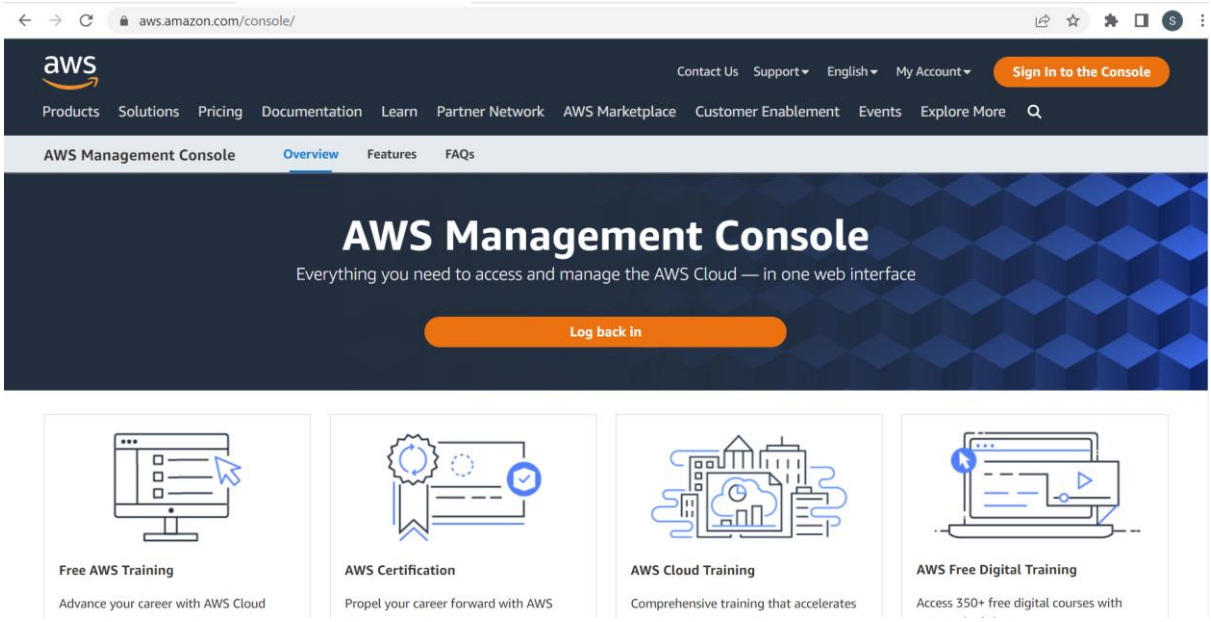
Terraform using Modules, CloudWatch, and Ansible

