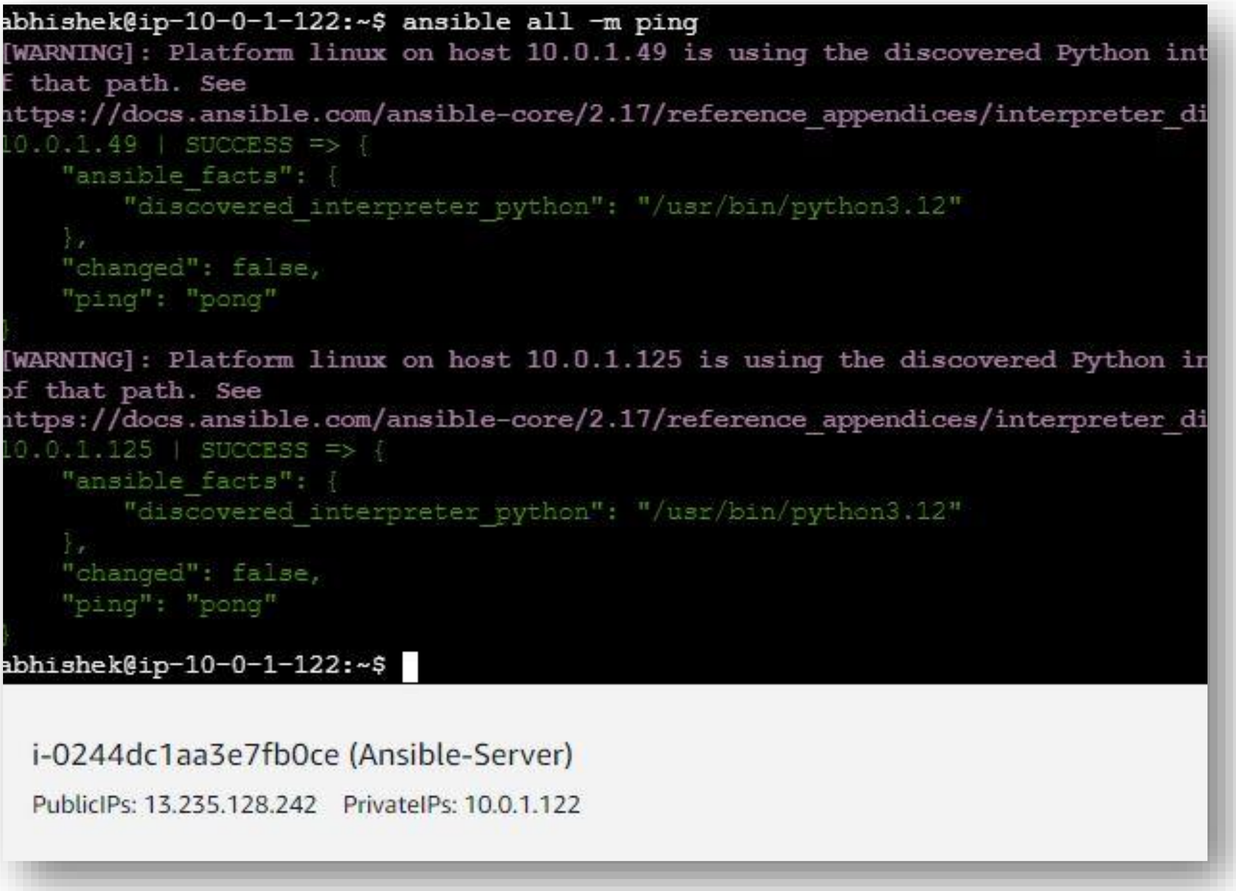
```
abhishek@ip-10-0-1-122:~$  cd ~/.ssh/
abhishek@ip-10-0-1-122:~/.ssh$  touch authorized_keys
abhishek@ip-10-0-1-122:~/.ssh$ ssh-copy-id abhishek@10.0.1.125
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/abhishek/.ssh/id_ed25519.pub"
The authenticity of host '10.0.1.125 (10.0.1.125)' can't be established.
ED25519 key fingerprint is SHA256:mD/PJCe1dwQOPTSUTf+1hWOpgU2iYowsNYdV9Ww484k.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
abhishek@10.0.1.125's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'abhishek@10.0.1.125'"
and check to make sure that only the key(s) you wanted were added.

abhishek@ip-10-0-1-122:~/.ssh$ ssh-copy-id abhishek@10.0.1.49
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/abhishek/.ssh/id_ed25519.pub"
The authenticity of host '10.0.1.49 (10.0.1.49)' can't be established.
ED25519 key fingerprint is SHA256:YvIDxjYQ3PIVO9/88o3HBj4SsBRUpzEnfDdgADhDuUo.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
abhishek@10.0.1.49's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'abhishek@10.0.1.49'"
and check to make sure that only the key(s) you wanted were added.

abhishek@ip-10-0-1-122:~/.ssh$
```

i-0244dc1aa3e7fb0ce (Ansible-Server)

PublicIPs: 13.235.128.242   PrivateIPs: 10.0.1.122

Now in ansible machine, run **ansible all -m ping** command for initial checks to ensure that all hosts in our inventory are accessible and ready for further Ansible operations. This command is a quick way to verify connectivity before i execute more complex playbooks or tasks.

Here, as shown below both machines are accessible :-

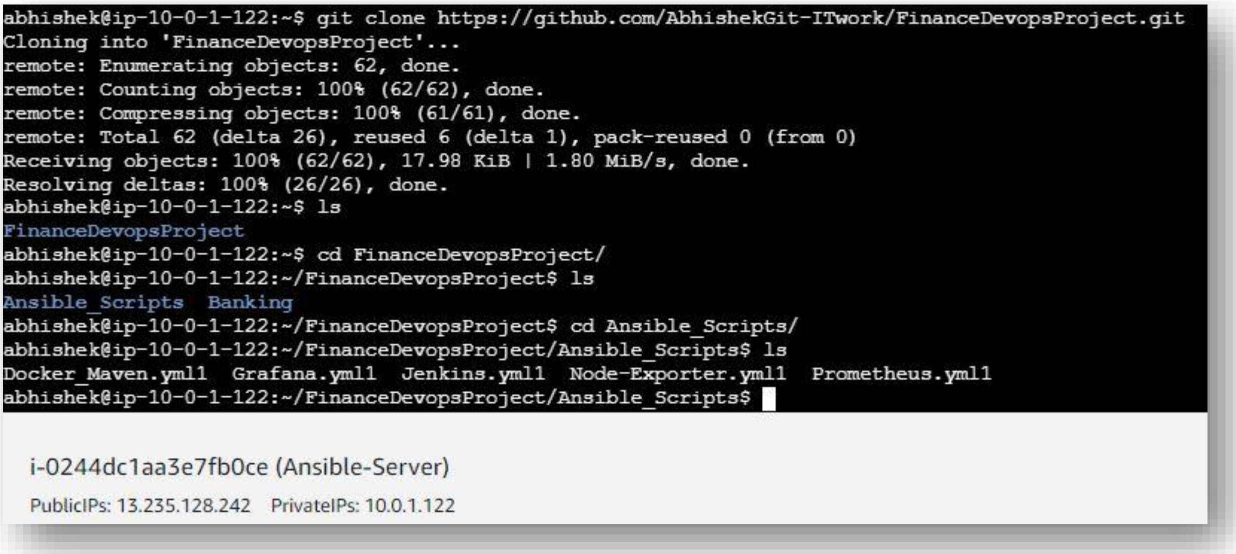


```
abhishek@ip-10-0-1-122:~$ ansible all -m ping
[WARNING]: Platform linux on host 10.0.1.49 is using the discovered Python int
f that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_di
10.0.1.49 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.12"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Platform linux on host 10.0.1.125 is using the discovered Python in
of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_di
10.0.1.125 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.12"
    },
    "changed": false,
    "ping": "pong"
}
abhishek@ip-10-0-1-122:~$
```

i-0244dc1aa3e7fb0ce (Ansible-Server)

PublicIPs: 13.235.128.242 PrivateIPs: 10.0.1.122

Now, clone this git repository (**https://github.com/AbhishekGit-ITwork/FinanceDevopsProject.git**) in ansible machine to get the playbook YML files.

