deerwalk
DWIT College

POWERWORSHOP
(DAY 1: HANDS-ON LAB)

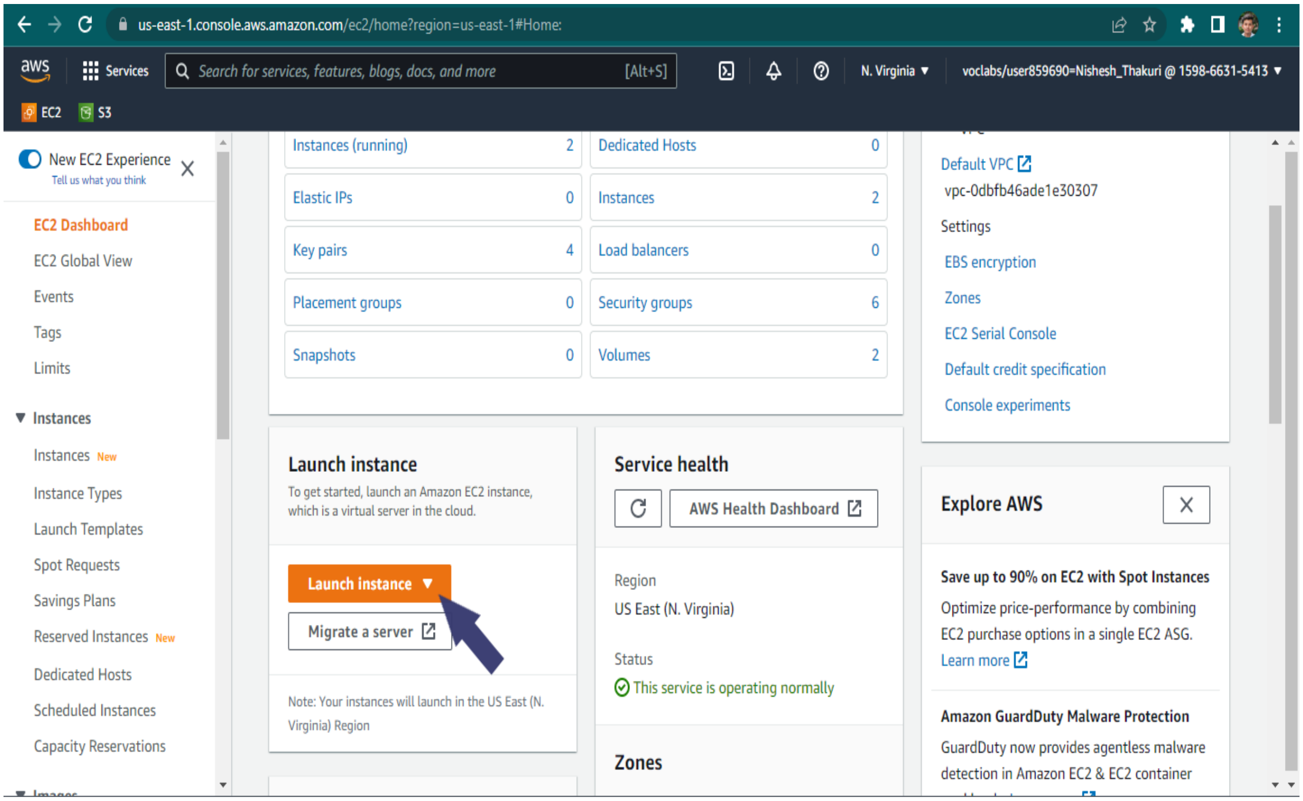
Part-I

Objectives

- Create an instance in EC2 service.
- Creating pair key.
- Connecting to created ec2 instance using SSH.

Instance creation using EC2.

1. Search for EC2 service.
2. Go to the EC2 dashboard and select launch instance option in the dashboard.



Instances (running)	2	Dedicated Hosts	0
Elastic IPs	0	Instances	2
Key pairs	4	Load balancers	0
Placement groups	0	Security groups	6
Snapshots	0	Volumes	2

Launch instance

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

Note: Your instances will launch in the US East (N. Virginia) Region

Service health

Region
US East (N. Virginia)

Status
This service is operating normally

Zones

Default VPC
vpc-0dbfb46ade1e30307

Settings

Explore AWS

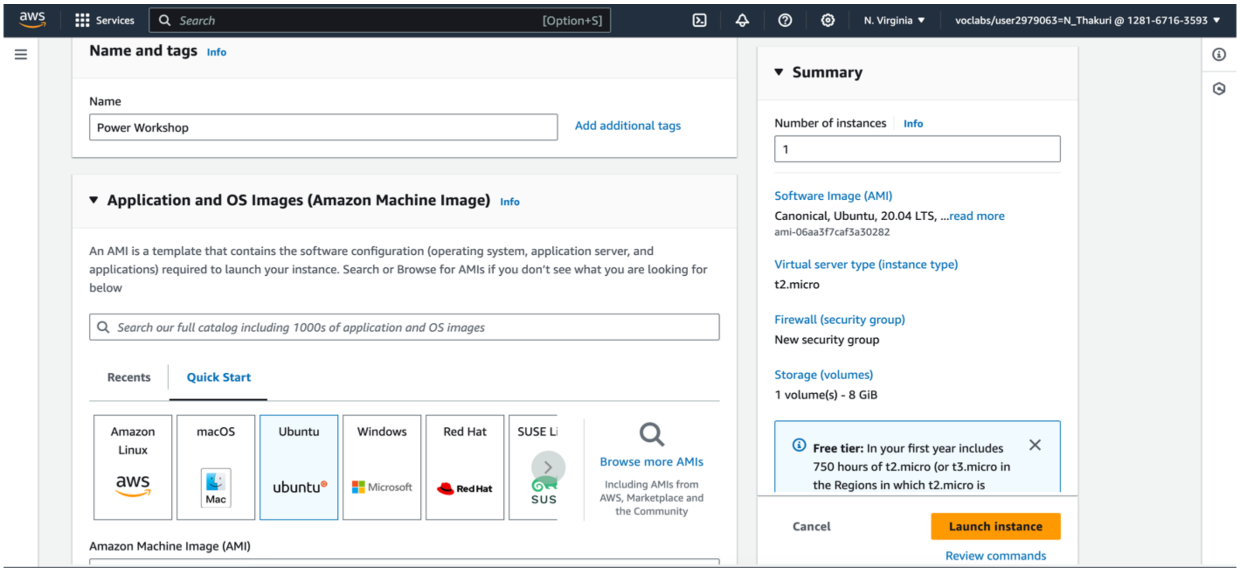
Save up to 90% on EC2 with Spot Instances

Optimize price-performance by combining EC2 purchase options in a single EC2 ASG.