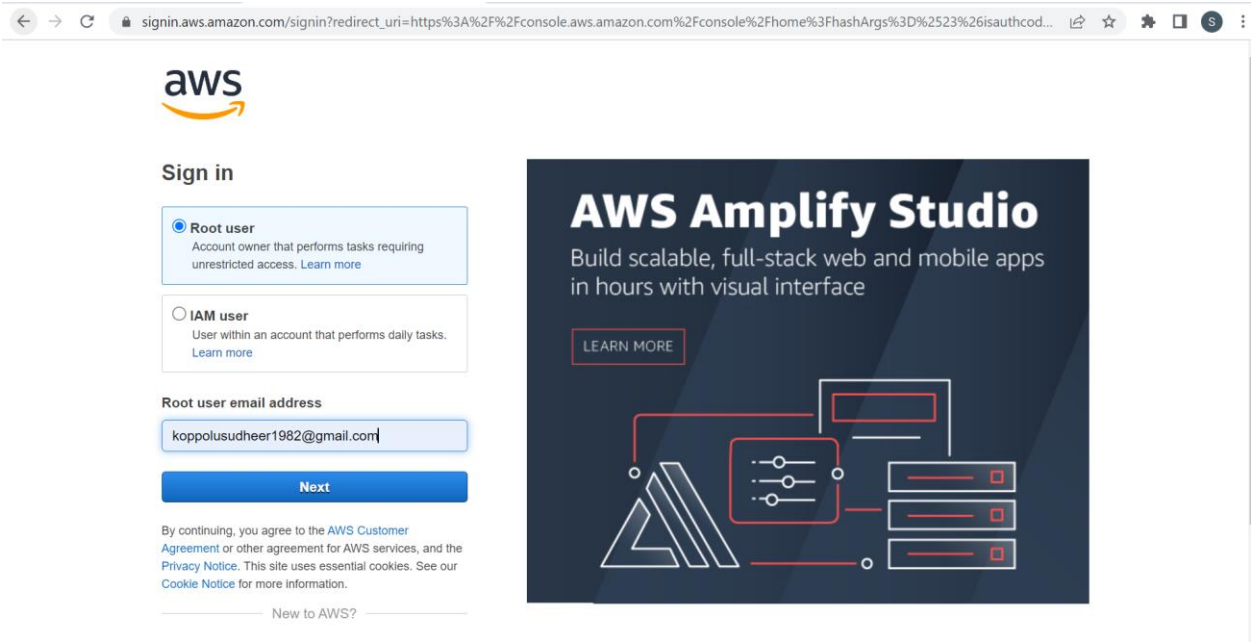
Terraform

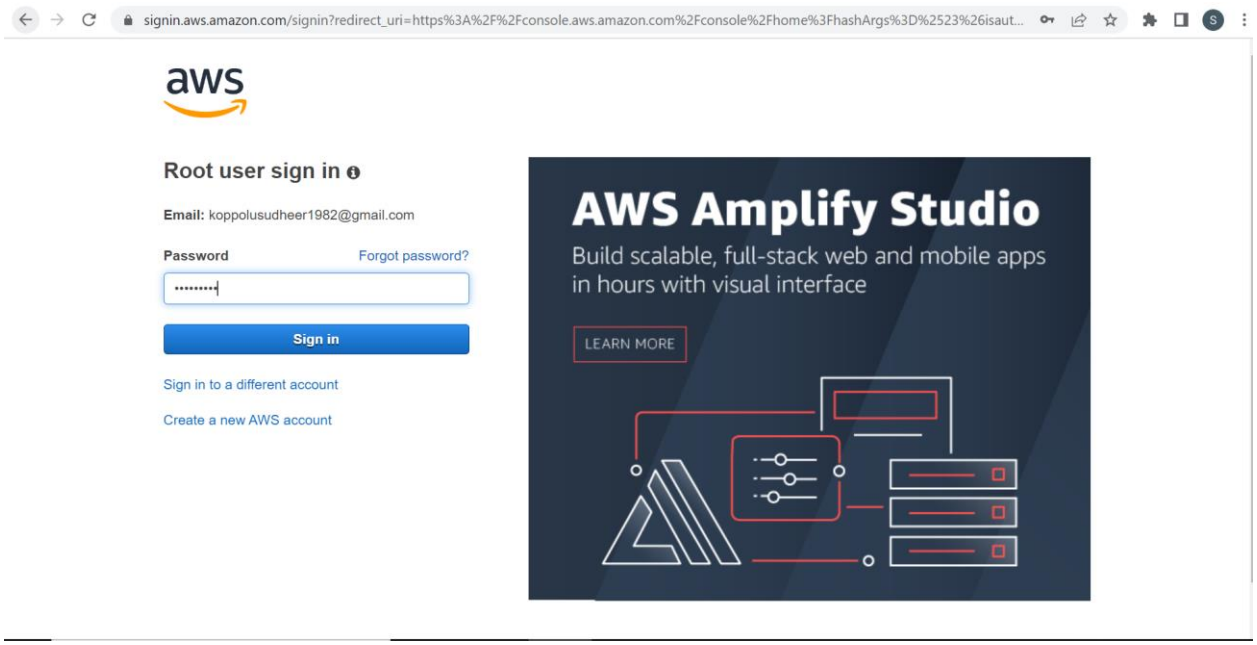
In this project, I have written Terraform code for creation of 4 EC2 machines, CloudWatch monitoring. Also installed Ansible by making of one master and 3 nodes. The terraform code will be available in github (https://github.com/sudheerkumar19/Terraform-Project-with-CloudWatch-Ansible.git).