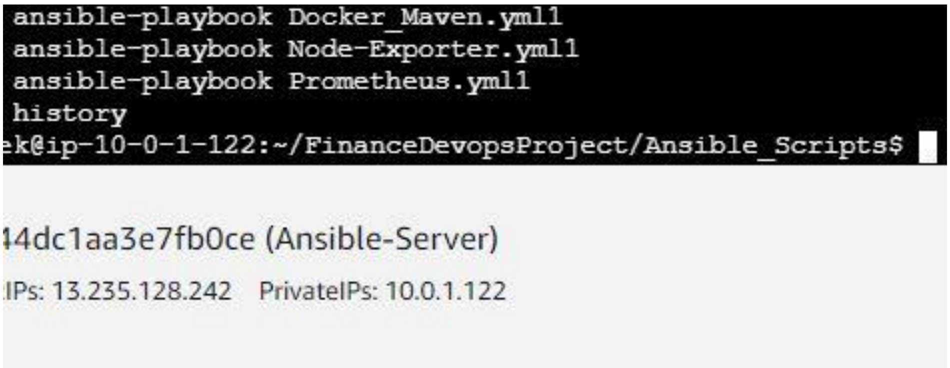


```
abhishek@ip-10-0-1-122:~$ git clone https://github.com/AbhishekGit-ITwork/FinanceDevopsProject.git
Cloning into 'FinanceDevopsProject'...
remote: Enumerating objects: 62, done.
remote: Counting objects: 100% (62/62), done.
remote: Compressing objects: 100% (61/61), done.
remote: Total 62 (delta 26), reused 6 (delta 1), pack-reused 0 (from 0)
Receiving objects: 100% (62/62), 17.98 KiB | 1.80 MiB/s, done.
Resolving deltas: 100% (26/26), done.
abhishek@ip-10-0-1-122:~$ ls
FinanceDevopsProject
abhishek@ip-10-0-1-122:~$ cd FinanceDevopsProject/
abhishek@ip-10-0-1-122:~/FinanceDevopsProject$ ls
Ansible_Scripts  Banking
abhishek@ip-10-0-1-122:~/FinanceDevopsProject$ cd Ansible_Scripts/
abhishek@ip-10-0-1-122:~/FinanceDevopsProject/Ansible_Scripts$ ls
Docker_Maven.yml1  Grafana.yml1  Jenkins.yml1  Node-Exporter.yml1  Prometheus.yml1
abhishek@ip-10-0-1-122:~/FinanceDevopsProject/Ansible_Scripts$
```

i-0244dc1aa3e7fb0ce (Ansible-Server)

PublicIPs: 13.235.128.242 PrivateIPs: 10.0.1.122

Now , run the **ansible-playbook *filename*.yml** to execute the tasks defined in our Ansible playbook to automating various aspects of configuration management or deployment across the specified hosts.

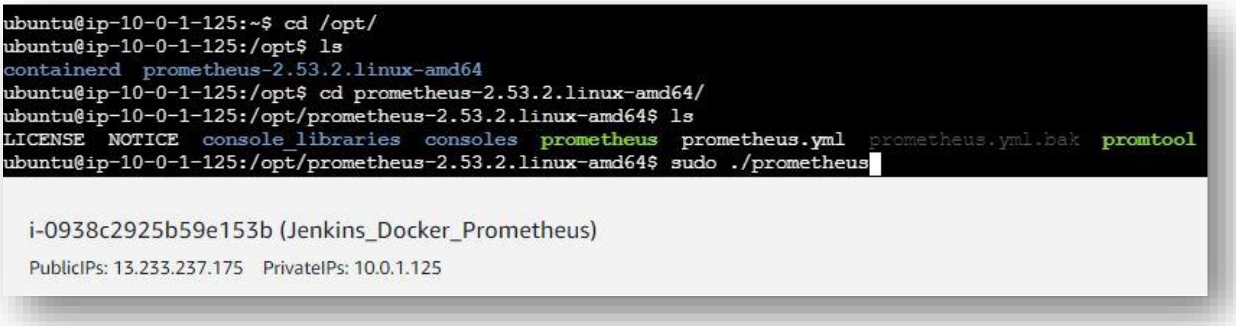


```
ansible-playbook Docker_Maven.yml1
ansible-playbook Node-Exporter.yml1
ansible-playbook Prometheus.yml1
history
ek@ip-10-0-1-122:~/FinanceDevopsProject/Ansible_Scripts$
```

14dc1aa3e7fb0ce (Ansible-Server)

IPs: 13.235.128.242 PrivateIPs: 10.0.1.122

Now , run the sudo **./prometheus** command to execute the Prometheus server binary, allowing me to start monitoring my systems and services by scraping metrics from various endpoints defined in the configuration file.

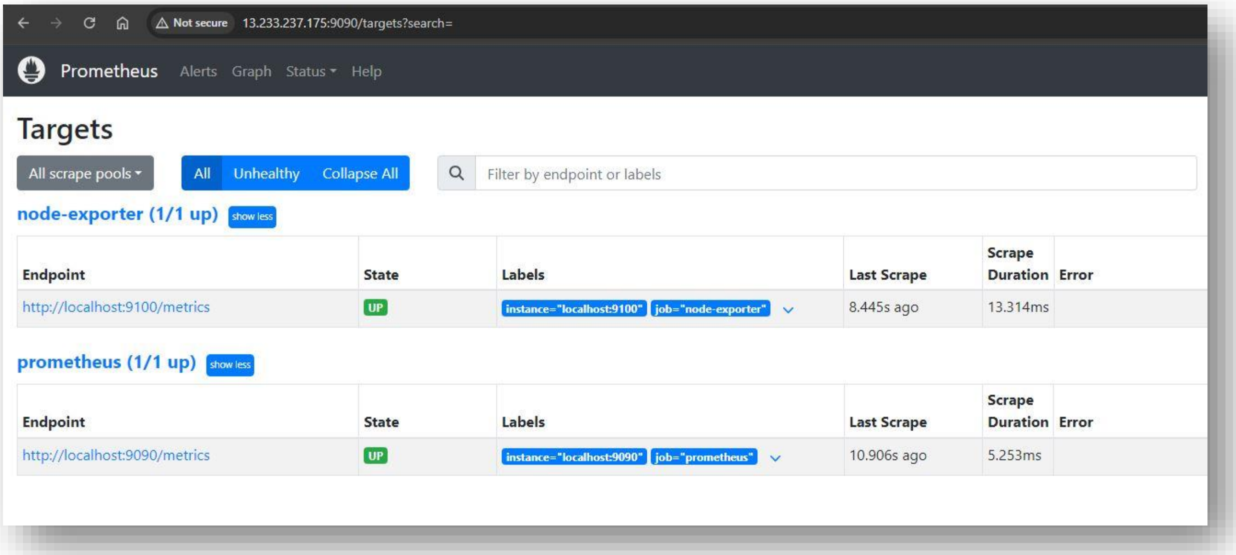


```
ubuntu@ip-10-0-1-125:~$ cd /opt/
ubuntu@ip-10-0-1-125:/opt$ ls
containerd  prometheus-2.53.2.linux-amd64
ubuntu@ip-10-0-1-125:/opt$ cd prometheus-2.53.2.linux-amd64/
ubuntu@ip-10-0-1-125:/opt/prometheus-2.53.2.linux-amd64$ ls
LICENSE  NOTICE  console_libraries  consoles  prometheus  prometheus.yml  prometheus.yml.bak  promtool
ubuntu@ip-10-0-1-125:/opt/prometheus-2.53.2.linux-amd64$ sudo ./prometheus
```

i-0938c2925b59e153b (Jenkins_Docker_Prometheus)

PublicIPs: 13.233.237.175   PrivateIPs: 10.0.1.125

Prometheus and node exporter running successfully :-



Targets

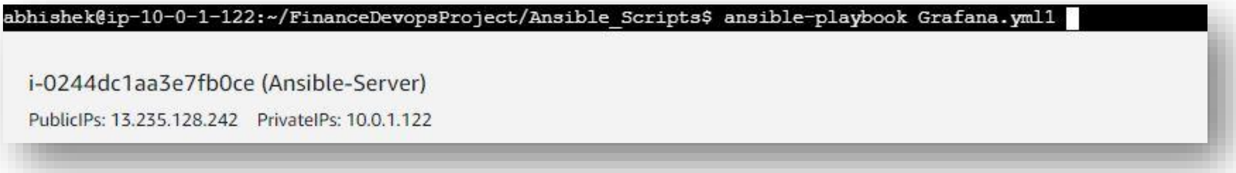
node-exporter (1/1 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9100/metrics	UP	instance="localhost:9100" job="node-exporter"	8.445s ago	13.314ms	

prometheus (1/1 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	10.906s ago	5.253ms	

Now run , Grafana yml file in Ansible Server to install Grafana in the Grafana Server

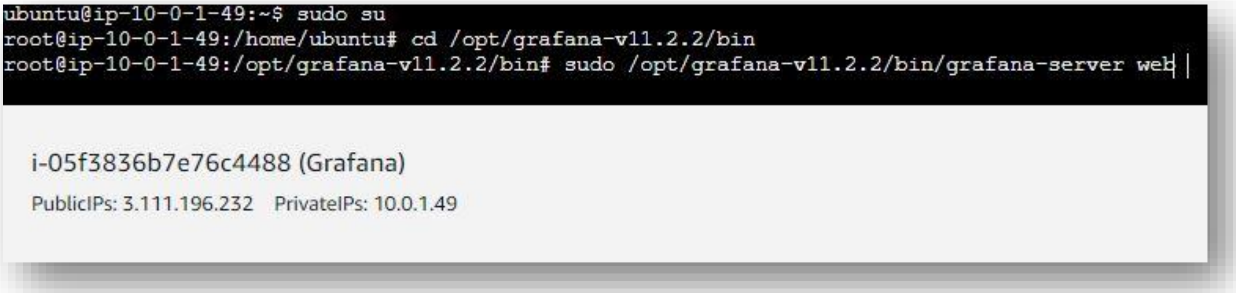


```
abhishek@ip-10-0-1-122:~/FinanceDevopsProject/Ansible_Scripts$ ansible-playbook Grafana.yml1
```

i-0244dc1aa3e7fb0ce (Ansible-Server)