Amazon GuardDuty Malware Protection

GuardDuty now provides agentless malware detection in Amazon EC2 & EC2 container

3. Choose an AMI. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.



Name and tags Info

Name: Power Workshop

Add additional tags

▼ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Amazon Machine Image (AMI)

▼ Summary

Number of instances Info: 1

Software Image (AMI): Canonical, Ubuntu, 20.04 LTS, ...read more
ami-06aa3f7caf3a30282

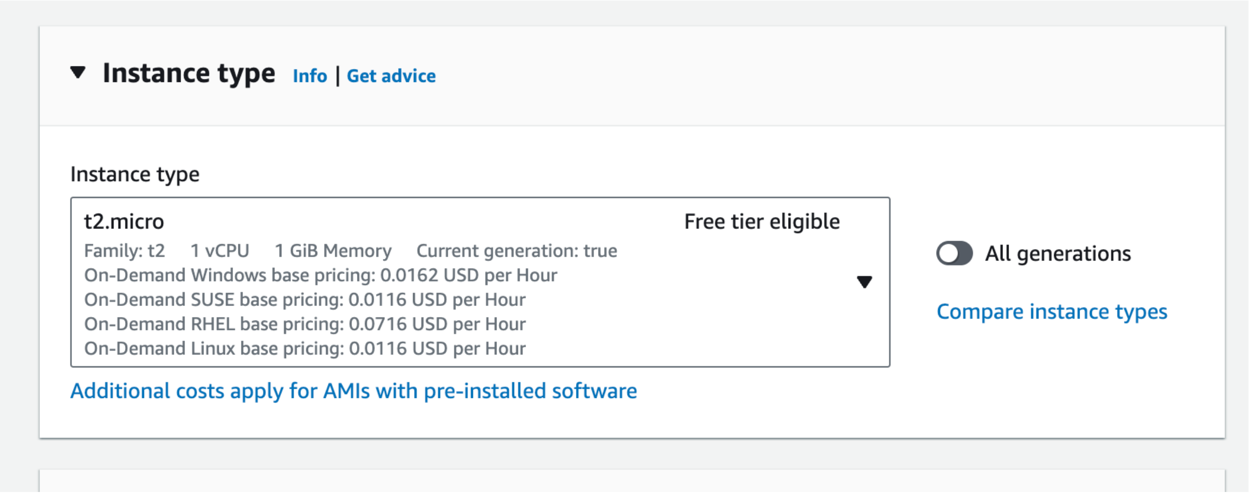
Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is

4. After selecting the AMI, the next step is to choose an instance type. They are grouped by characteristics in terms of compute, memory, storage, and networking resources. To select an EC2 instance, you must understand the application infrastructure requirements and the right EC2 instance type to meet them. For now, we are using t2.micro instance type.