1. **Open aws.amazon.com**

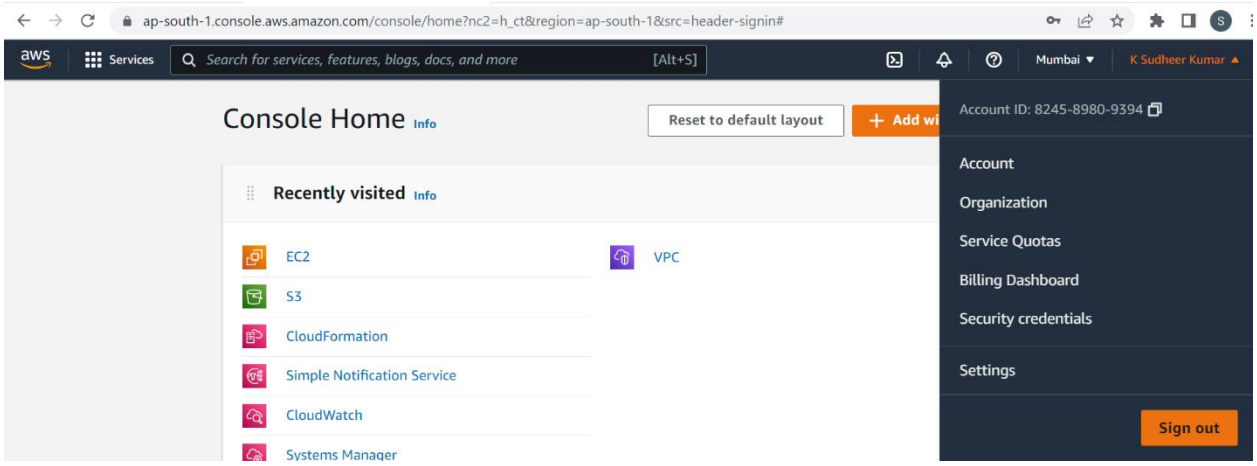


2. **Login with credentials.**

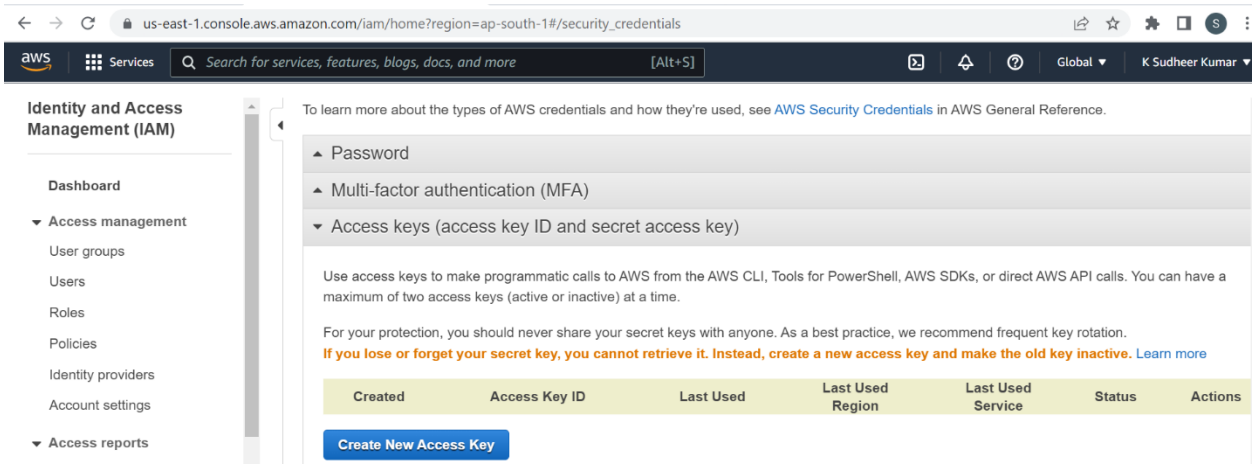




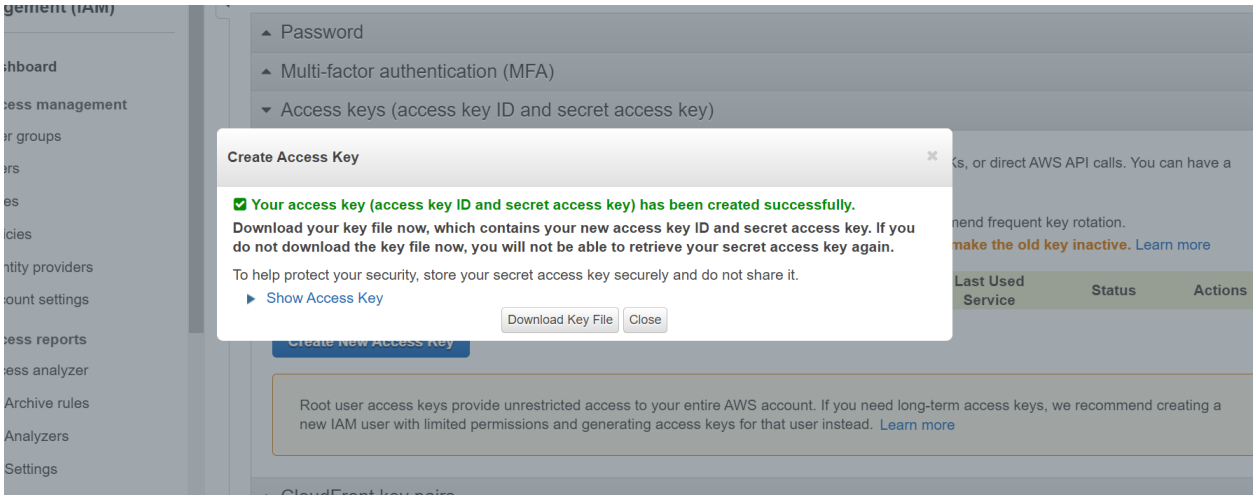
3. **Select security credentials by clicking on account.**



4. **Click on "Create New Access Key"**



5. **Download the file to get access key and secret key.**



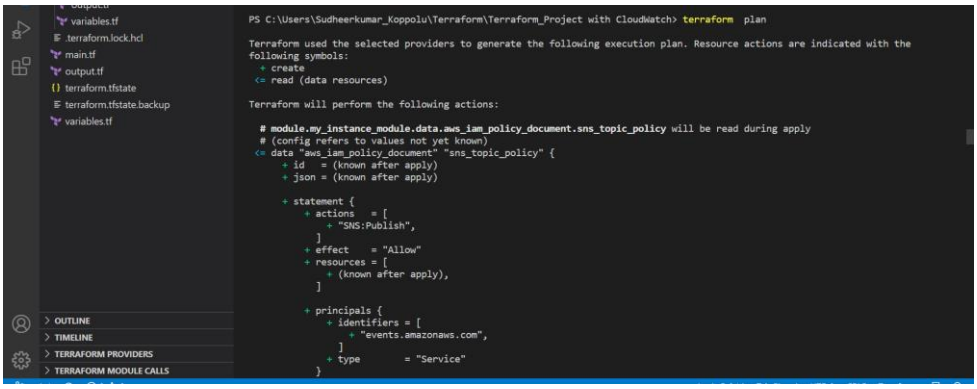
6. **Now, go to CLI and type aws configure and enter key values to access aws.**

```
PS C:\Users\Sudheerkumar_Koppolu\Terraform\terraform_project> aws configure
AWS Access Key ID [****************F6QV]: AKIA377LB63ZC4N7Q5V4
AWS Secret Access Key [****************Ci8s]: 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
Default region name [ap-south-1]:
Default output format [json]:
PS C:\Users\Sudheerkumar_Koppolu\Terraform\terraform_project>
```

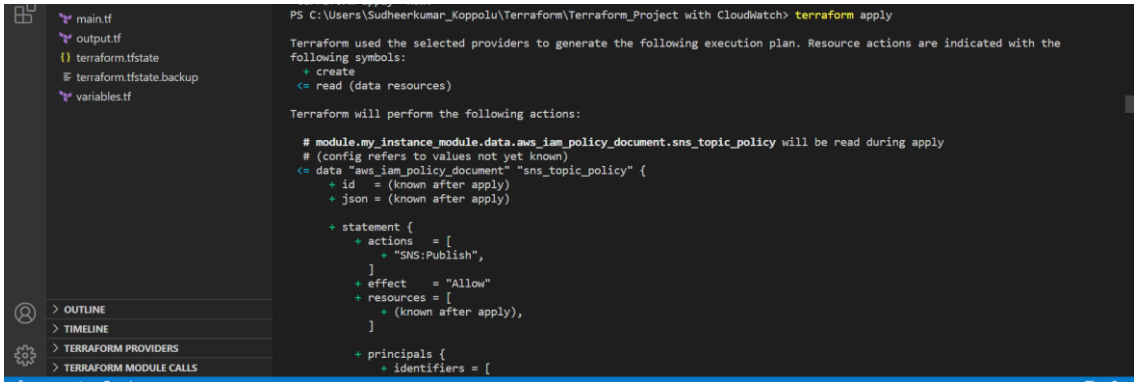
7. **I have uploaded terraform code to the github.**
8. **The first command to give is terraform init.**

```
PS C:\Users\Sudheerkumar_Koppolu\Terraform\terraform_project> terraform init
Initializing modules...

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v4.23.0

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.
PS C:\Users\Sudheerkumar_Koppolu\Terraform\terraform_project>
```