PublicIPs: 13.235.128.242   PrivateIPs: 10.0.1.122

As shown below, in Grafana machine, start the Grafana server with elevated privileges, enabling the web interface for data visualization and monitoring :-

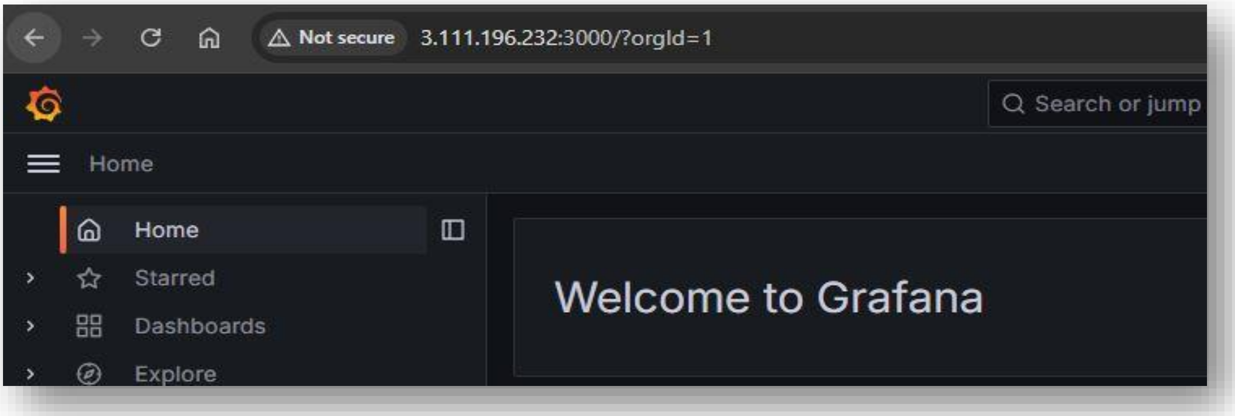


```
ubuntu@ip-10-0-1-49:~$ sudo su
root@ip-10-0-1-49:/home/ubuntu# cd /opt/grafana-v11.2.2/bin
root@ip-10-0-1-49:/opt/grafana-v11.2.2/bin# sudo /opt/grafana-v11.2.2/bin/grafana-server web
```

i-05f3836b7e76c4488 (Grafana)

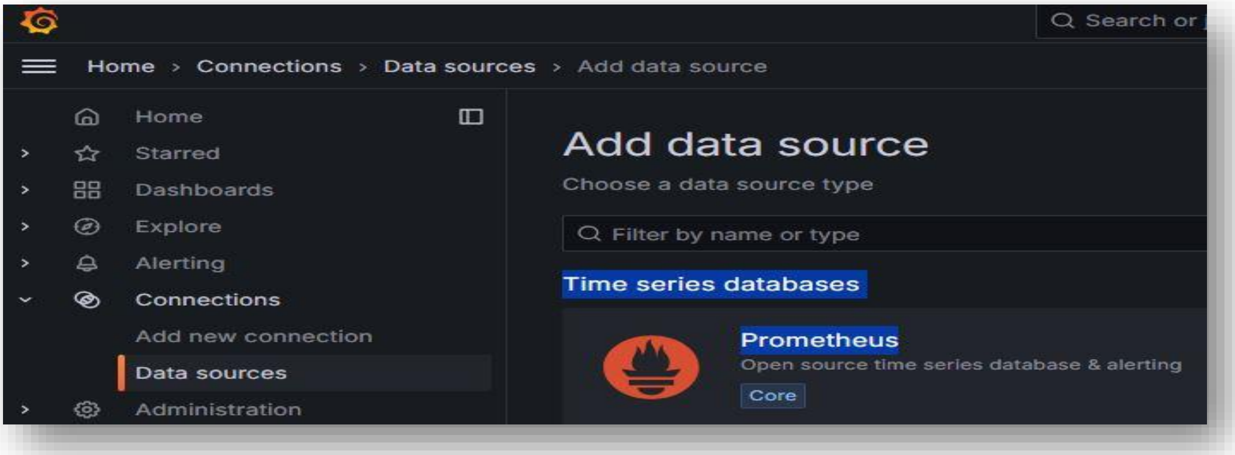
PublicIPs: 3.111.196.232 PrivateIPs: 10.0.1.49

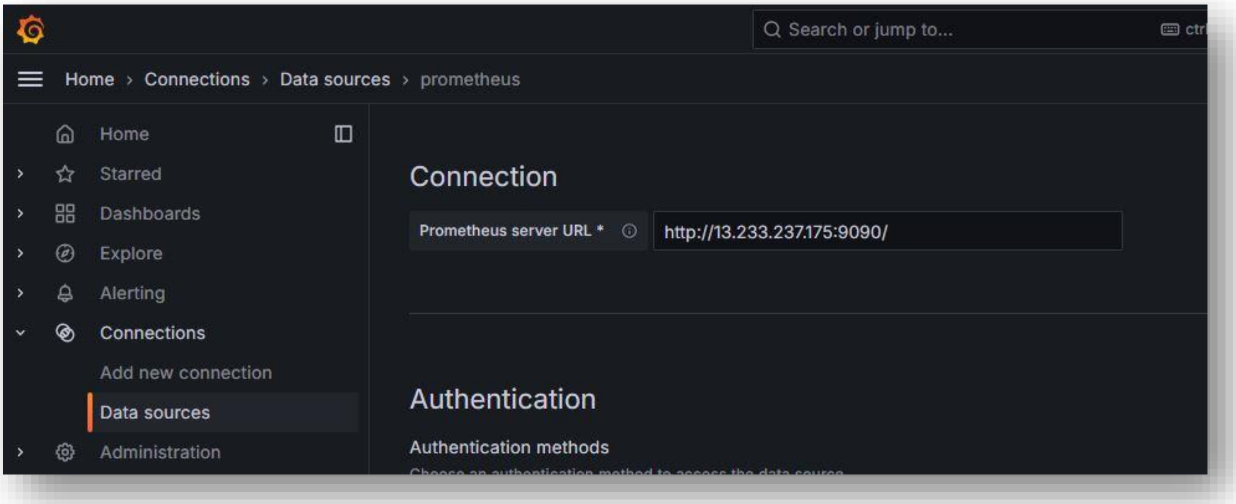
Grafana started and running successfully :-



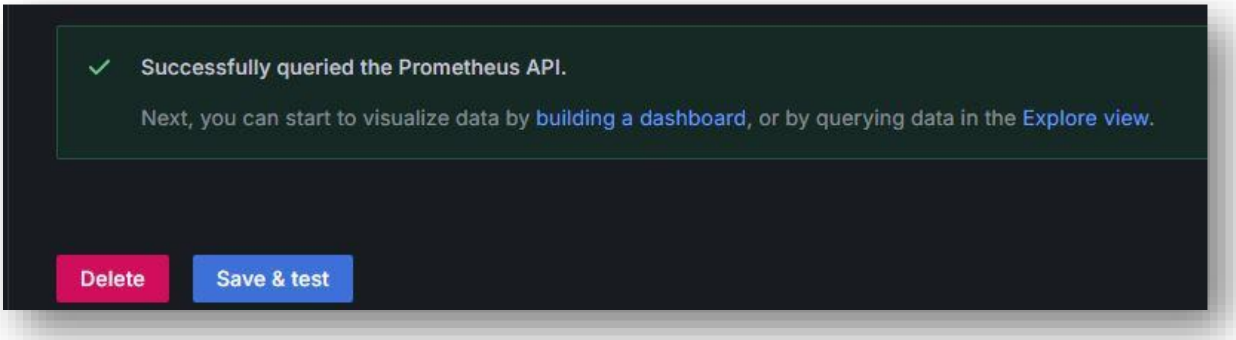
Not secure 3.111.196.232:3000/?orgId=1

Add Prometheus as the data source for visualization in Grafana :-





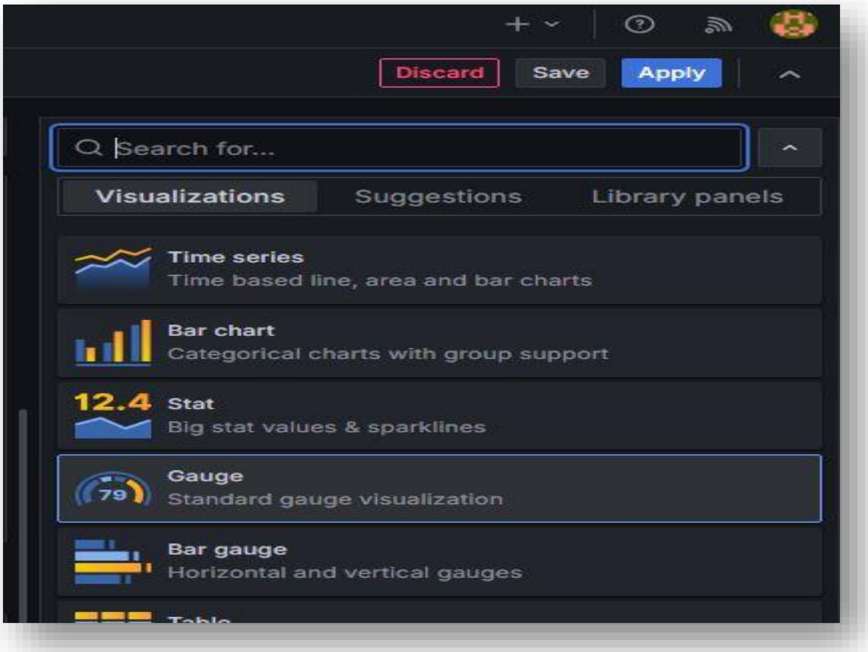
Prometheus server URL * http://13.233.237.175:9090/