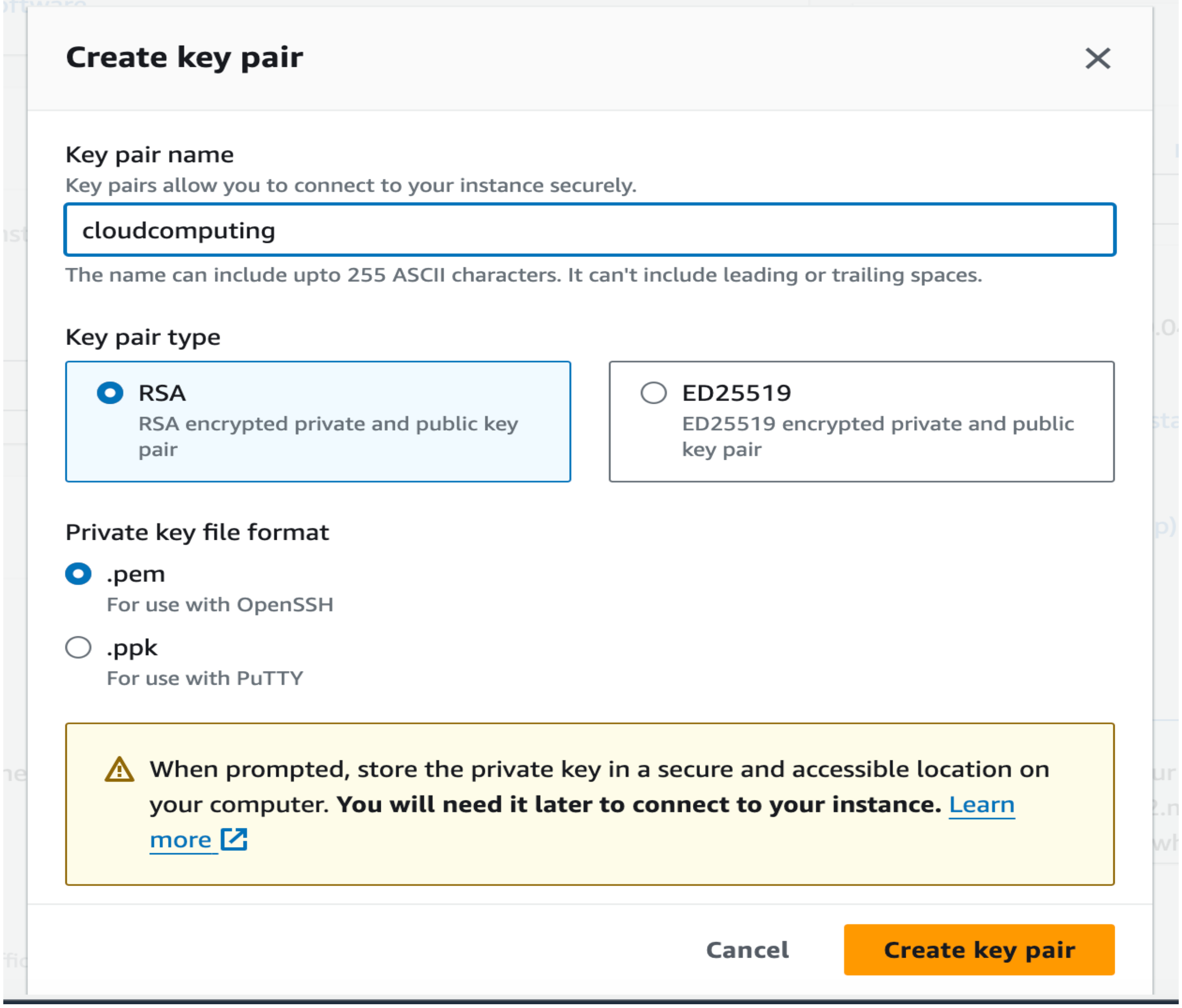
▼ Instance type Info | Get advice

Instance type

t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.0716 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

Additional costs apply for AMIs with pre-installed software

5. After selecting the instance type, the next step is to create an EC2 key or select an existing one. The key is used to enable Secure Shell (SSH) access into the EC2 instance. AWS stores a copy of the public key inside the EC2 instance. Users keep the private key.



Create key pair

Key pair name

Key pairs allow you to connect to your instance securely.

cloudcomputing

The name can include upto 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

● RSA
RSA encrypted private and public key pair

○ ED25519
ED25519 encrypted private and public key pair

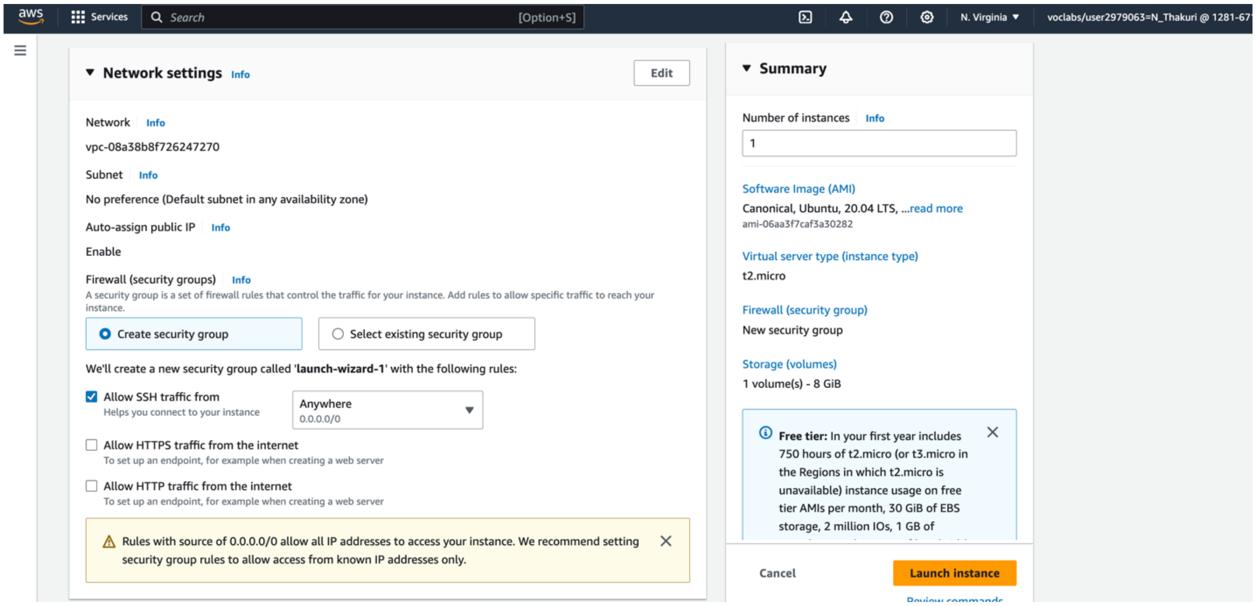
Private key file format

● .pem
For use with OpenSSH

○ .ppk
For use with PuTTY

⚠ When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** Learn more

6. The next step is to configure network settings. Security groups in AWS determine a set of access rules for both incoming and outgoing traffic in the EC2 instance. The settings include port ranges, IPs or security group IDs assigned to resources trying to access an EC2 instance. You can either select an existing security group or create a new one. For now, we are using default options.