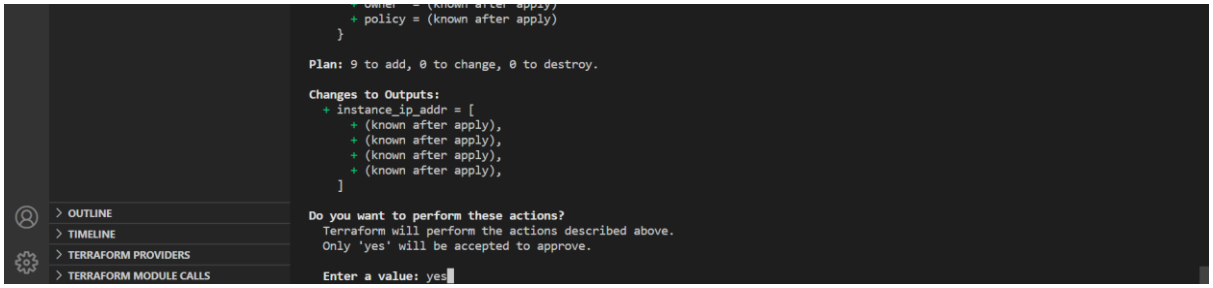
9. **Next command to use is “terraform plan”**



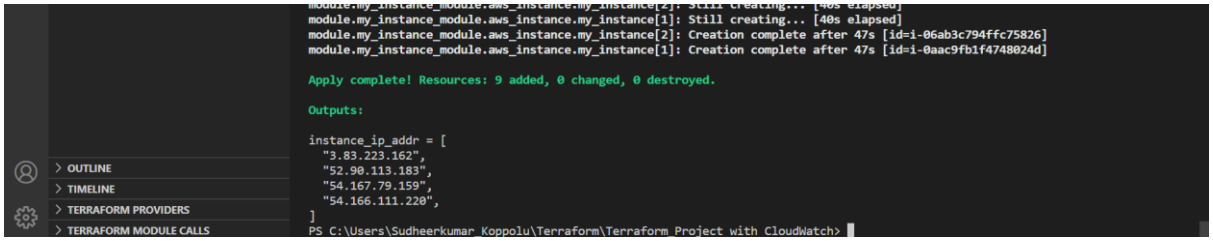
10. **We need to allocate resources from the aws. So, we need to execute “terraform apply”.**



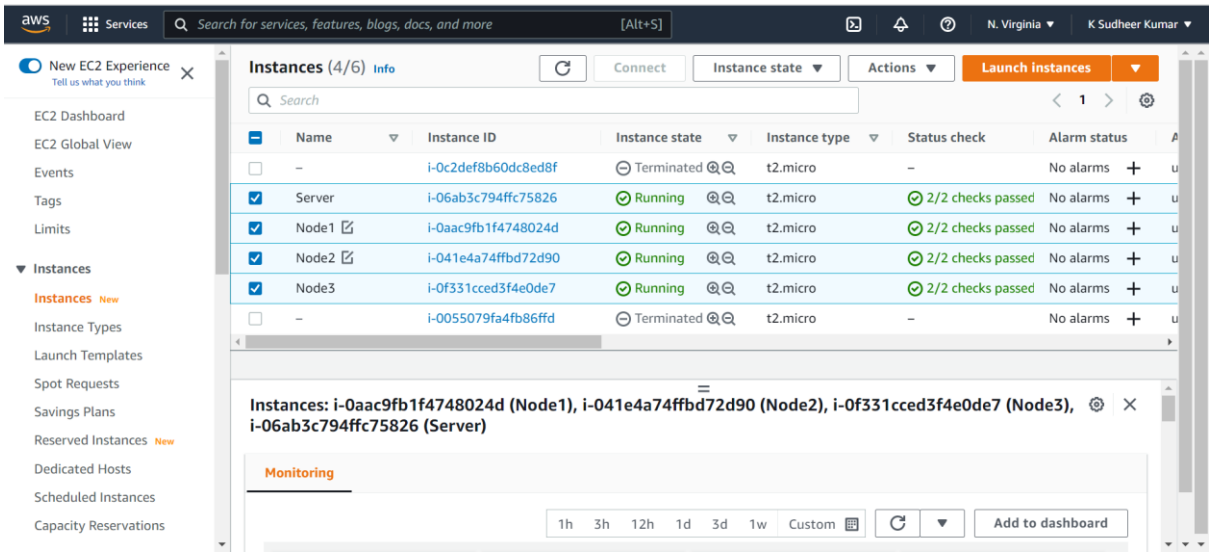
11. **Type “yes” and Enter**



12. **Resources are added.**



13. **We check in the aws console, 4 EC2 machines are created.**



Ansible (Configuration Tool)

Now, we are going to connect to the instances. One instance made as server in which we are installing Ansible. The others considered as nodes for configuration of softwares.

1. **Connect to node1.**

```
ubuntu@ip-172-31-82-115: ~
EPAM+Sudheerkumar_Koppolu@EP-5CG9377900 MINGW64 ~/Desktop/key (master)
$ ssh -i "demo0308.pem" ubuntu@ec2-52-90-113-183.compute-1.amazonaws.com
The authenticity of host 'ec2-52-90-113-183.compute-1.amazonaws.com (52.90.113.183)' can't be established.
ED25519 key fingerprint is SHA256:vsBnbZbRNBPMIbq7fMDsBtymX0lJ448s+mIgmsttREA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-52-90-113-183.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1078-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Thu Aug  4 05:33:00 UTC 2022

  System load:  0.05              Processes:           95
  Usage of /:   16.4% of 7.58GB   Users logged in:     0
  Memory usage: 19%               IP address for eth0: 172.31.82.115
  Swap usage:   0%

0 updates can be applied immediately.


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-82-115:~$
```

2. **Establish password less ssh connection. change PasswordAuthentication→ yes, Save and QUIT**
3. **Repeat the same steps in node2 and node3**

```
ubuntu@ip-172-31-82-115:~$ sudo passwd ubuntu
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
ubuntu@ip-172-31-82-115:~$ sudo vim /etc/ssh/sshd_config
ubuntu@ip-172-31-82-115:~$ sudo service ssh restart
ubuntu@ip-172-31-82-115:~$ exit
logout
Connection to ec2-52-90-113-183.compute-1.amazonaws.com closed.
```