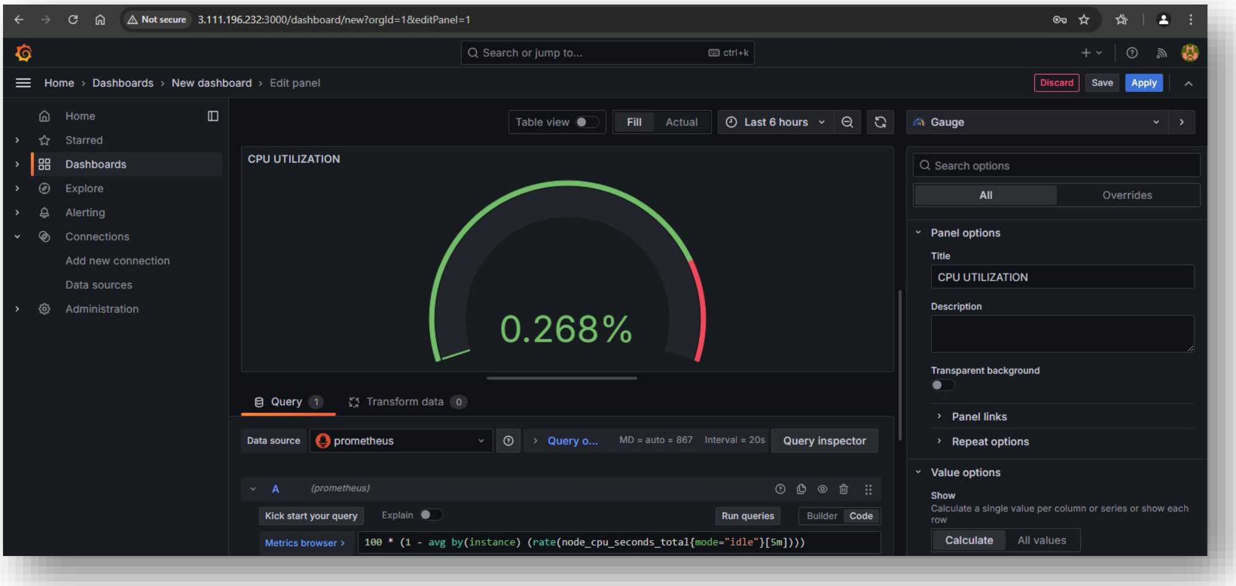
Successfully queried the Prometheus API.

Next, you can start to visualize data by building a dashboard, or by querying data in the Explore view.

Here, chose Visualization type - Gauge


Discard
Save
Apply
Search for...
Visualizations
Suggestions
Library panels
Time series
Time based line, area and bar charts
Bar chart
Categorical charts with group support
12.4
Stat
Big stat values & sparklines
79
Gauge
Standard gauge visualization
Bar gauge
Horizontal and vertical gauges
Table

CPU UTILIZATION -



CPU UTILIZATION

0.268%

Title: CPU UTILIZATION

Description:

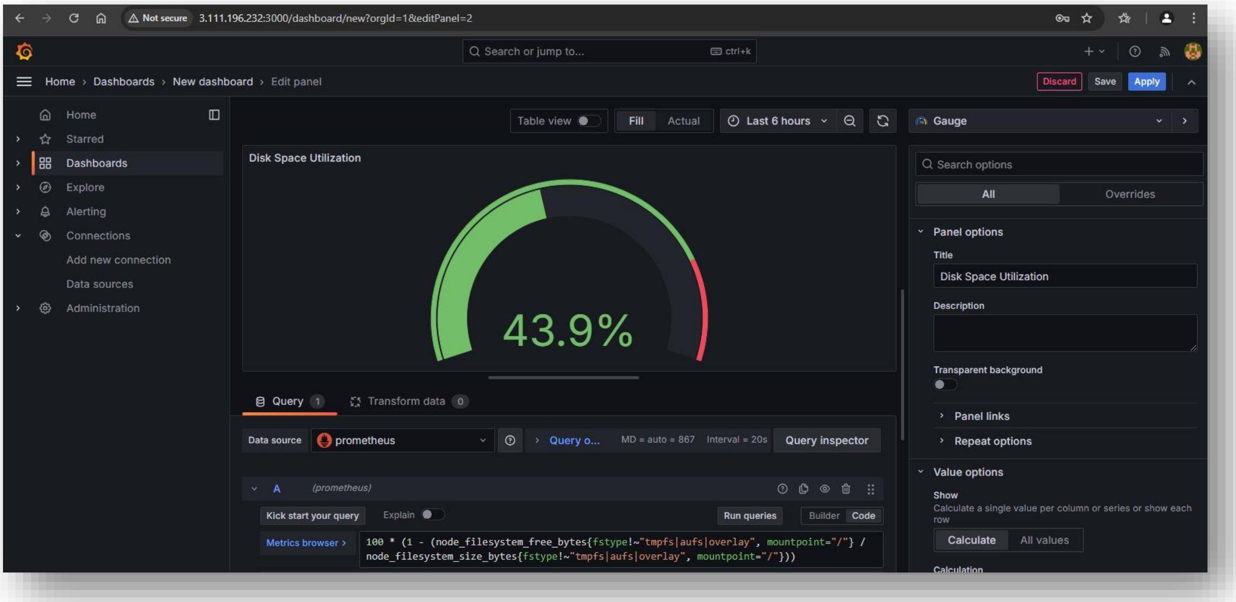
Transparent background

Show: Calculate a single value per column or series or show each row

Data source: prometheus

```
100 * (1 - avg by(instance) (rate(node_cpu_seconds_total{mode="idle"}[5m])))
```

Disk Space Utilization :-



Disk Space Utilization

43.9%

Title: Disk Space Utilization

Description:

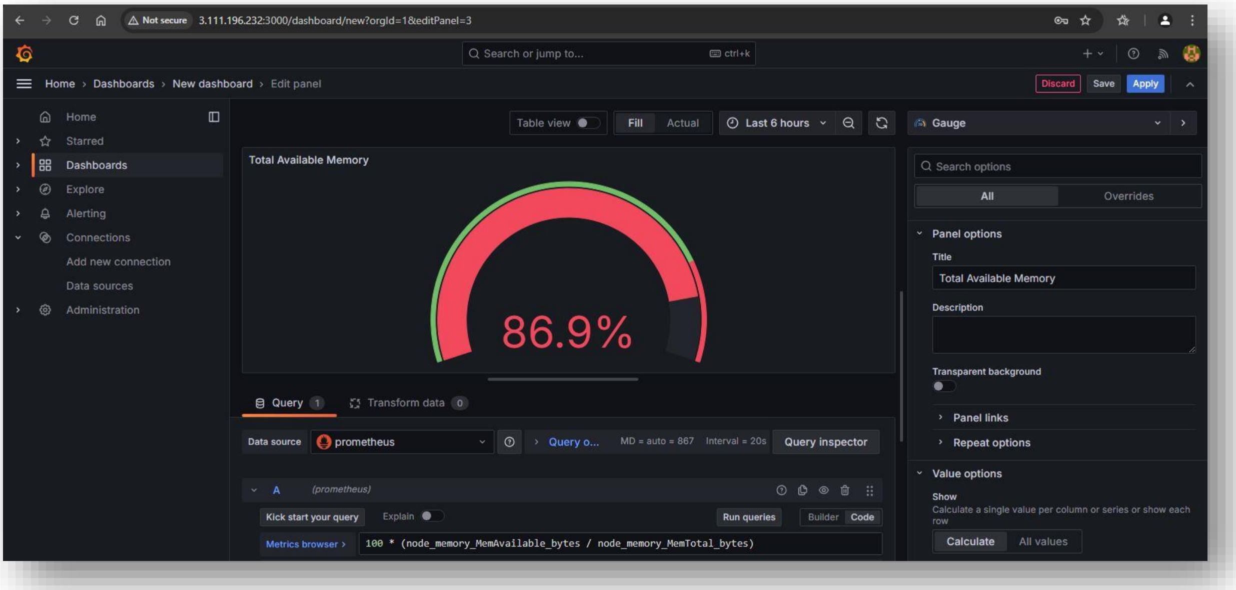
Transparent background

Show: Calculate a single value per column or series or show each row

Data source: prometheus

```
100 * (1 - (node_filesystem_free_bytes{fstype!~"tmpfs|aufs|overlay", mountpoint="/"} /
node_filesystem_size_bytes{fstype!~"tmpfs|aufs|overlay", mountpoint="/"}))
```