▼ **Network settings** Info

Network Info

vpc-08a38b8f726247270

Subnet Info

No preference (Default subnet in any availability zone)

Auto-assign public IP Info

Enable

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

● Create security group ○ Select existing security group

We'll create a new security group called '**launch-wizard-1**' with the following rules:

☑ Allow SSH traffic from
Helps you connect to your instance

Anywhere
0.0.0.0/0

□ Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server

□ Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

▼ **Summary**

Number of instances Info

1

Software Image (AMI)

Canonical, Ubuntu, 20.04 LTS, ...read more
ami-06aa3f7caf3a30282

Virtual server type (instance type)

t2.micro

Firewall (security group)

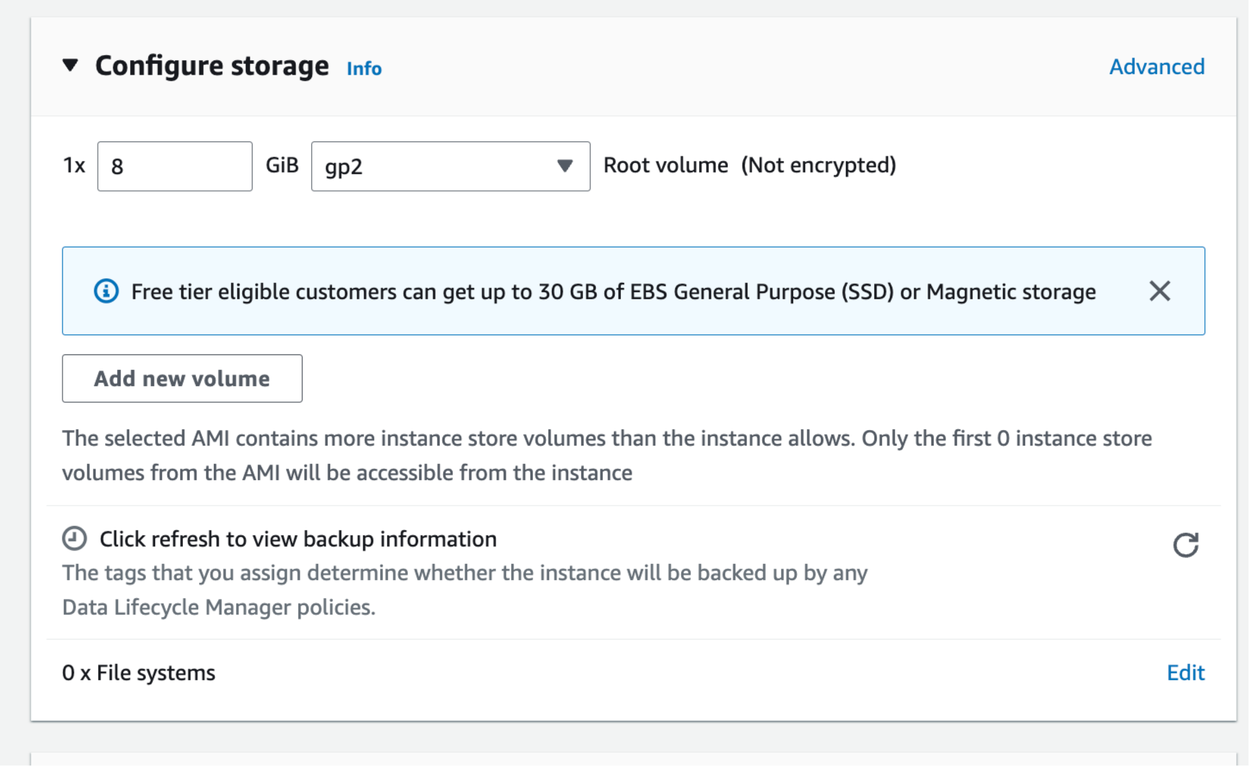
New security group

Storage (volumes)

1 volume(s) - 8 GiB

ⓘ **Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of

7. Specify the storage size in gigabytes and the storage type options. Your instance will be launched with the following storage device settings.



▼ Configure storage Info

Advanced

1x 8 GiB gp2 ▼ Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

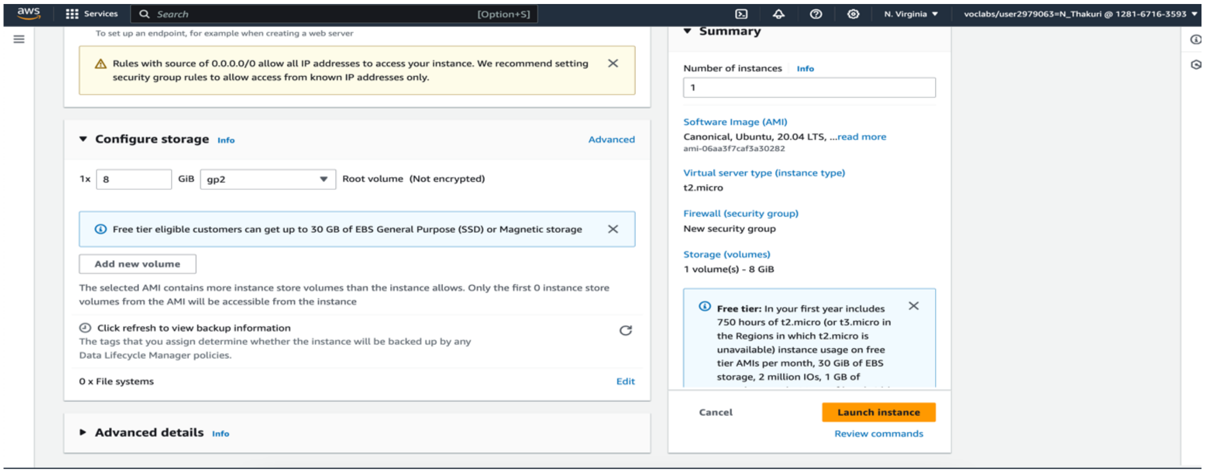
The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

Click refresh to view backup information

The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

0 x File systems

8. At last, review the summary of instance you are going to create and click Launch instance button.



To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

▼ Configure storage Info — Advanced

1x 8 GiB gp2 ▼ Root volume (Not encrypted)

ⓘ Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

Add new volume

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

Click refresh to view backup information
The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

0 x File systems — Edit

▶ Advanced details Info

▼ Summary

Number of instances Info
1

Software Image (AMI)
Canonical, Ubuntu, 20.04 LTS, ...read more
ami-06aa3f7caf3a30282