4. **Now, Connect to controller (server).**

```
ubuntu@ip-172-31-91-175:~$ ssh-copy-id ubuntu@172.31.82.115
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/ubuntu/.ssh/id_rsa.pub"
The authenticity of host '172.31.82.115 (172.31.82.115)' can't be established.
ECDSA key fingerprint is SHA256:dT/Vu9vO0Y88UCl197VP02FP4XUTqAcnAswSUvLb0TY.
Are you sure you want to continue connecting (yes/no)? 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
ubuntu@172.31.82.115's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'ubuntu@172.31.82.115'"
and check to make sure that only the key(s) you wanted were added.
```

5. **We need to generate ssh connections.**

```
ubuntu@ip-172-31-91-175:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa):
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:cNANsZRCeDJWmqJMf3k/cf09swNZszyyO4GVltuWrCU ubuntu@ip-172-31-91-175
The key's randomart image is:
+---[RSA 2048]----+
|     ++.+=       |
|    =oooo..      |
| ...o+..o       o|
|o...  .o    .  B |
|.. . o .S. . .O *|
|    . . . o  E.X+|
|         o    O=+|
|          .  + o+|
|              ...|
+----[SHA256]-----+
```

6. **Now copy the key to managed nodes.**

```
ubuntu@ip-172-31-91-175:~$ sudo apt-get install software-properties-common
Reading package lists... Done
Building dependency tree
Reading state information... Done
software-properties-common is already the newest version (0.96.24.32.18).
software-properties-common set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-91-175:~$ sudo apt-add-repository ppa:ansible/ansible
 Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid
 writing scripts or custom code to deploy and update your applications— automate in a language that approaches plain E
nglish, using SSH, with no agents to install on remote systems.

http://ansible.com/
```

```
ubuntu@ip-172-31-91-175:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:5 http://ppa.launchpad.net/ansible/ansible/ubuntu bionic InRelease
Reading package lists... Done
```

7. **Installing ansible now in controller (server)**

```
ubuntu@ip-172-31-91-175:~$ sudo apt-get install -y ansible
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libpython-stdlib libpython2.7-minimal libpython2.7-stdlib python python-asn1crypto python-cffi-backend
  python-crypto python-cryptography python-enum34 python-httplib2 python-idna python-ipaddress python-jinja2
  python-markupsafe python-minimal python-paramiko python-pkg-resources python-pyasn1 python-setuptools python-six
  python-yaml python2.7 python2.7-minimal sshpass
Suggested packages:
  python-doc python-tk python-crypto-doc python-cryptography-doc python-cryptography-vectors python-enum34-doc
  python-jinja2-doc python-gssapi python-setuptools-doc python2.7-doc binutils binfmt-support
The following NEW packages will be installed:
  ansible libpython-stdlib libpython2.7-minimal libpython2.7-stdlib python python-asn1crypto python-cffi-backend
  python-crypto python-cryptography python-enum34 python-httplib2 python-idna python-ipaddress python-jinja2
  python-markupsafe python-minimal python-paramiko python-pkg-resources python-pyasn1 python-setuptools python-six
  python-yaml python2.7 python2.7-minimal sshpass
0 upgraded, 25 newly installed, 0 to remove and 35 not upgraded.
Need to get 11.4 MB of archives.
```

8. **To check whether Ansible installed or not.**

```
ubuntu@ip-172-31-91-175:~$ ansible --version
ansible 2.9.27
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/ubuntu/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.17 (default, Jul  1 2022, 15:56:32) [GCC 7.5.0]
```

9. **Write the ip address of nodes in the inventory file. cd /etc/ansible, ls, sudo vim hosts**

```
ubuntu@ip-172-31-24-99: /etc/ansible
172.31.27.2
# This is the default ansible 'hosts' file.
#
# It should live in /etc/ansible/hosts
#
#   - Comments begin with the '#' character
#   - Blank lines are ignored
#   - Groups of hosts are delimited by [header] elements
#   - You can enter hostnames or ip addresses
#   - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers.

## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

-- INSERT --                                                    1,12          Top
```

10. **Creating a role “my-role”.**

```
ubuntu@ip-172-31-91-175:/etc/ansible$ ls
ansible.cfg  files  master.yml  my-role   prerequisites
apache       hosts  mongodb     playbook  roles
ubuntu@ip-172-31-91-175:/etc/ansible$ cd role
-bash: cd: role: No such file or directory
ubuntu@ip-172-31-91-175:/etc/ansible$ cd prerequisites/tasks/
ubuntu@ip-172-31-91-175:/etc/ansible/prerequisites/tasks$ sudo vim main.yml
ubuntu@ip-172-31-91-175:/etc/ansible/prerequisites/tasks$ ls
main.yml
```

11. **Writing code in main.yml in prerequisites, apache, and mongodb to install git, apache, and mongodb into node1, node2, node3.**

```
ubuntu@ip-172-31-91-175:/etc/ansible$ cd role
-bash: cd: role: No such file or directory
ubuntu@ip-172-31-91-175:/etc/ansible$ cd prerequisites/tasks/
ubuntu@ip-172-31-91-175:/etc/ansible/prerequisites/tasks$ sudo vim main.yml
ubuntu@ip-172-31-91-175:/etc/ansible/prerequisites/tasks$ ls
main.yml
ubuntu@ip-172-31-91-175:/etc/ansible/prerequisites/tasks$ cd ..
ubuntu@ip-172-31-91-175:/etc/ansible/prerequisites$ ls
README.md  defaults  files  handlers  meta  tasks  templates  tests  vars
ubuntu@ip-172-31-91-175:/etc/ansible/prerequisites$ cd ..
ubuntu@ip-172-31-91-175:/etc/ansible$ ls
ansible.cfg  files  master.yml  my-role   prerequisites
apache       hosts  mongodb     playbook  roles
ubuntu@ip-172-31-91-175:/etc/ansible$ cd mongodb
ubuntu@ip-172-31-91-175:/etc/ansible/mongodb$ ls
README.md  defaults  files  handlers  meta  tasks  templates  tests  vars
ubuntu@ip-172-31-91-175:/etc/ansible/mongodb$ cd tasks
```