Total Available Memory :-



Data source: prometheus

Query o... MD = auto = 867 Interval = 20s

A (prometheus)

```
100 * (node_memory_MemAvailable_bytes / node_memory_MemTotal_bytes)
```

Panel options

Title: Total Available Memory

Description:

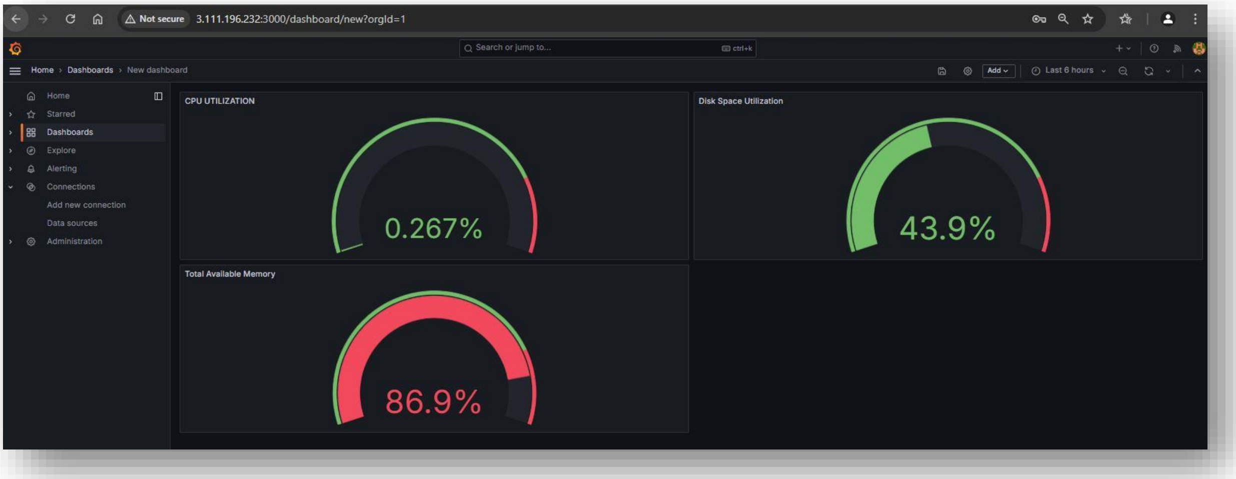
Transparent background

Value options

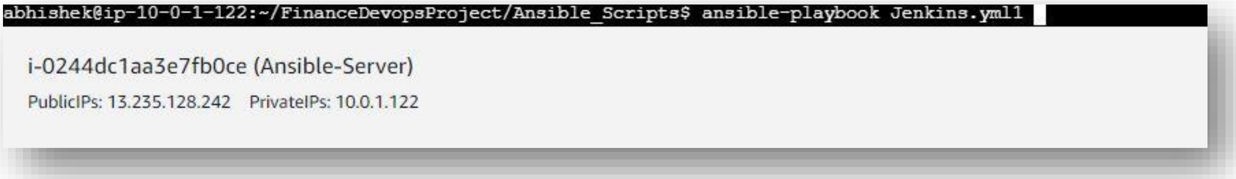
Show: Calculate a single value per column or series or show each row

Calculate | All values

CPU Utilization , Disk Space Utilization and Total Available Memory together **:-**



Now , execute the Jenkins yml file :-

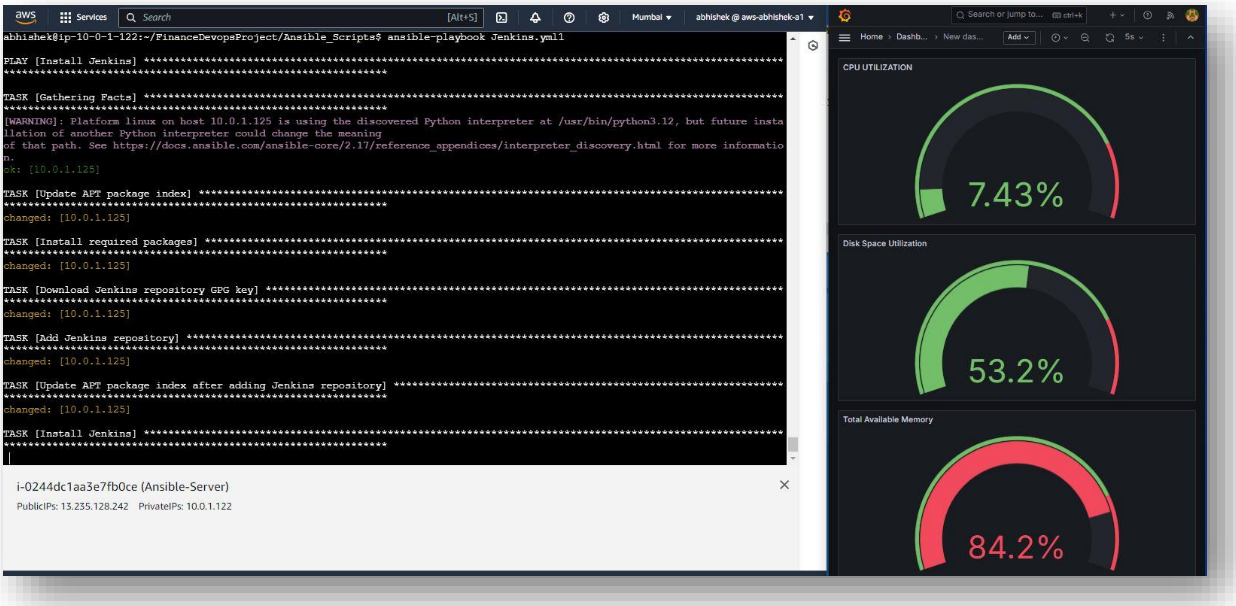


```
abhishek@ip-10-0-1-122:~/FinanceDevopsProject/Ansible_Scripts$ ansible-playbook Jenkins.yml1
```

i-0244dc1aa3e7fb0ce (Ansible-Server)

PublicIPs: 13.235.128.242   PrivateIPs: 10.0.1.122

Now, in Continuous Monitoring Grafana server , here we can see the real time fluctuations in the **CPU Utilization , Disk Space Utilization and Total Available Memory** metrics while executing Jenkins yml file :-

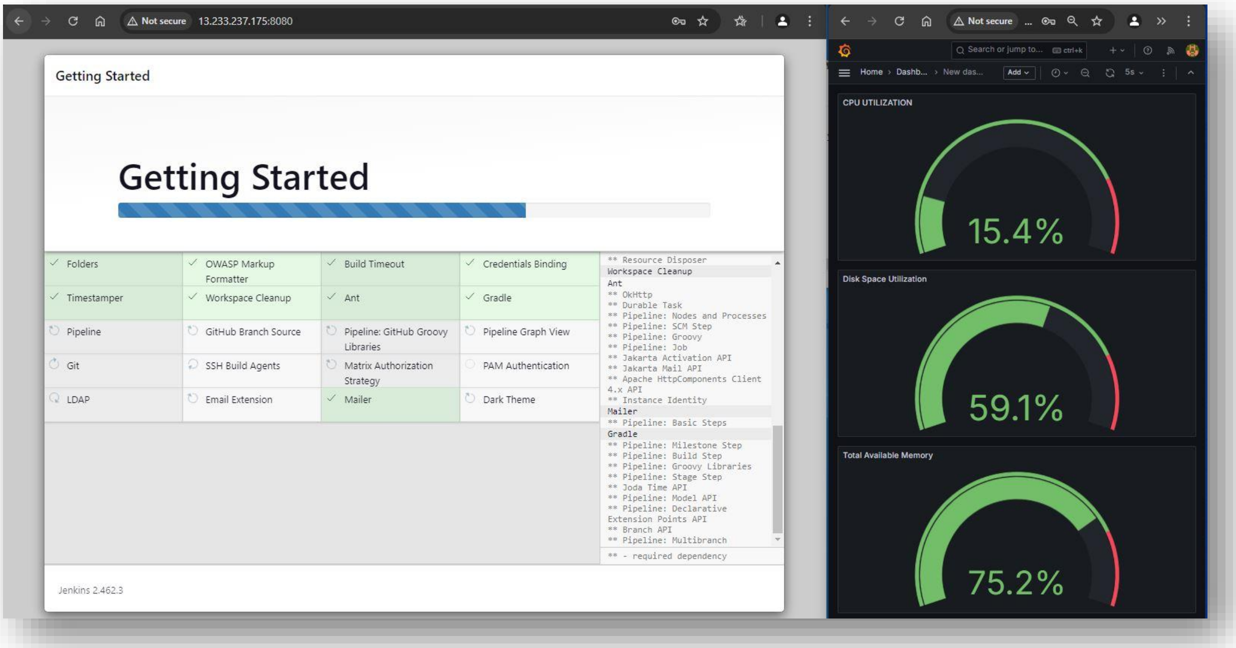


```
abhishek@ip-10-0-1-122:~/FinanceDevopsProject/Ansible_Scripts$ ansible-playbook Jenkins.yml1

PLAY [Install Jenkins] ***************************************************************************
**************************************************************

TASK [Gathering Facts] ***************************************************************************
**************************************************************
[WARNING]: Platform linux on host 10.0.1.125 is using the discovered Python interpreter at /usr/bin/python3.12, but future insta
llation 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 informatio
n.
ok: [10.0.1.125]

TASK [Update APT package index] ******************************************************************
**************************************************************
changed: [10.0.1.125]

TASK [Install required packages] ****************************************************************
**************************************************************
changed: [10.0.1.125]

TASK [Download Jenkins repository GPG key] ******************************************************
**************************************************************
changed: [10.0.1.125]

TASK [Add Jenkins repository] *******************************************************************
**************************************************************
changed: [10.0.1.125]

TASK [Update APT package index after adding Jenkins repository] *********************************
**************************************************************
changed: [10.0.1.125]

TASK [Install Jenkins] ***************************************************************************
**************************************************************
```