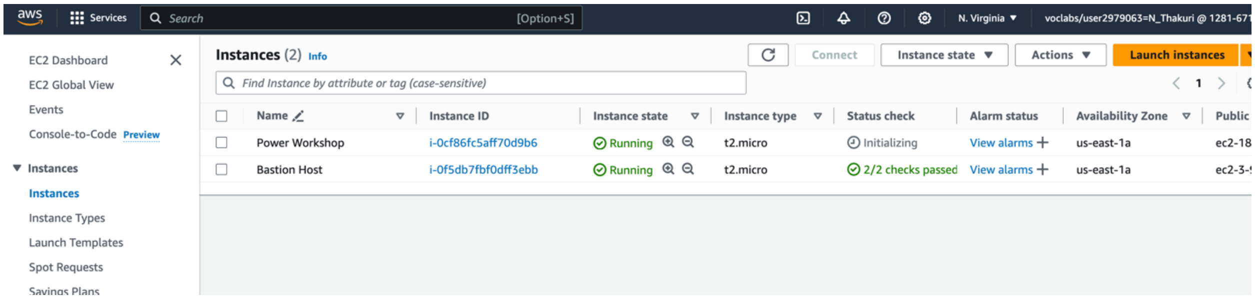
Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

ⓘ **Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of

9. You can view your launched instances in instance dashboard.



Instances (2) Info

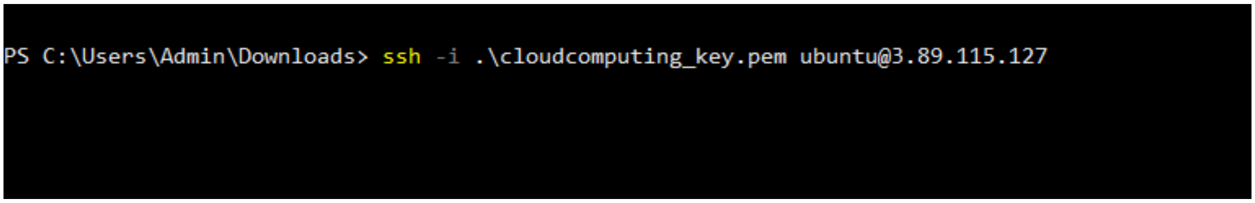
□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public
□	Power Workshop	i-0cf86fc5aff70d9b6	✓ Running	t2.micro	Initializing	View alarms +	us-east-1a	ec2-18
□	Bastion Host	i-0f5db7fbf0dff3ebb	✓ Running	t2.micro	✓ 2/2 checks passed	View alarms +	us-east-1a	ec2-3-

Connecting instance using SSH

1. Now that I have created the instance. Now you will see in this step how you can connect the instance using keypair. You can use puTTY or simply windows power shell for connecting to instance.
2. Navigate to the folder in your local machine where you have downloaded keypair.
3. Now we will use ssh command to securely connect to the instance. Before that, we need to provide the permission to the key.

    a. For Windows: icacls '.\directory keyname' /grant Owner:R
    b. For MAC: chmod 400 keyname

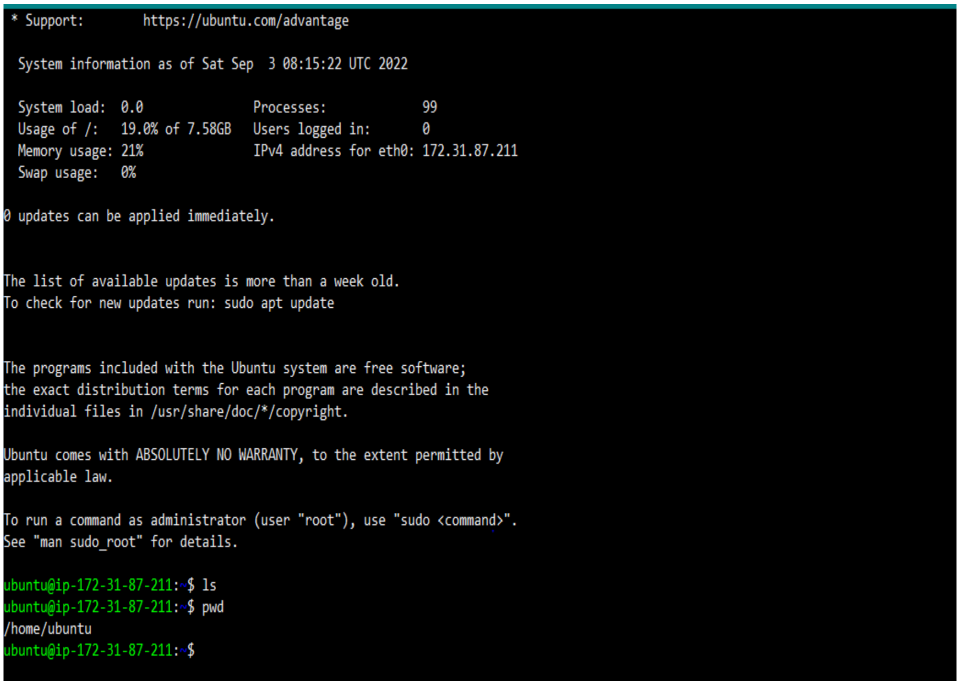
4. Use ssh command to securely connect to the instance.



```
PS C:\Users\Admin\Downloads> ssh -i .\cloudcomputing_key.pem ubuntu@3.89.115.127
```