Prerequisites-→Tasks-→ main.yml

```
ubuntu@ip-172-31-91-175: /etc/ansible/prerequisites/tasks
ubuntu@ip-172-31-91-175:/etc/ansible$ cd prerequisites
ubuntu@ip-172-31-91-175:/etc/ansible/prerequisites$ cd tasks
ubuntu@ip-172-31-91-175:/etc/ansible/prerequisites/tasks$ cat main.yml
---
# tasks file for prerequisites
- name: Install git
  apt:
     name: git
     state: present
     update_cache: yes
ubuntu@ip-172-31-91-175:/etc/ansible/prerequisites/tasks$ ^C
ubuntu@ip-172-31-91-175:/etc/ansible/prerequisites/tasks$
```

Apache -→ tasks -→ main.yml

```
ubuntu@ip-172-31-91-175:/etc/ansible$ cd apache
ubuntu@ip-172-31-91-175:/etc/ansible/apache$ cd tasks
ubuntu@ip-172-31-91-175:/etc/ansible/apache/tasks$ cat main.yml
---
# tasks file for apache
- name: install Apache web server
  apt:
     name=apache2
     state=present
     update_cache=yes
ubuntu@ip-172-31-91-175:/etc/ansible/apache/tasks$ ^C
ubuntu@ip-172-31-91-175:/etc/ansible/apache/tasks$
```

Mongodab -→ tasks -→ main.yml

```
ubuntu@ip-172-31-91-175:/etc/ansible/mongodb/tasks$ cat main.yml
---
# tasks file for mongodb
- name: Install MongoDB
  apt:
     name: mongodb
     state: present
     update_cache: yes

- name: Start Mongod daemon
  shell: "mongod &"
ubuntu@ip-172-31-91-175:/etc/ansible/mongodb/tasks$
```

Etc/ansible -→ stack.yml

```
ubuntu@ip-172-31-91-175:/etc/ansible$ cd mongodb
ubuntu@ip-172-31-91-175:/etc/ansible/mongodb$ cd tasks
ubuntu@ip-172-31-91-175:/etc/ansible/mongodb/tasks$ ls
main.yml
ubuntu@ip-172-31-91-175:/etc/ansible/mongodb/tasks$ cd ..
ubuntu@ip-172-31-91-175:/etc/ansible/mongodb$ cd ..
ubuntu@ip-172-31-91-175:/etc/ansible$ ls
ansible.cfg  files  master.yml  my-role   prerequisites  stack.yml
apache       hosts  mongodb     playbook  roles
ubuntu@ip-172-31-91-175:/etc/ansible$ cat stack.yml
---
- hosts: localhost
  become: yes
  roles:
        -  prerequisites
        -  mongodb
        -  apache
ubuntu@ip-172-31-91-175:/etc/ansible$
```

12. **Now, executing stack.yml playbook which has prerequisites, mongodb, and apache roles.**

```
ubuntu@ip-172-31-91-175:/etc/ansible$ sudo vi stack.yml
ubuntu@ip-172-31-91-175:/etc/ansible$ sudo ansible-playbook /etc/ansible/st
ack.yml

PLAY [all] ********************************************************************
*****

TASK [Gathering Facts] ********************************************************
*****
The authenticity of host '172.31.82.115 (172.31.82.115)' can't be establish
ed.
ECDSA key fingerprint is SHA256:dT/Vu9vO0Y88UCl197VP02FP4XUTqAcnAswSUvLb0TY
.
```

```
TASK [Gathering Facts] ********************************************************
*****
ok: [localhost]

TASK [prerequisites : Install git] ********************************************
*****

[WARNING]: Updating cache and auto-installing missing dependency: python-ap
t
ok: [localhost]

TASK [mongodb : Install MongoDB] ******************************************
*****
changed: [localhost]

TASK [mongodb : Start Mongod daemon] ******************************************
*****
```

```
TASK [apache : install Apache web server] *************************************
*****
changed: [localhost]

PLAY RECAP ********************************************************************
*****
localhost                  : ok=5    changed=3    unreachable=0    failed=0
    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-91-175:/etc/ansible$
```

13. **To check whether apche2 is installed in the node.**

```
ubuntu@ip-172-31-91-175:/etc/ansible$ whereis apache2
apache2: /usr/sbin/apache2 /usr/lib/apache2 /etc/apache2 /usr/share/apache2
 /usr/share/man/man8/apache2.8.gz
ubuntu@ip-172-31-91-175:/etc/ansible$ /usr/sbin/apache2 -v
Server version: Apache/2.4.29 (Ubuntu)
Server built:   2022-06-23T12:51:37
```

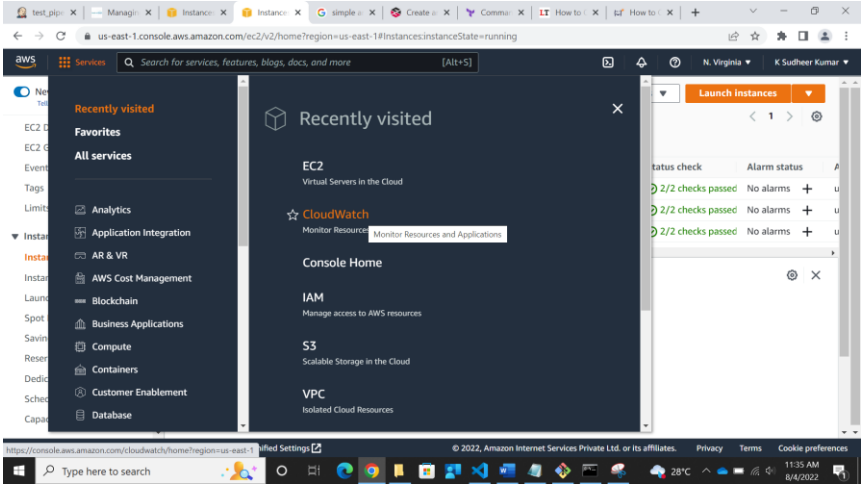
14. **To check whether mongodb installed in the node.**

```
ubuntu@ip-172-31-91-175:/etc/ansible$ mongod --version
db version v3.6.3
git version: 9586e557d54ef70f9ca4b43c26892cd55257e1a5
OpenSSL version: OpenSSL 1.1.1  11 Sep 2018
allocator: tcmalloc
modules: none
build environment:
    distarch: x86_64
    target_arch: x86_64
```