i-0244dc1aa3e7fb0ce (Ansible-Server)

PublicIPs: 13.235.128.242 PrivateIPs: 10.0.1.122

More Real time fluctuation in CPU Utilization , Disk Space Utilization and Total Available Memory :-



Getting Started

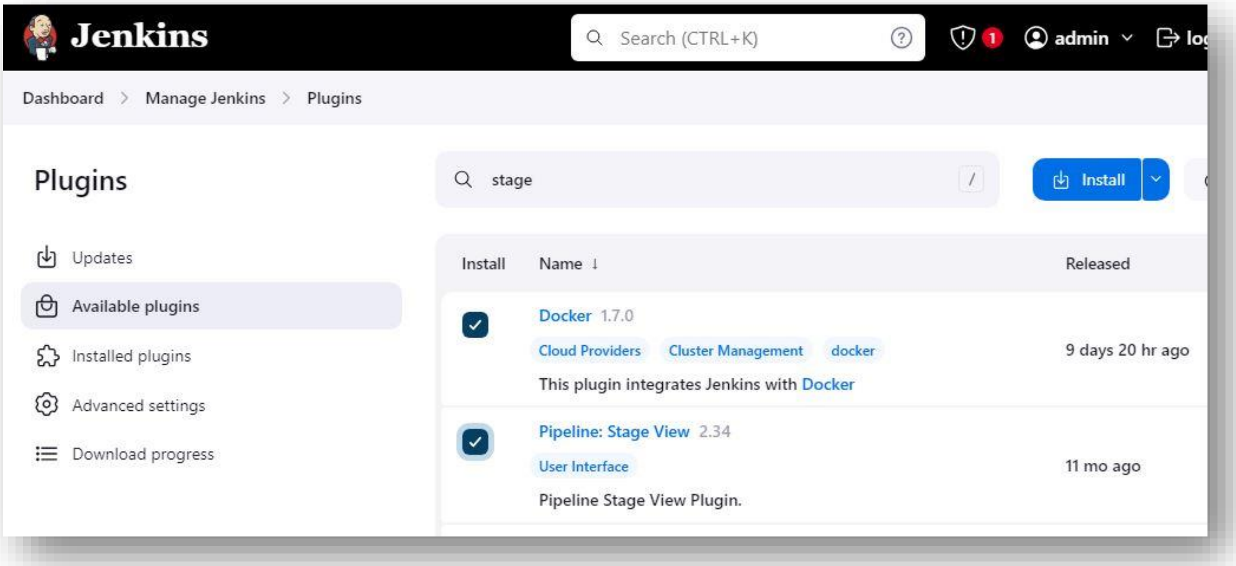
Getting Started

✓ Folders	✓ OWASP Markup Formatter	✓ Build Timeout	✓ Credentials Binding	** Resource Disposer Workspace Cleanup Ant ** OkHttp ** Durable Task ** Pipeline: Nodes and Processes ** Pipeline: SCM Step ** Pipeline: Groovy ** Pipeline: Job ** Jakarta Activation API ** Jakarta Mail API ** Apache HttpComponents Client 4.x API ** Instance Identity Mailer ** Pipeline: Basic Steps Gradle ** Pipeline: Milestone Step ** Pipeline: Build Step ** Pipeline: Groovy Libraries ** Pipeline: Stage Step ** Joda Time API ** Pipeline: Model API ** Pipeline: Declarative Extension Points API ** Branch API ** Pipeline: Multibranch
✓ Timestamper	✓ Workspace Cleanup	✓ Ant	✓ Gradle	
Pipeline	GitHub Branch Source	Pipeline: GitHub Groovy Libraries	Pipeline Graph View	
Git	SSH Build Agents	Matrix Authorization Strategy	PAM Authentication	
LDAP	Email Extension	✓ Mailer	Dark Theme	

** - required dependency

Jenkins 2.462.3

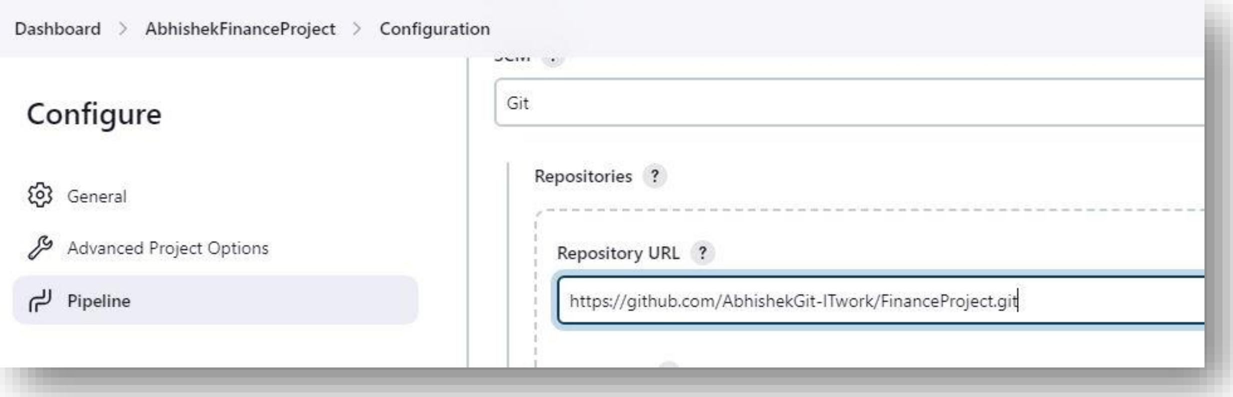
Install additional necessary plugins in Jenkins :-



stage

Install	Name ↓	Released
☑	Docker 1.7.0 Cloud Providers Cluster Management docker This plugin integrates Jenkins with Docker	9 days 20 hr ago
☑	Pipeline: Stage View 2.34 User Interface Pipeline Stage View Plugin.	11 mo ago

While Configuration Jenkins Pipeline add necessary git repository web link – https://github.com/AbhishekGit-ITwork/FinanceProject.git as below :-