Syntax: ssh -i keypairname username@public_ip_of_instance

5. Now you can access the created machine.



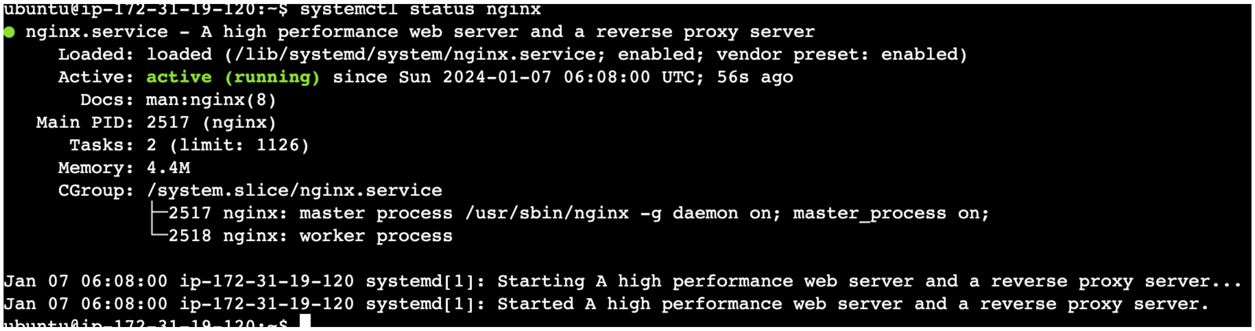
```
 * Support:        https://ubuntu.com/advantage

  System information as of Sat Sep  3 08:15:22 UTC 2022

  System load:  0.0               Processes:             99
  Usage of /:   19.0% of 7.58GB   Users logged in:       0
  Memory usage: 21%               IPv4 address for eth0: 172.31.87.211
  Swap usage:   0%

0 updates can be applied immediately.


The list of available updates is more than a week old.
To check for new updates run: sudo apt update


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-172-31-87-211:~$ ls
ubuntu@ip-172-31-87-211:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-87-211:~$
```

Part-II

Objectives

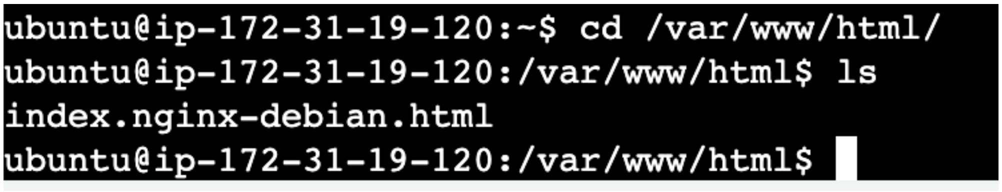
- Install and configure Nginx.
- Host a simple hello world page.

1. Once you ssh into your server, run the following commands to update your system and install Nginx:
    a. *sudo apt update*
    b. *sudo apt install nginx*
2. Once installed you can check the status using command:
    a. *systemctl status nginx*



```
ubuntu@ip-172-31-19-120:~$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2024-01-07 06:08:00 UTC; 56s ago
       Docs: man:nginx(8)
   Main PID: 2517 (nginx)
      Tasks: 2 (limit: 1126)
     Memory: 4.4M
     CGroup: /system.slice/nginx.service
             ├─2517 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
             └─2518 nginx: worker process

Jan 07 06:08:00 ip-172-31-19-120 systemd[1]: Starting A high performance web server and a reverse proxy server...
Jan 07 06:08:00 ip-172-31-19-120 systemd[1]: Started A high performance web server and a reverse proxy server.
ubuntu@ip-172-31-19-120:~$
```

3. Now you need to configure your website, for that navigate to /var/www/html folder, use command:
    a. cd /var/www/html



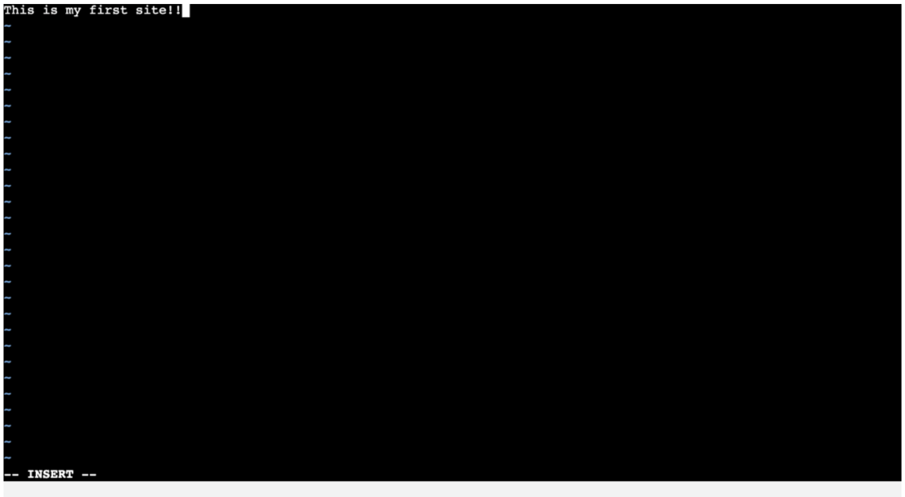
```
ubuntu@ip-172-31-19-120:~$ cd /var/www/html/
ubuntu@ip-172-31-19-120:/var/www/html$ ls
index.nginx-debian.html
ubuntu@ip-172-31-19-120:/var/www/html$
```

b. Create a new html file, use command:

- ➔ sudo vi myfile.html

  This command creates a file name myfile.html and opens the visual editor with sudo privileges.

c. Once you run the above command, it will open the visual editor, you need to press **i** so that you can enter your text in the editor, after pressing the letter **i,** you'll see the –INSERT—displayed on bottom of the screen.