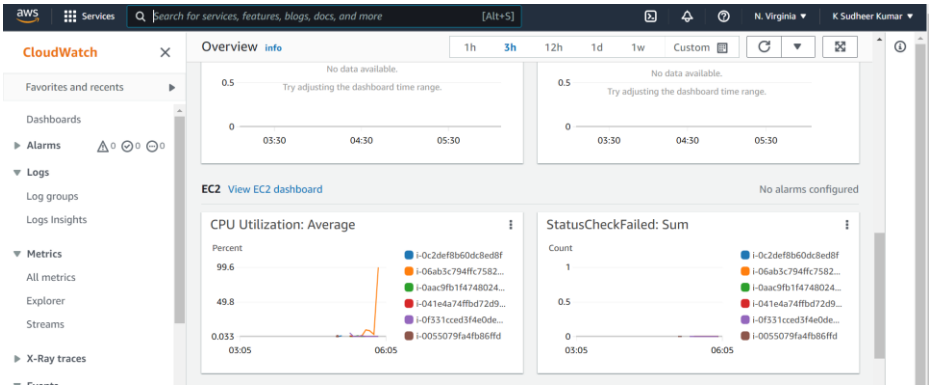
CloudWatch (Monitoring Tool)

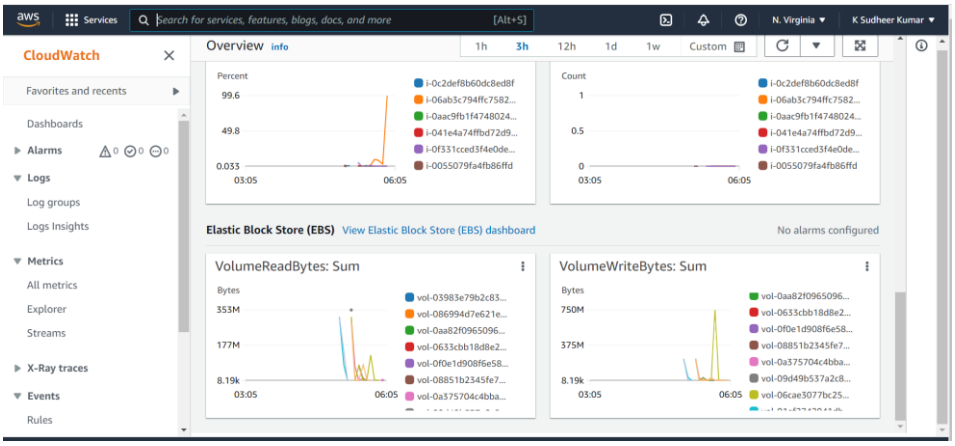
I connected those 4 ec2 instances to CloudWatch (code is in the terraform file) to monitor CPU Utilization, Volume check.