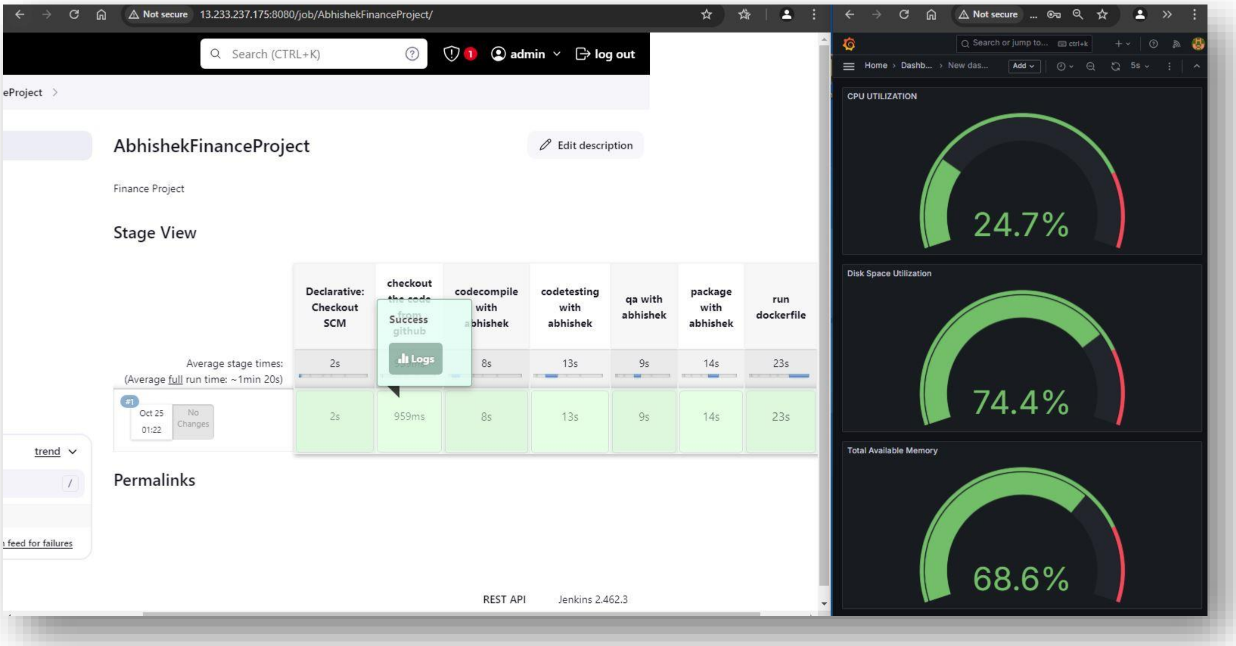
Git

Repositories ?

Repository URL ?

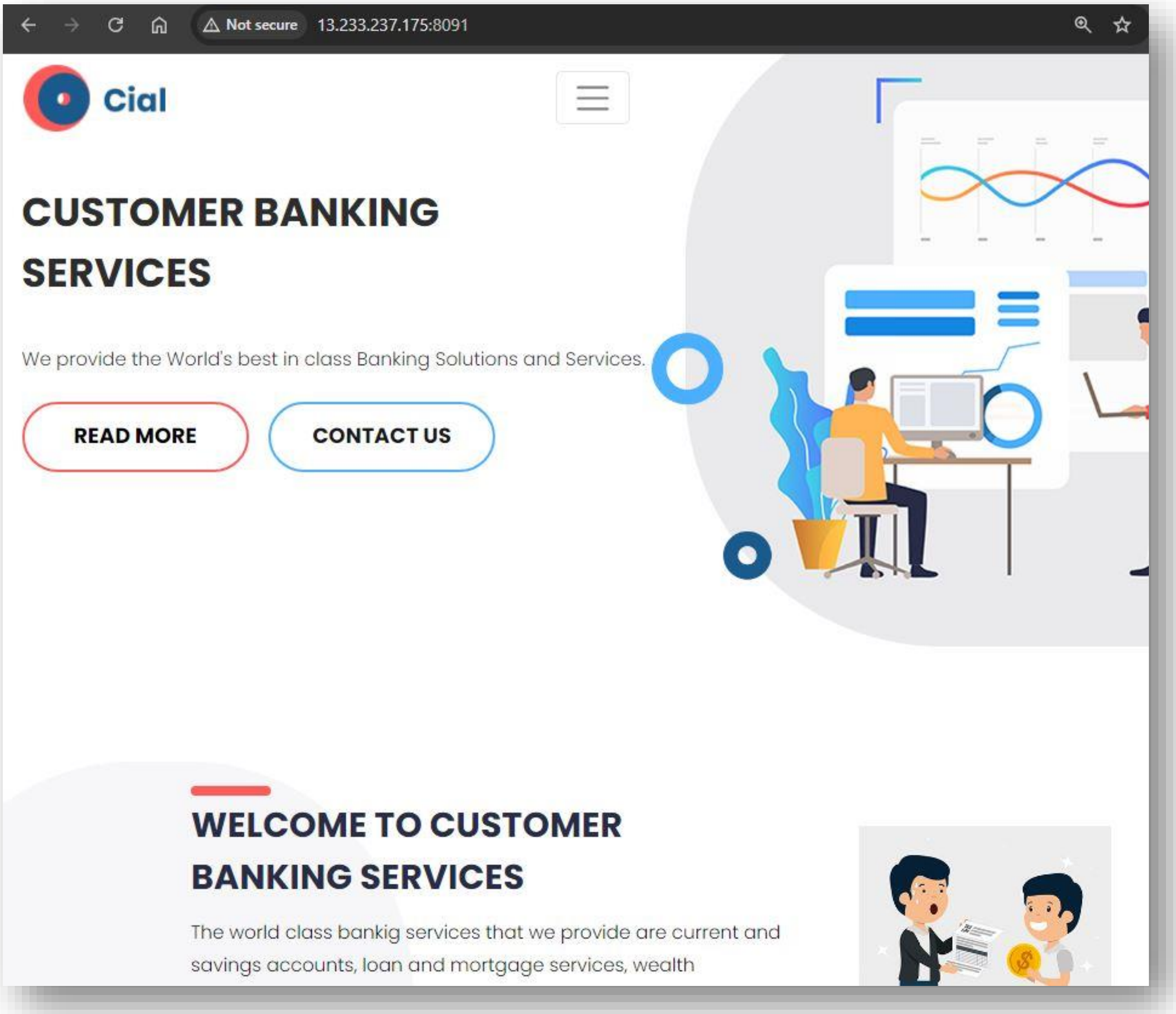
https://github.com/AbhishekGit-ITwork/FinanceProject.git

Now start to build and see the real time increase in CPU Utilization , Disk Space Utilization and Total Available Memory Metrics -



	Declarative: Checkout SCM	checkout the code from github	codecompile with abhishek	codetesting with abhishek	qa with abhishek	package with abhishek	run dockerfile
Average stage times: (Average full run time: ~1min 20s)	2s		8s	13s	9s	14s	23s
#1 Oct 25 01:22 No Changes	2s	959ms	8s	13s	9s	14s	23s

Now run the **Jenkins server IP with 8091 port**, here **Finance Web Application running successfully**, showing that the Jenkins server is successfully deployed for serving the website.



⚠ Not secure 13.233.237.175:8091

Cial

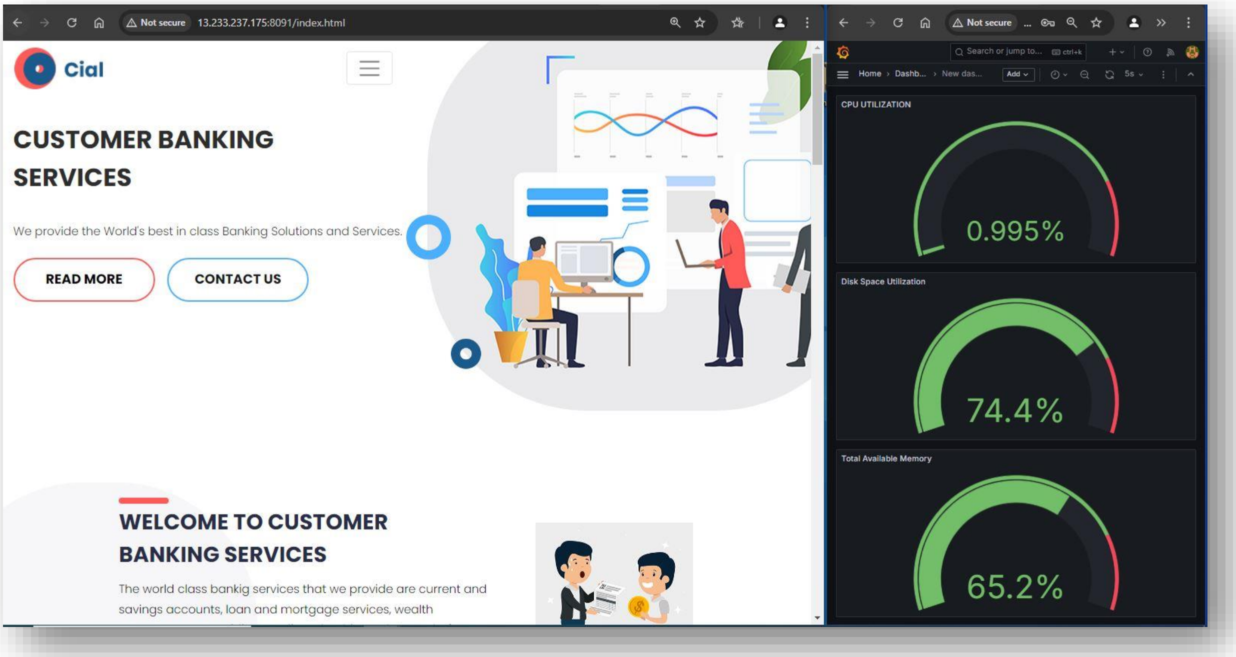
CUSTOMER BANKING SERVICES

We provide the World's best in class Banking Solutions and Services.

WELCOME TO CUSTOMER BANKING SERVICES

The world class bankig services that we provide are current and savings accounts, loan and mortgage services, wealth

Here we can see that after successful deployment and with running of Finance web application website , CPU Utilization reduced below 1%.



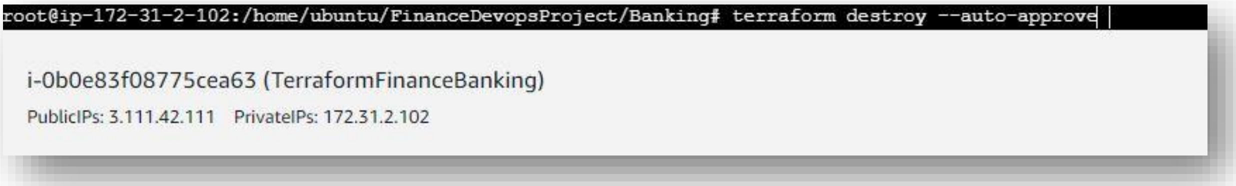
CUSTOMER BANKING SERVICES

We provide the World's best in class Banking Solutions and Services.

WELCOME TO CUSTOMER BANKING SERVICES

The world class bankig services that we provide are current and savings accounts, loan and mortgage services, wealth

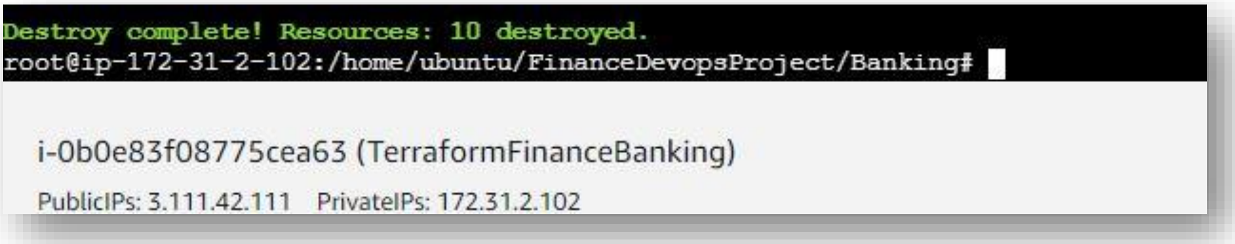
Now , in terraform machine , run the **terraform destroy –auto-approve** command to remove all the infrastructure that were created at beginning using Terraform configuration files.



```
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking# terraform destroy --auto-approve
```

i-0b0e83f08775cea63 (TerraformFinanceBanking)

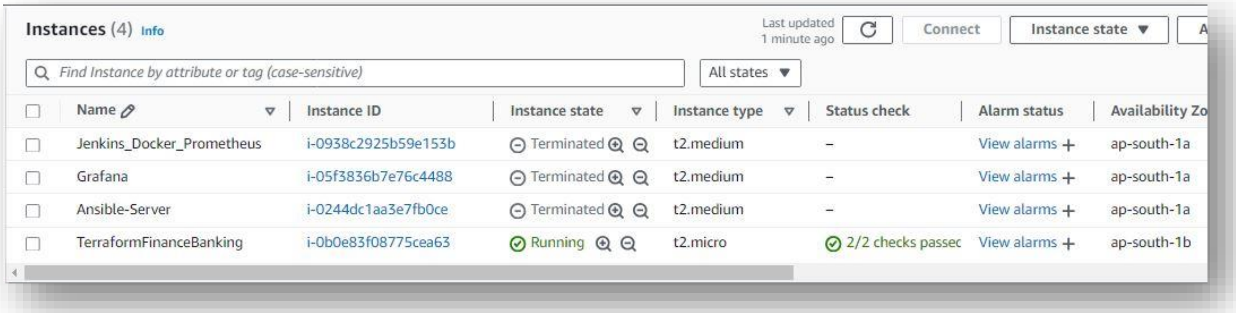
PublicIPs: 3.111.42.111 PrivateIPs: 172.31.2.102



```
Destroy complete! Resources: 10 destroyed.
root@ip-172-31-2-102:/home/ubuntu/FinanceDevopsProject/Banking#
```

i-0b0e83f08775cea63 (TerraformFinanceBanking)

PublicIPs: 3.111.42.111 PrivateIPs: 172.31.2.102

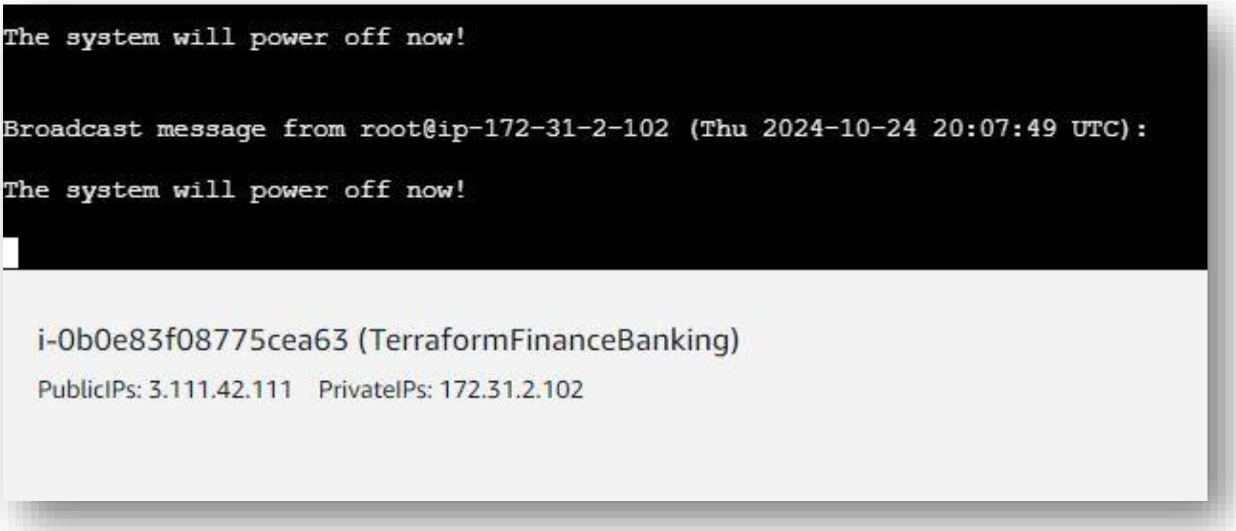


Instances (4) Info

Last updated 1 minute ago

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zo
□	Jenkins_Docker_Prometheus	i-0938c2925b59e153b	Terminated	t2.medium	–	View alarms +	ap-south-1a
□	Grafana	i-05f3836b7e76c4488	Terminated	t2.medium	–	View alarms +	ap-south-1a
□	Ansible-Server	i-0244dc1aa3e7fb0ce	Terminated	t2.medium	–	View alarms +	ap-south-1a
□	TerraformFinanceBanking	i-0b0e83f08775cea63	Running	t2.micro	2/2 checks passec	View alarms +	ap-south-1b

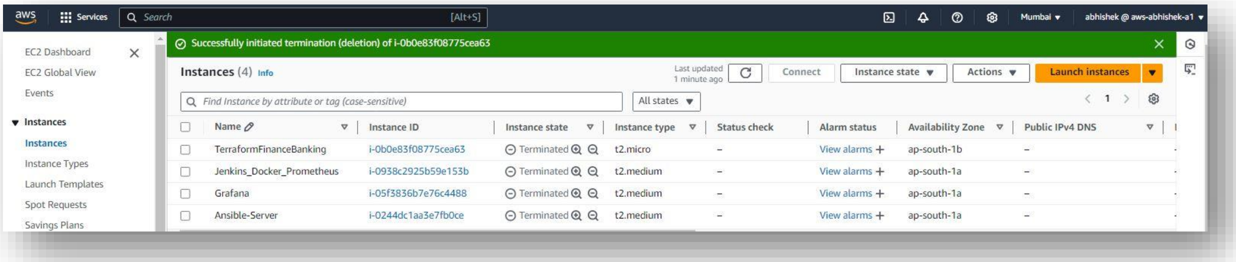
Now terminate the Terraform instance manually



```
The system will power off now!

Broadcast message from root@ip-172-31-2-102 (Thu 2024-10-24 20:07:49 UTC):

The system will power off now!

```

i-0b0e83f08775cea63 (TerraformFinanceBanking)

PublicIPs: 3.111.42.111   PrivateIPs: 172.31.2.102



Successfully initiated termination (deletion) of i-0b0e83f08775cea63

Instances (4) Info

Last updated 1 minute ago

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
□	TerraformFinanceBanking	i-0b0e83f08775cea63	Terminated	t2.micro	–	View alarms +	ap-south-1b	–
□	Jenkins_Docker_Prometheus	i-0938c2925b59e153b	Terminated	t2.medium	–	View alarms +	ap-south-1a	–
□	Grafana	i-05f3836b7e76c4488	Terminated	t2.medium	–	View alarms +	ap-south-1a	–
□	Ansible-Server	i-0244dc1aa3e7fb0ce	Terminated	t2.medium	–	View alarms +	ap-south-1a	–

****