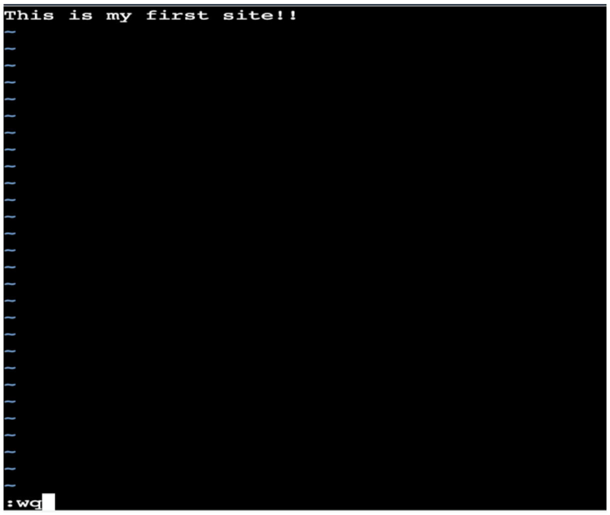


```
This is my first site!!
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --
```

d. After entering the text you need to save and quit the editor for that you need to use:

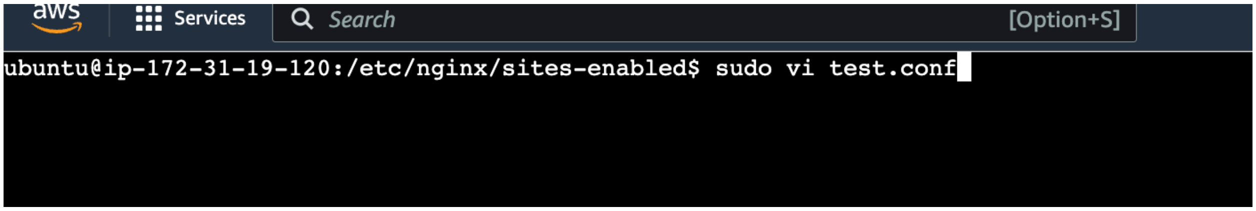
- ➔ At first press ESC
- ➔ Then press :wq

This will save and exit you from the editor.



```
This is my first site!!
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq
```

4. Now you need to configure the Nginx conf file , for that navigate to folder:
    a. /etc/nginx/sites-enabled
        i. Use command: cd /etc/nginx/sites-enabled
    b. Make a new conf file, test.conf
        i. Use command: sudo vi test.conf



```
ubuntu@ip-172-31-19-120:/etc/nginx/sites-enabled$ sudo vi test.conf
```

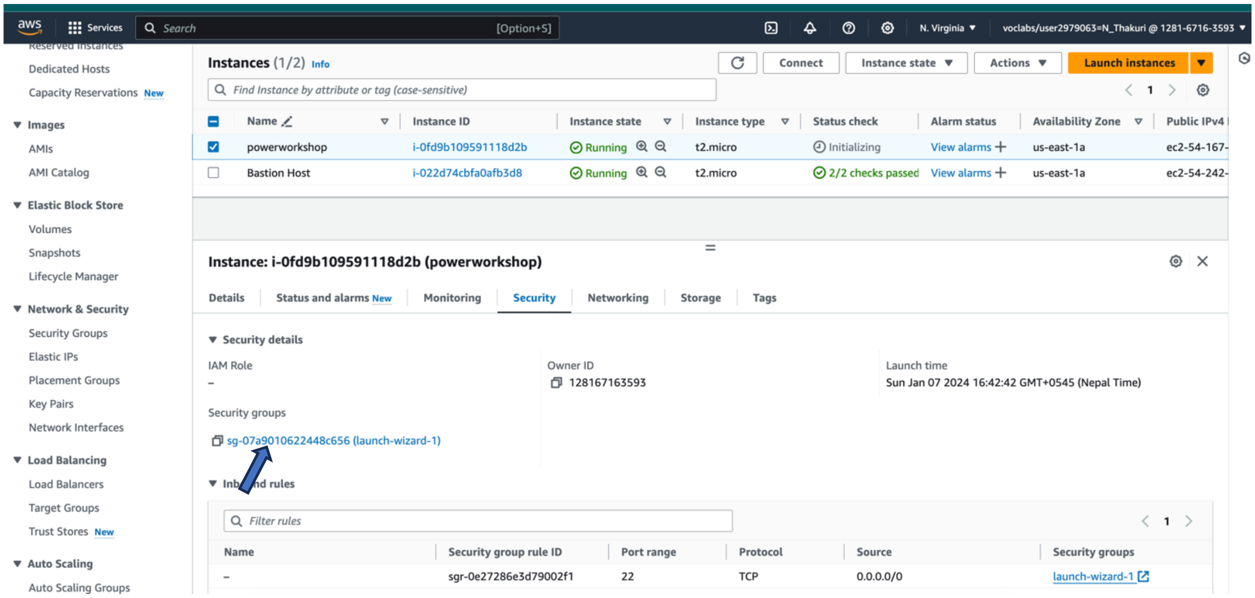
ii. Inside test.conf:

```
  server {
 listen 80;
server_name your_domain.com;
 root /var/www/html;
index myfile.html;
location / {
        try_files $uri $uri/ =404;
}
error_page 500 502 503 504 /50x.html;
```

```
                    location = /50x.html {
                root /usr/share/nginx/html;
            }
        }
```

listen 80: Specifies that this server block will listen on port 80 for incoming connections. Port 80 is the default port for HTTP.

5. You also need to open port 80 in security group.
   a. Select the instance.
   b. Click on Security tab as shown below.
   c. Click security group link pointed by arrow.



Instances (1/2) Info

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
☑	powerworkshop	i-0fd9b109591118d2b	Running	t2.micro	Initializing	View alarms +	us-east-1a	ec2-54-167-
□	Bastion Host	i-022d74cbfa0afb3d8	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-54-242-

Instance: i-0fd9b109591118d2b (powerworkshop)

Security details

IAM Role
–

Owner ID
128167163593

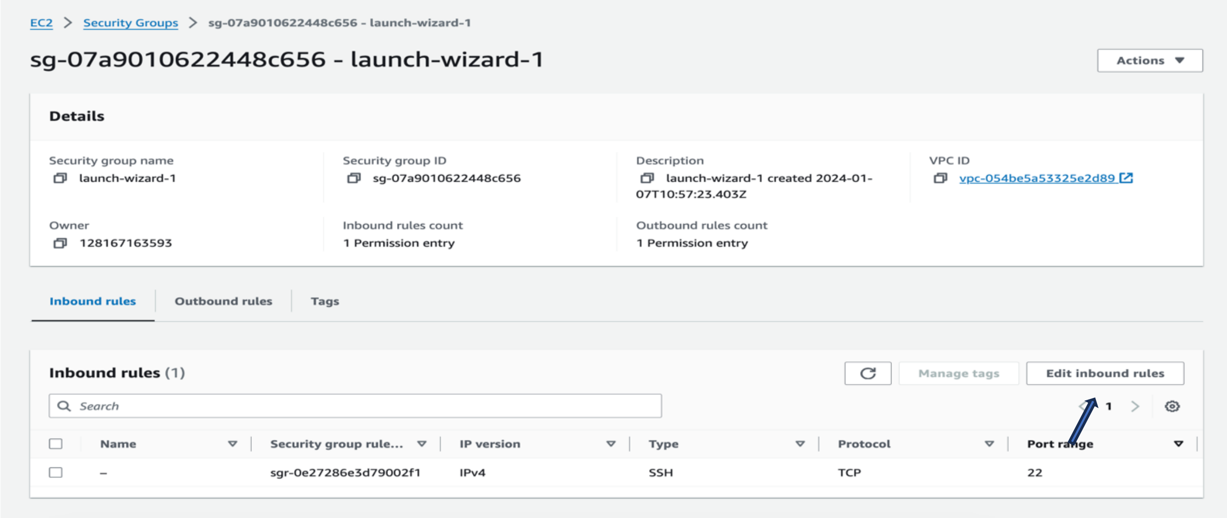
Launch time
Sun Jan 07 2024 16:42:42 GMT+0545 (Nepal Time)

Security groups
sg-07a9010622448c656 (launch-wizard-1)

Inbound rules

Name	Security group rule ID	Port range	Protocol	Source	Security groups
–	sgr-0e27286e3d79002f1	22	TCP	0.0.0.0/0	launch-wizard-1

d. Once you click the link, you'll be redirected to below page. Click edit inbound rules button.



EC2 > Security Groups > sg-07a9010622448c656 - launch-wizard-1

sg-07a9010622448c656 - launch-wizard-1

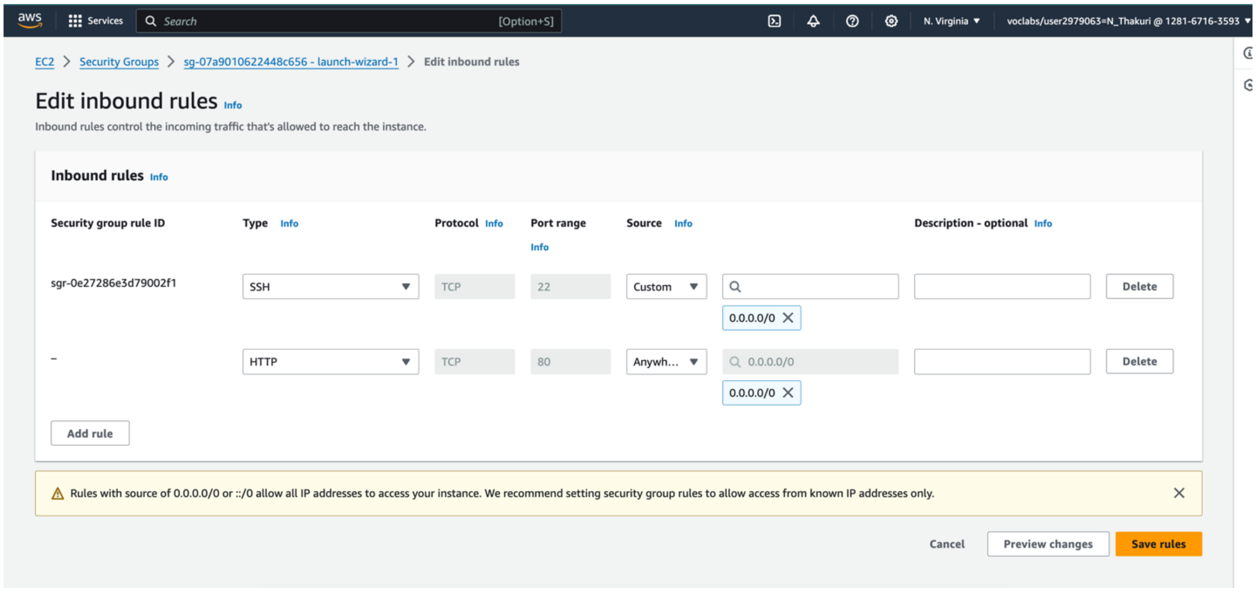
Details

Security group name launch-wizard-1	Security group ID sg-07a9010622448c656	Description launch-wizard-1 created 2024-01-07T10:57:23.403Z	VPC ID vpc-054be5a53325e2d89
Owner 128167163593	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

Inbound rules (1)

□	Name	Security group rule...	IP version	Type	Protocol	Port range
□	–	sgr-0e27286e3d79002f1	IPv4	SSH	TCP	22

e. Click on Add rule, open port 80, you can directly select type as HTTP or if the type is Custom TCP, then you need to manually enter 80 in port range. After adding rules, click save rules.



EC2 > Security Groups > sg-07a9010622448c656 - launch-wizard-1 > Edit inbound rules

Edit inbound rules Info

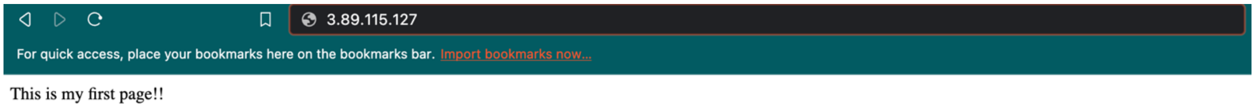
Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-0e27286e3d79002f1	SSH	TCP	22	Custom	0.0.0.0/0		Delete
-	HTTP	TCP	80	Anywh...	0.0.0.0/0		Delete

Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

6. Now, enter the public IP of your instance in web browser.



3.89.115.127
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

This is my first page!!