1. **Go to aws console, select CloudWatch service.**

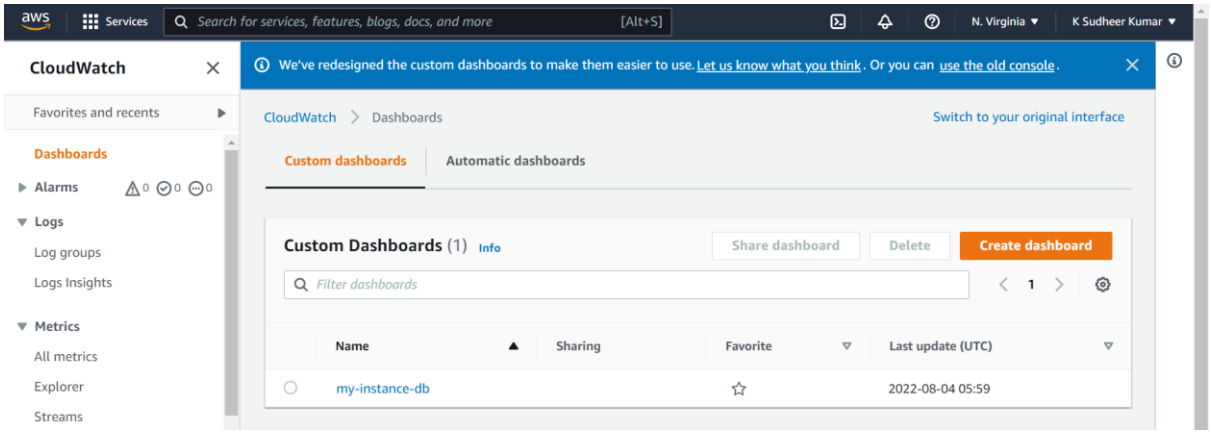


2. **CloudWatch Dashboard (Name: View EC2 Dashboard)**

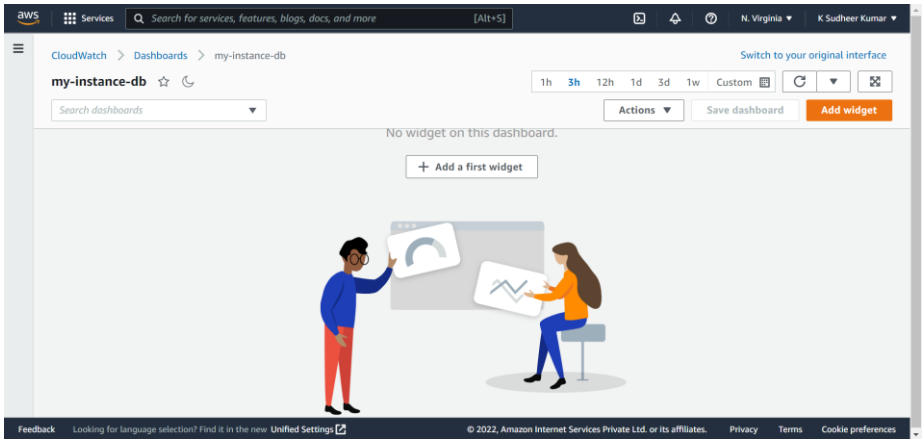




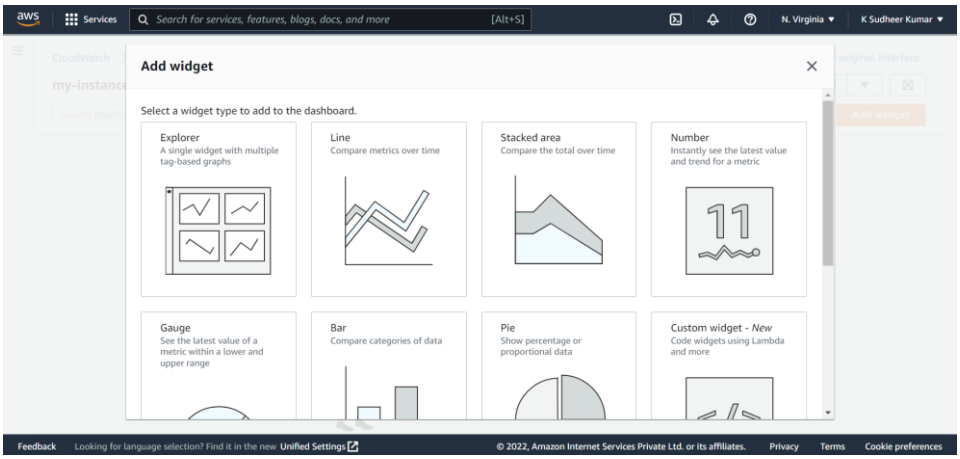
Here created custom Dashboard: my-instance-db

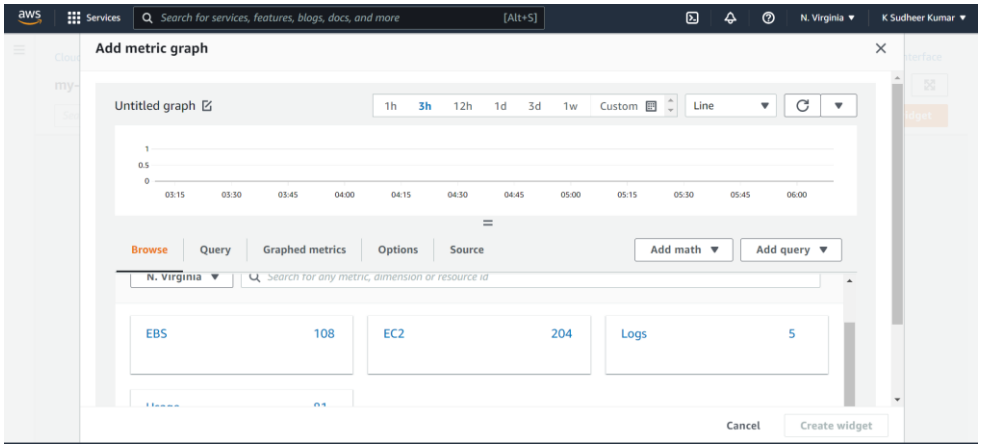


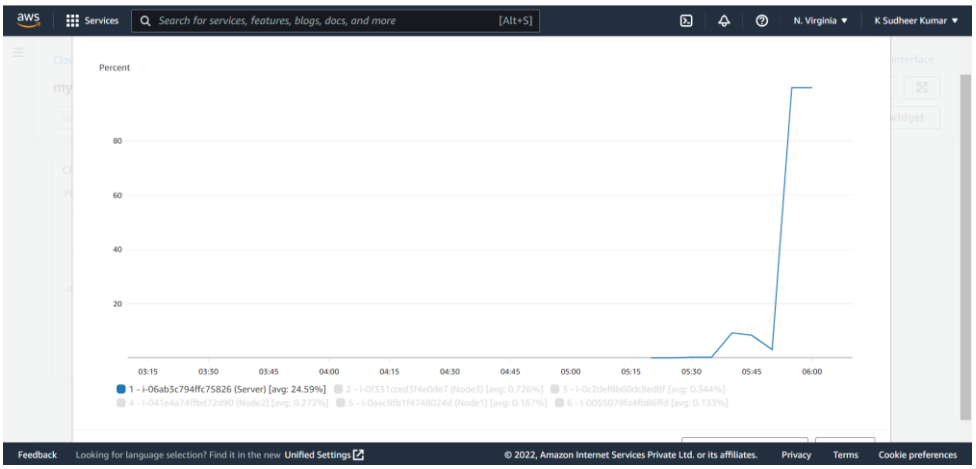
Click on Add widget



Select any chart to represent.




aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
K Sudheer Kumar
Add metric graph
Untitled graph
1h
3h
12h
1d
3d
1w
Custom
Line
1
0.5
0
03:15
03:30
03:45
04:00
04:15
04:30
04:45
05:00
05:15
05:30
05:45
06:00
Browse
Query
Graphed metrics
Options
Source
Add math
Add query
N. Virginia
Search for any metric, dimension or resource id
EBS
108
EC2
204
Logs
5
Cancel
Create widget


aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
K Sudheer Kumar
Percent
80
60
40
20
03:15
03:30
03:45
04:00
04:15
04:30
04:45
05:00
05:15
05:30
05:45
06:00
1 - i-06ab3c794ffc75826 (Server) [avg: 24.59%]
2 - i-0f531cced3f4e0de7 (Node3) [avg: 0.726%]
3 - i-0c2def8b60dc8ed8f [avg: 0.344%]
4 - i-041e4a74ffbd72d90 (Node2) [avg: 0.272%]
5 - i-0aac9fb1f4748024d (Node1) [avg: 0.167%]
6 - i-0055079fa4fb86ffd [avg: 0.133%]
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences