MIT-WPU

।। विश्वशान्तिर्ध्रुवं ध्रुवा ।।

Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

CET4034B: Cloud Infrastructure and Security

SCHOOL OF COMPUTER ENGINEERING AND TECHNOLOGY

T. Y. B. TECH. CSE(CYBERSECURITY AND FORENSICS)

Assignment:3

- CREATE AN ACCOUNT ON AWS.
- DEPLOY A WEBSITE FOR ADMISSION PORTAL ON THE EC2 SERVICE.
- CONFIGURE THE TRAFFIC RULES OF THE SERVER FOR A SPECIFIC NEED.
- CREATION OF APPLICATION LOAD BALANCER

Creating an account on AWS

Create a standalone AWS account

1. Open the Amazon Web Services home page . Visit web page: https://aws.amazon.com/
2. Choose Create an AWS account.

Note

If you signed in to AWS recently, that option might not be there. Instead, choose Sign in to the Console. Then, if Create a new AWS account still isn't visible, first choose Sign in to a different account, and then choose Create a new AWS account.

3. Enter your account information, and then choose **Verify email address**. This will send a verification code to your specified email address.

4. Enter your verification code, and then choose **Verify**.

Create a standalone AWS account

5. Enter a strong password for your root user, confirm it, and then choose **Continue**. AWS requires that your password meet the following conditions:

   - It must have a minimum of 8 characters and a maximum of 128 characters.
   - It must include a minimum of three of the following mix of character types: uppercase, lowercase, numbers, and ! @ # $ % ^ & * () <> [] {} | _+-= symbols.
   - It must not be identical to your AWS account name or email address.

6. Choose **Business** or **Personal**. Personal accounts and business accounts have the same features and functions.

7. Enter your company or personal information.

8. Read and accept the AWS Customer Agreement. Be sure that you read and understand the terms of the AWS Customer Agreement.

Create a standalone AWS account

9. Choose **Continue**. At this point, you'll receive an email message to confirm that your AWS account is ready to use. You can sign in to your new account by using the email address and password you provided during sign up. However, you can't use any AWS services until you finish activating your account.

10. Enter the information about your payment method, and then choose **Verify and Continue**. If you want to use a different billing address for your AWS billing information, choose **Use a new address**.

You can't proceed with the sign-up process until you add a valid payment method.

11. Enter your country or region code from the list, and then enter a phone number where you can be reached in the next few minutes.

12. Enter the code displayed in the CAPTCHA, and then submit.

Create a standalone AWS account

13. When the automated system contacts you, enter the PIN you receive and then submit.

14. Select one of the available AWS Support plans. For a description of the available Support plans and their benefits, see Compare AWS Support plans.

15. Choose **Complete sign up**. A confirmation page appears that indicates that your account is being activated.

16. Check your email and spam folder for an email message that confirms your account was activated. Activation usually takes a few minutes but can sometimes take up to 24 hours.

After you receive the activation message, you have full access to all AWS services.

Deploying a website for admission portal on the EC2 service

Steps to deploy a website on the EC2 instance

- AWS EC2 is one of the very most popular services of AWS.
- It is a virtual computer where you can deploy your application.

Step 1: Log in to your AWS account and open the EC2 service.

Step 2: Add name and tag.

Step3: Choose AMI(Amazon Machine Image).

Step 4: Select the Instance type.

Step 5: Create Key pair.

Step 6: Network settings.

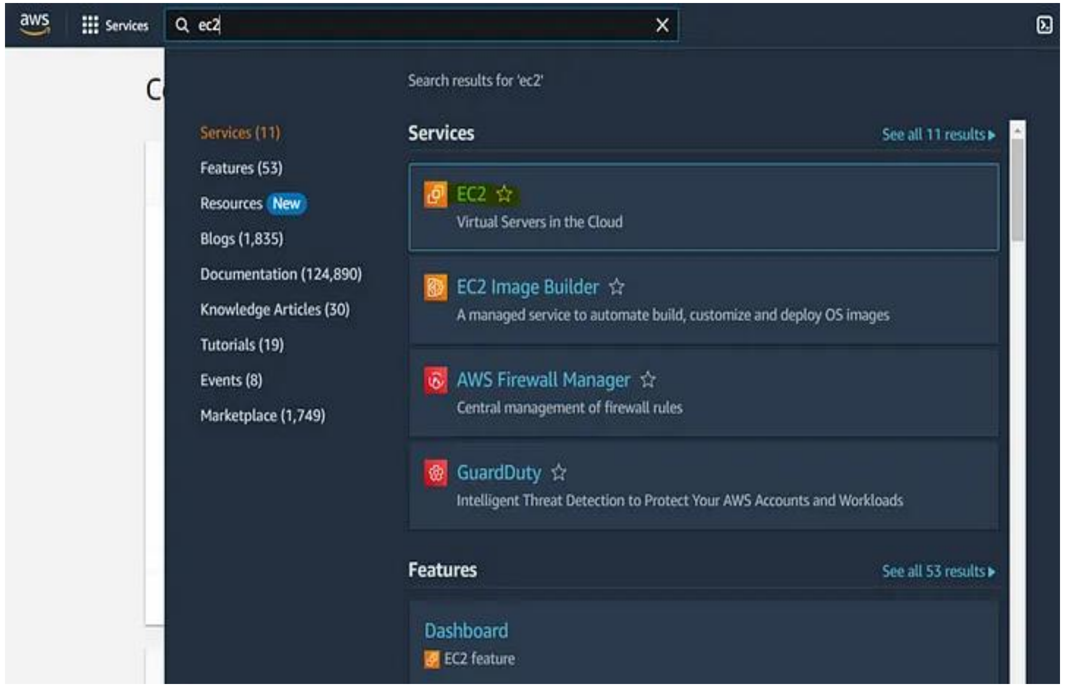
Step 7: Configure storage.

Step 8: Add user data in the Advance tab.

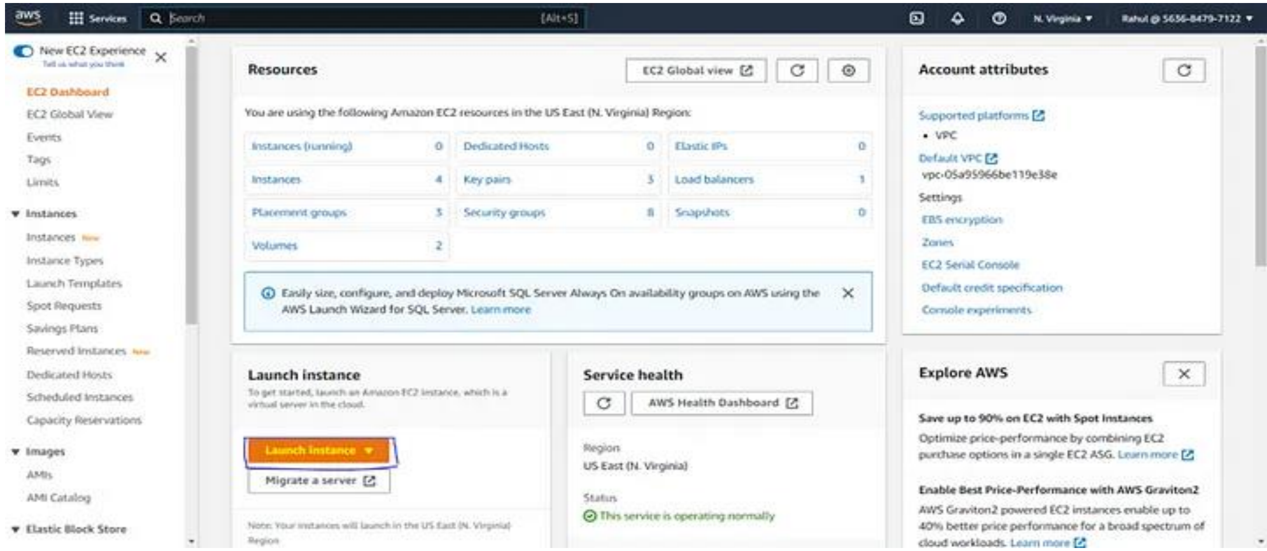
Steps to deploy a website on the EC2 instance

Step 1: Log in to your AWS account and open the EC2 service.

- Login to your AWS account and
- In the search bar search for **EC2**



- From the search result click on EC2
- This will opens up the EC2 dashboard as per the below screenshot shown



You are using the following Amazon EC2 resources in the US East (N. Virginia) Region:

Easily size, configure, and deploy Microsoft SQL Server Always On availability groups on AWS using the AWS Launch Wizard for SQL Server. Learn more

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

This service is operating normally

Save up to 90% on EC2 with Spot Instances

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

Enable Best Price-Performance with AWS Graviton2

AWS Graviton2 powered EC2 instances enable up to 40% better price performance for a broad spectrum of cloud workloads. Learn more

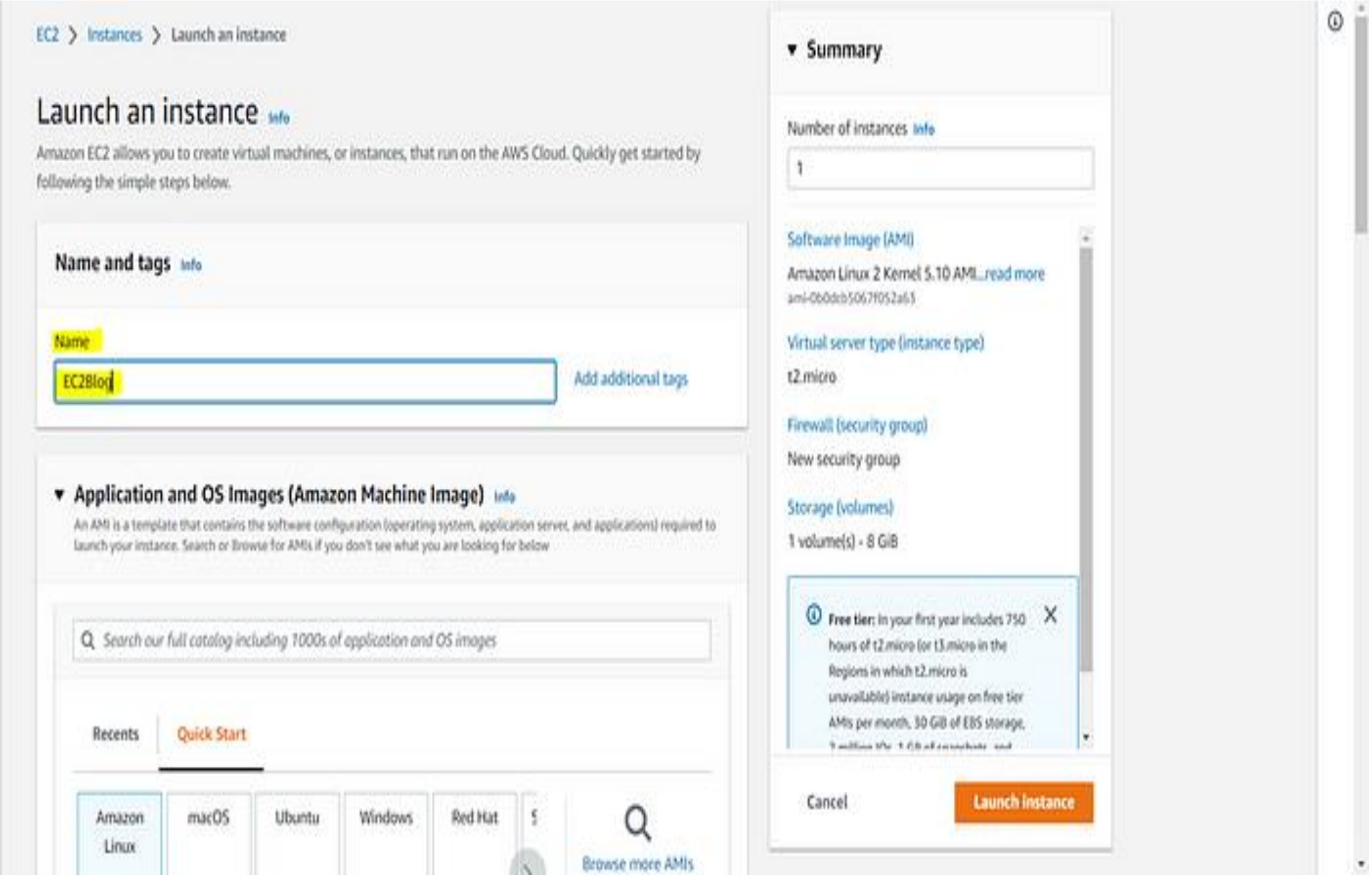
From the above screen just click on Lunch Instance and that will leads you to the next step.

Steps to deploy a website on the EC2 instance

Step 2: Add name and tag.

This is the main configuration screen for the EC2 instance, in this screen, you will set up Compute capacity, Networking settings, and storage.

- ✓ Give the Name of the instance as per your requirements.
- ✓ You can also add tags to easily identify your instance. (Tag is very useful when you are working with multiple EC2 instances)



EC2 > Instances > Launch an instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name: EC2Blog

▾ 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

▾ Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2 Kernel 5.10 AMI...read more
ami-0b0dcb5067f052a63

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

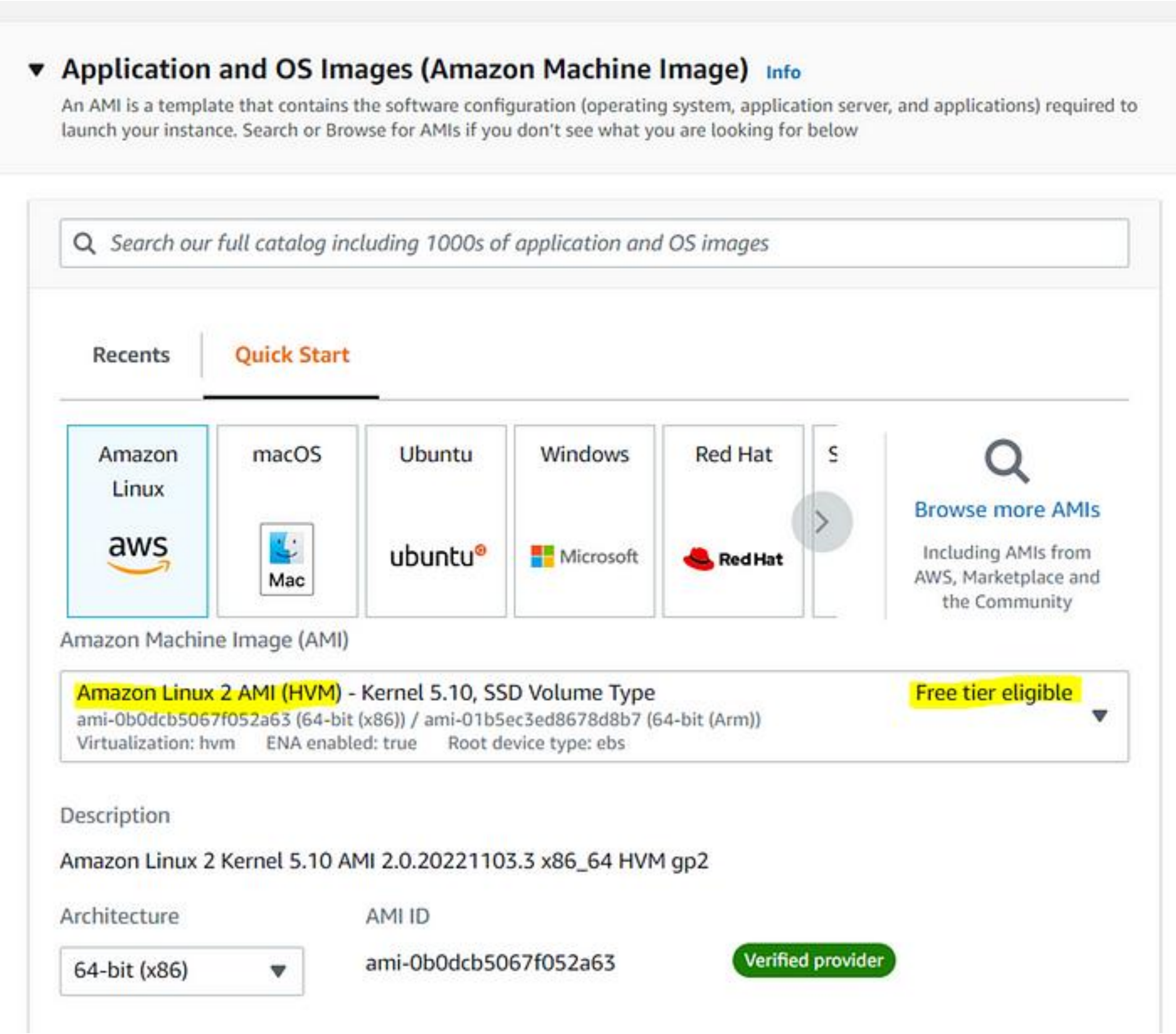
Storage (volumes): 1 volume(s) - 8 GiB

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,

Steps to deploy a website on the EC2 instance

Step3: Choose AMI(Amazon Machine Image).

- An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance.
- I am going with Amazon Linux 2 AMI which is in the free tier.
- Choose as per your requirements there are other options available as well as you can see in the screenshot. e.g. macOS, Ubuntu, Windows, Red Hat, etc.
- ***Note: If you are using it for learning purposes then make sure you are choosing the free tier otherwise you will get charged.***



▼ 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)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type — Free tier eligible
ami-0b0dcb5067f052a63 (64-bit (x86)) / ami-01b5ec3ed8678d8b7 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2 Kernel 5.10 AMI 2.0.20221103.3 x86_64 HVM gp2

Architecture: 64-bit (x86)

AMI ID: ami-0b0dcb5067f052a63

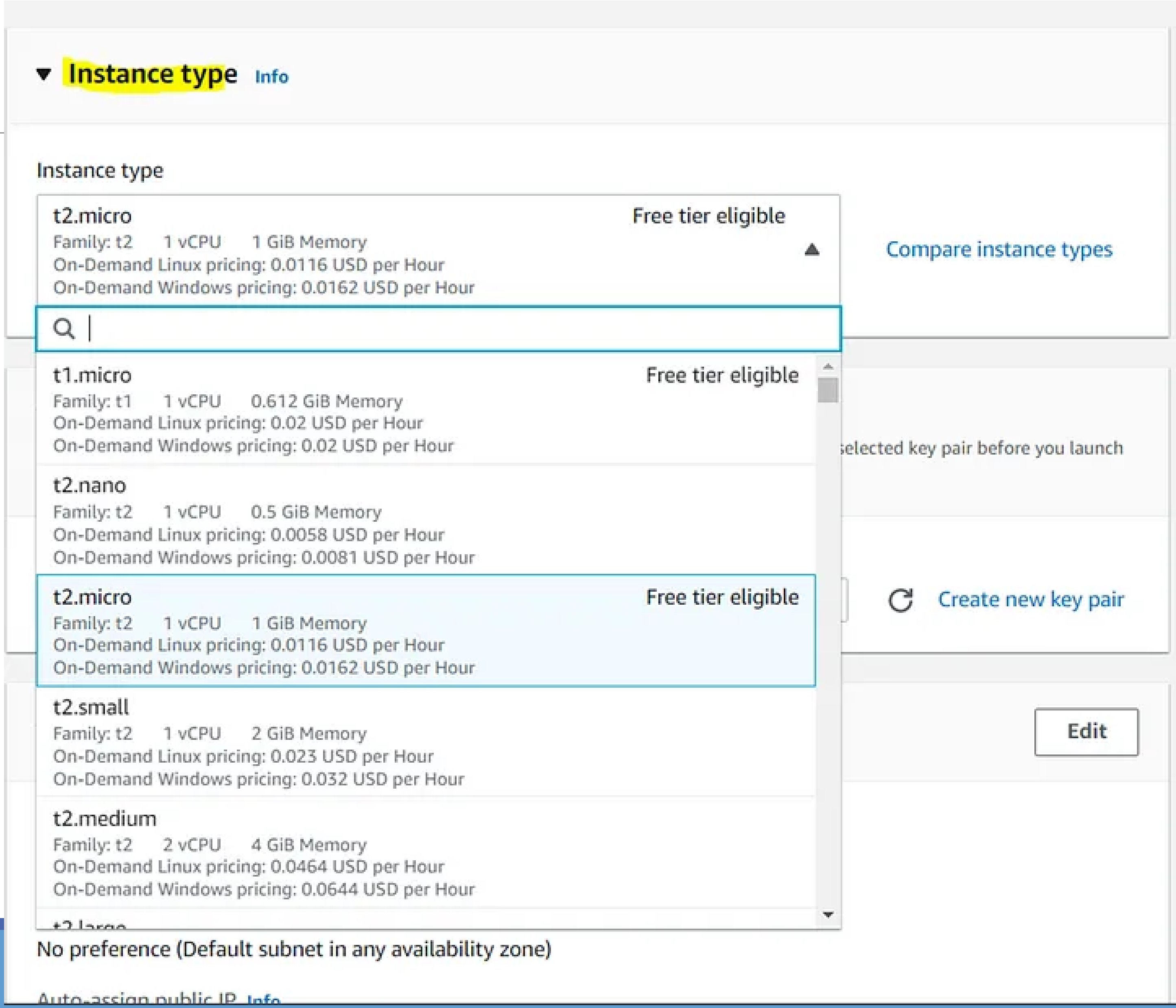
Verified provider

Steps to deploy a website on the EC2 instance

So right now, we’ll be using a T2 micro.

Step 4: Select the Instance type.

- So instance types are going to differ based on the number of CPUs they have, the amount of memory they have, and how much they cost.
- I have a T2 micro selected.
- This one is free tier eligible, so it will be free to launch one of them during an entire month if we leave it running,
- You could scroll down and look at other types of instances. For example, T1 micro is also free tier eligible, but that’s the older generation.
- If you wanted to compare the instance types, you will just click on that link, and it will show you all the types of instances here, as well as how much memory they have and so on.



▼ Instance type Info

Instance type

t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory
On-Demand Linux pricing: 0.0116 USD per Hour
On-Demand Windows pricing: 0.0162 USD per Hour

t1.micro Free tier eligible
Family: t1 1 vCPU 0.612 GiB Memory
On-Demand Linux pricing: 0.02 USD per Hour
On-Demand Windows pricing: 0.02 USD per Hour

t2.nano
Family: t2 1 vCPU 0.5 GiB Memory
On-Demand Linux pricing: 0.0058 USD per Hour
On-Demand Windows pricing: 0.0081 USD per Hour

t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory
On-Demand Linux pricing: 0.0116 USD per Hour
On-Demand Windows pricing: 0.0162 USD per Hour

t2.small
Family: t2 1 vCPU 2 GiB Memory
On-Demand Linux pricing: 0.023 USD per Hour
On-Demand Windows pricing: 0.032 USD per Hour

t2.medium
Family: t2 2 vCPU 4 GiB Memory
On-Demand Linux pricing: 0.0464 USD per Hour
On-Demand Windows pricing: 0.0644 USD per Hour

t2.large

selected key pair before you launch

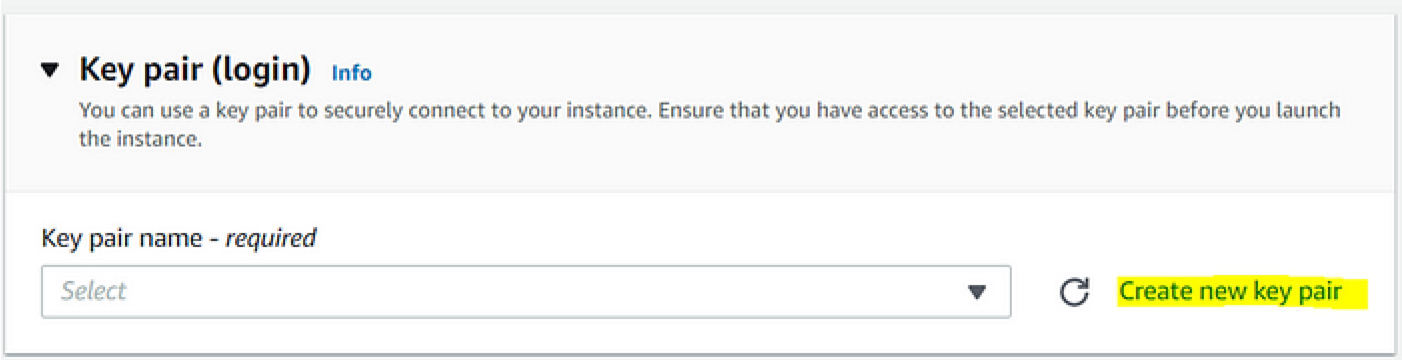
No preference (Default subnet in any availability zone)

Auto-assign public IP Info

Steps to deploy a website on the EC2 instance

Step 5: Create Key pair.

- The key pair is to log in to your instance. So it is required to create key pair.



Key pair (login) Info

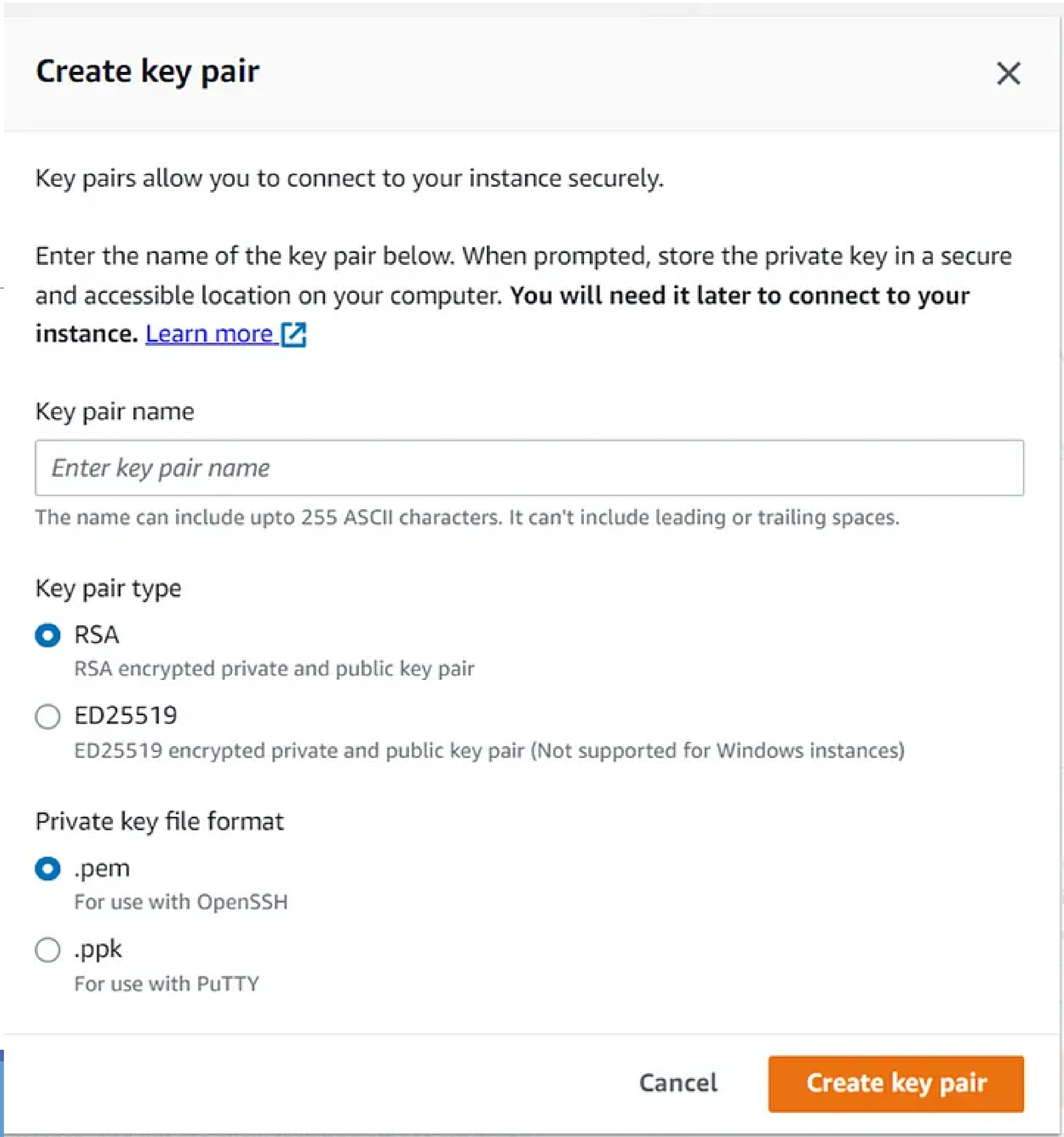
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

There is no key pair, so let’s go ahead and create a new key pair.

Steps to deploy a website on the EC2 instance

- Enter the name of your choice, then you need to choose a key pair type, so we'll be using the RSA encrypted.
- For key pair formats, if you have Mac or Linux, or Windows 10, then you can use the .pem format.
- If you have Windows less than version 10, for example, Windows 7 or Windows 8, then you can do a little shortcut and directly use a PPK, which is going to be used for PuTTY.



Create key pair

Key pairs allow you to connect to your instance securely.

Enter the name of the key pair below. 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

Key pair name

Enter key pair name

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 (Not supported for Windows instances)

Private key file format

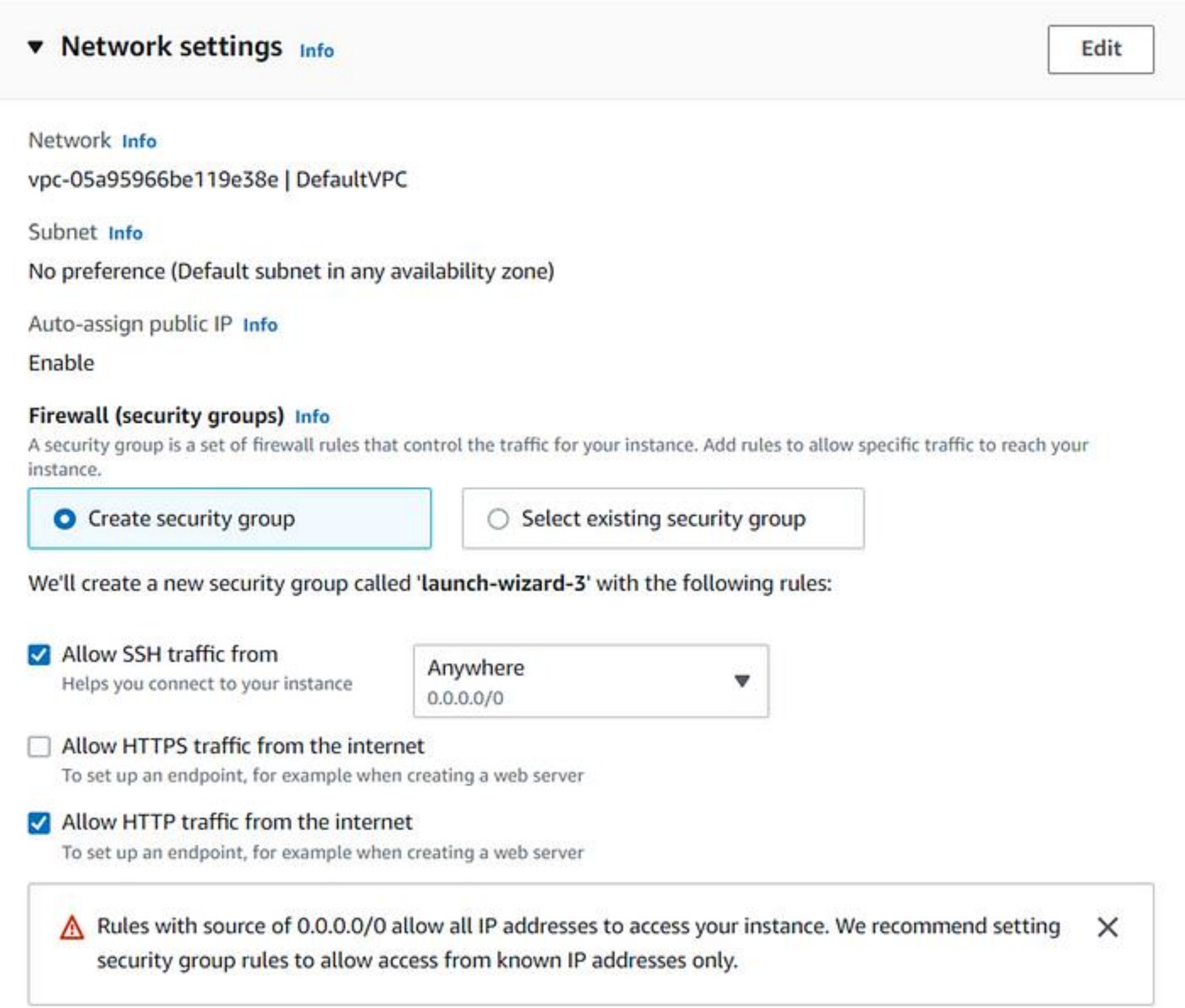
● .pem
For use with OpenSSH

○ .ppk
For use with PuTTY

Steps to deploy a website on the EC2 instance

Step 6: Network settings.

- we are not going to touch anything in the network settings
- The First security group created will be called launch-wizard-1, Here in the screenshot, it is launch-wizard-3 because I have already created 2.
- we can define multiple rules, so the first rule we want to have is to allow SSH traffic from anywhere.
- we also want to allow HTTP traffic from the internet, check mark allow HTTP traffic from the internet and this is because we're going to launch a web server
- We're not going to use HTTPS for now, we don't need to tick the second box.



▾ Network settings Info

Edit

Network Info

vpc-05a95966be119e38e | DefaultVPC

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-3**' with the following rules:

☑ Allow SSH traffic from — Anywhere 0.0.0.0/0 ▾
Helps you connect to your instance

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

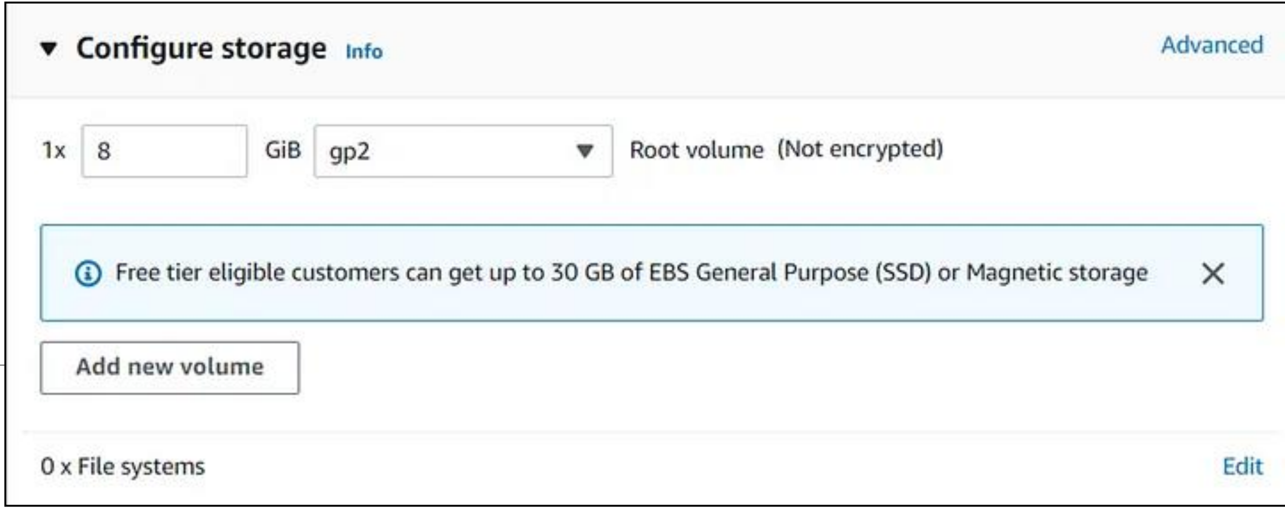
☑ Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

⚠ 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. ✕

Steps to deploy a website on the EC2 instance

Step 7: Configure storage.

- Let’s configure the storage, as we can see, we have eight gigabytes of gp2 root volume
- In the free tier, we can get up to 30 gigabytes of EBS General Purpose SSD storage, And we only have one volume necessary.
- If you go into advanced, you could configure them
- one important thing to note in here is the deletion on termination. By default it is enabled to yes, That means that once we terminate our EC2 instance, then that volume is also going to be deleted.



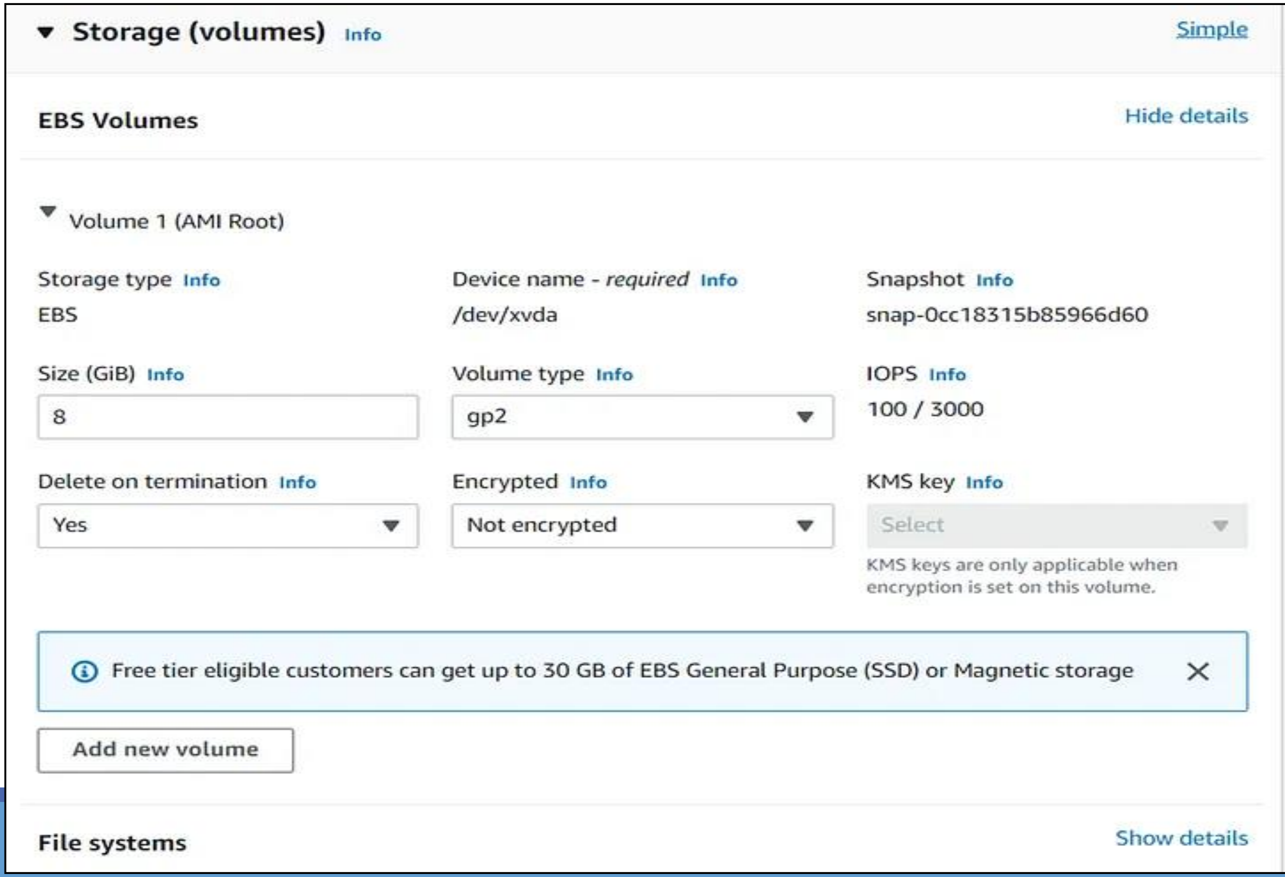
▼ Configure storage Info — Advanced

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

0 x File systems — Edit



▼ Storage (volumes) Info — Simple

EBS Volumes — Hide details

▼ Volume 1 (AMI Root)

Storage type Info: EBS

Device name - *required* Info: /dev/xvda

Snapshot Info: snap-0cc18315b85966d60

Size (GiB) Info: 8

Volume type Info: gp2 ▼

IOPS Info: 100 / 3000

Delete on termination Info: Yes ▼

Encrypted Info: Not encrypted ▼

KMS key Info: Select ▼

KMS keys are only applicable when encryption is set on this volume.

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

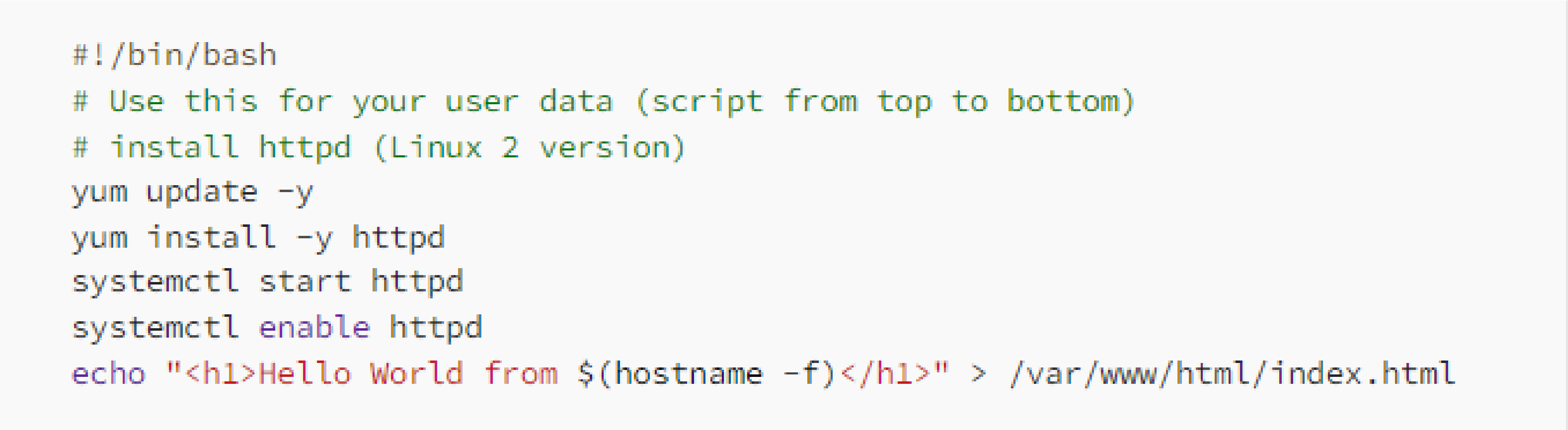
Add new volume

File systems — Show details

Steps to deploy a website on the EC2 instance

Step 8: Add user data in the Advance tab.

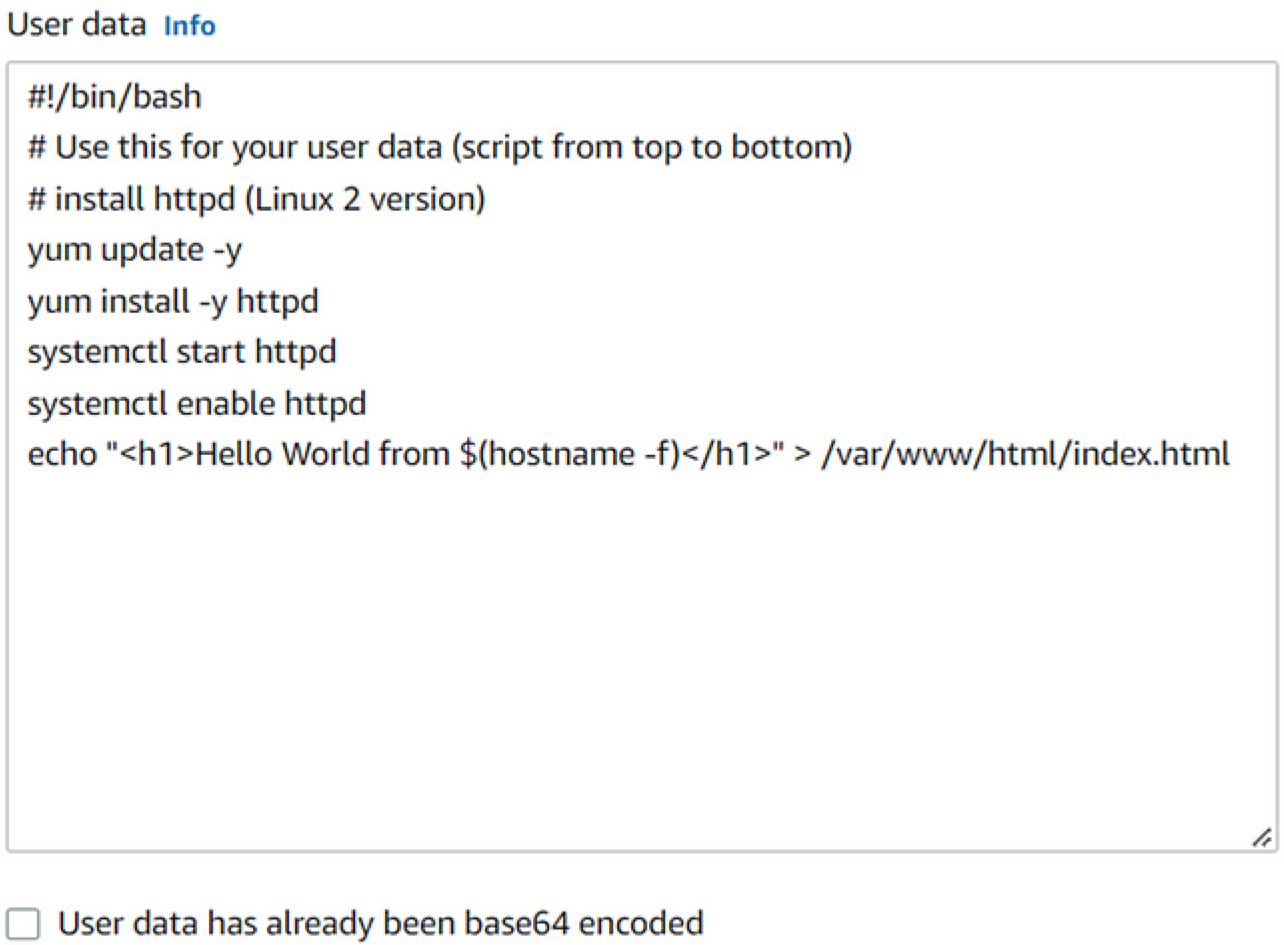
- Next for advanced details, this is where it gets interesting. so let’s scroll down all the way to the bottom.
- In User data, we will pass a script, so some comments, to our EC2 instance to execute on the first launch of our E2 instance and only the first launch.
- And therefore, on the first launch, we want to be able to pass these commands right here.
- Paste the following code into user data



```
#!/bin/bash
# Use this for your user data (script from top to bottom)
# install httpd (Linux 2 version)
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
echo "<h1>Hello World from $(hostname -f)</h1>" > /var/www/html/index.html
```

Steps to deploy a website on the EC2 instance

- it’s going to update a few things, then install the HTPD web server on the machine, and then write aHTML file,



User data Info

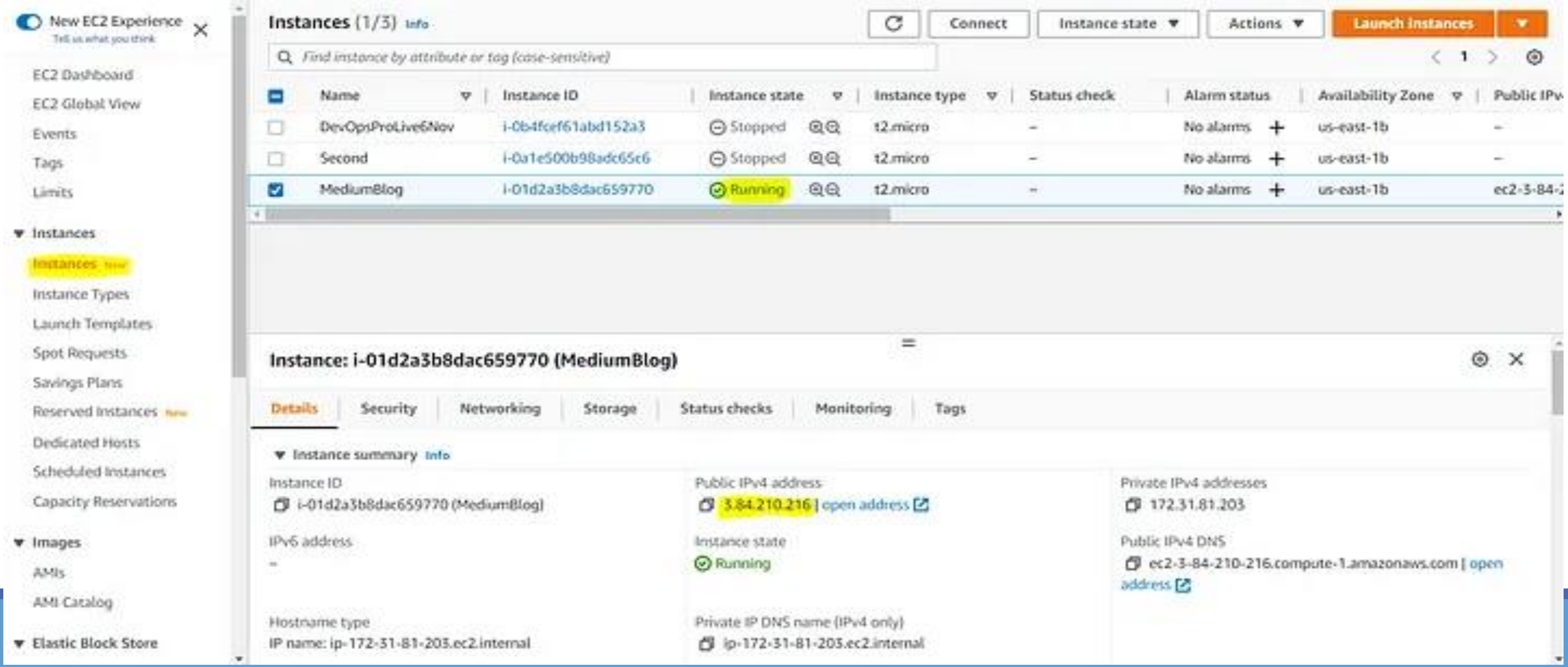
```
#!/bin/bash
# Use this for your user data (script from top to bottom)
# install httpd (Linux 2 version)
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
echo "<h1>Hello World from $(hostname -f)</h1>" > /var/www/html/index.html
```

□ User data has already been base64 encoded

Steps to deploy a website on the EC2 instance

Step 9: Verify the installation

- Go to EC2 dashboard and wait for some time to change the instance state to Running.
- Once it is running then copy IPv4 address and paste it into a new browser.



Instances (1/3) Info

☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv
☐	DevOpsProLive6Nov	i-0b4fcef61abd152a3	Stopped	t2.micro	–	No alarms	us-east-1b	–
☐	Second	i-0a1e500b98adc65c6	Stopped	t2.micro	–	No alarms	us-east-1b	–
☑	MediumBlog	i-01d2a3b8dac659770	Running	t2.micro	–	No alarms	us-east-1b	ec2-3-84-2

Instance: i-01d2a3b8dac659770 (MediumBlog)

Instance summary Info

Instance ID: i-01d2a3b8dac659770 (MediumBlog)

Public IPv4 address: 3.84.210.216 | open address

Private IPv4 addresses: 172.31.81.203

IPv6 address: –

Instance state: Running

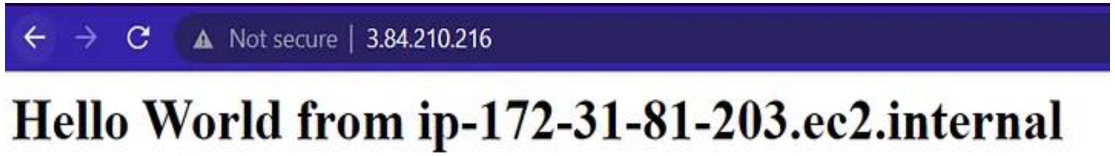
Public IPv4 DNS: ec2-3-84-210-216.compute-1.amazonaws.com | open address

Hostname type: IP name: ip-172-31-81-203.ec2.internal

Private IP DNS name (IPv4 only): ip-172-31-81-203.ec2.internal

Steps to deploy a website on the EC2 instance

Congratulations!! 🎉 You have successfully deployed your first ever website on EC2.



Not secure | 3.84.210.216

Hello World from ip-172-31-81-203.ec2.internal

Configuring the Traffic rules of the Server for a specific need

Configuring the Traffic rules of the Server for a specific need

- To configure traffic rules for a server on AWS, typically you would use security groups and network access control lists (ACLs).
- **Here's a step-by-step guide on how to configure traffic rules for a specific need:**

1. **Identify the Requirements:** Understand what type of traffic you need to allow or restrict. This could be HTTP, HTTPS, SSH, RDP, custom ports, etc.

2. **Use Security Groups:** Security groups act as a virtual firewall for your instance to control inbound and outbound traffic. Each instance in AWS can be associated with one or more security groups. Here's how to configure them:
   - ✓ Go to the AWS Management Console and open the EC2 dashboard.
   - ✓ In the navigation pane, choose "Security Groups".
   - ✓ Select the security group associated with your instance.
   - ✓ Click on the "Inbound rules" tab.
   - ✓ Add rules to allow traffic from specific IP addresses or ranges, on specific ports, as per your requirements.
   - ✓ Similarly, configure outbound rules if needed.

Configuring the Traffic rules of the Server for a specific need

3. Use Network ACLs (NACLs): NACLs are stateless and operate at the subnet level. They can be used to control traffic at the subnet level by allowing or denying traffic based on the rules you define. Here's how to configure them:

- ✓ In the EC2 dashboard, go to "Network ACLs" in the navigation pane.
- ✓ Select the appropriate NACL associated with your subnet.
- ✓ Configure inbound and outbound rules similar to security groups, but keep in mind that NACLs are evaluated in order, and the first rule that matches is applied.

4. Testing and Monitoring:

- After configuring the rules, it's essential to test them to ensure they are functioning as expected.
- You can do this by attempting to access your server from allowed IP addresses or trying to access specific ports.
- Additionally, regularly monitor your traffic logs and adjust rules as necessary based on usage patterns and security requirements.

Configuring the Traffic rules of the Server for a specific need

5. Follow Security Best Practices:

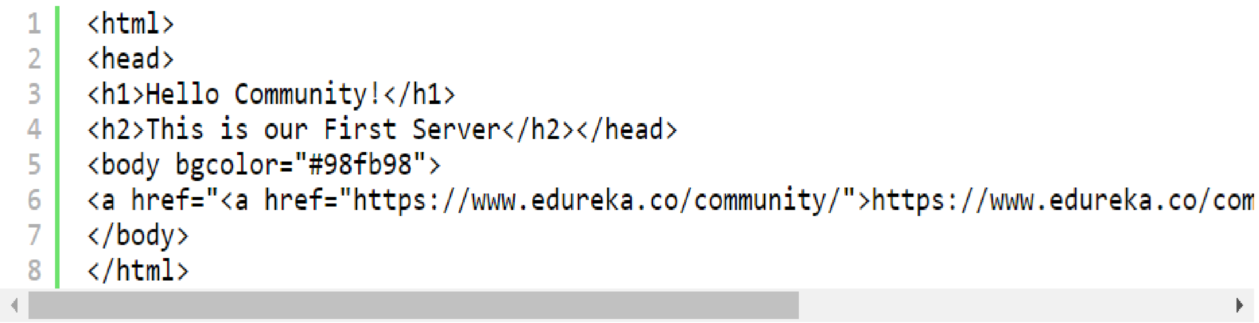
- ✓ Ensure that your security groups and NACLs follow security best practices, such as the principle of least privilege.
- ✓ Only allow the minimum necessary traffic to your server to reduce the attack surface.

6. Automate if Possible:

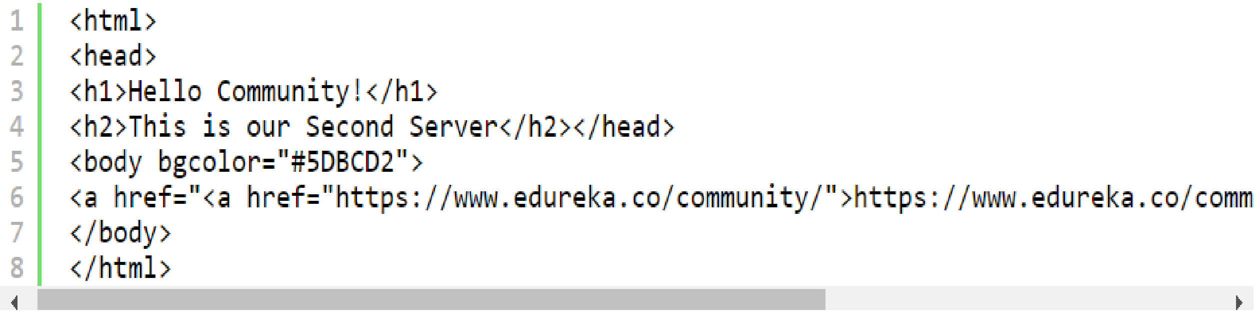
- ✓ If you have a dynamic environment with changing requirements, consider automating the configuration of security groups and NACLs using AWS APIs, AWS CloudFormation, or Infrastructure as Code (IaC) tools like AWS CloudFormation, Terraform, or AWS CDK

By following these steps, you can configure traffic rules for your server on AWS to meet your specific needs while maintaining security and compliance.

Create an HTML file named “*index.html*”



```
<html>
<head>
<h1>Hello Community!</h1>
<h2>This is our First Server</h2></head>
<body bgcolor="#98fb98">
<a href="<a href="https://www.edureka.co/community/">https://www.edureka.co/com
</body>
</html>
```



```
<html>
<head>
<h1>Hello Community!</h1>
<h2>This is our Second Server</h2></head>
<body bgcolor="#5DBCD2">
<a href="<a href="https://www.edureka.co/community/">https://www.edureka.co/comm
</body>
</html>
```

Creation of Application Load Balancer

Creation of Application Load Balancer

Creating an Application Load Balancer (ALB) on AWS involves several steps. Below is a step-by-step guide to create an ALB:

1. **Sign in to the AWS Management Console**: Go to the AWS Management Console at https://aws.amazon.com/ and sign in to your AWS account.
2. **Navigate to the EC2 Service**: Once logged in, navigate to the EC2 service by clicking on "Services" in the top-left corner of the screen and selecting "EC2" under the "Compute" section.
3. **Go to Load Balancers**: In the EC2 dashboard, locate and click on "Load Balancers" in the navigation pane.
4. **Create Load Balancer**: Click on the "Create Load Balancer" button.
5. **Select Load Balancer Type**: Choose the "Application Load Balancer" option. Click "Create".

Creation of Application Load Balancer

6. Configure Load Balancer: Fill out the configuration details:

- **Name**: Provide a name for your load balancer.
- **Scheme**: Choose whether your load balancer should be internet-facing or internal.
- **Listeners**: Define the listener configuration (e.g., HTTP or HTTPS) and the ports.
- **Availability Zones**: Select the availability zones where you want your load balancer to distribute traffic.
- **Security Settings**: Configure security settings such as SSL certificates if using HTTPS.
- **Configure Security Groups**: Select existing security groups or create new ones to control traffic to your load balancer.
- **Configure Routing**: Define target groups to route traffic to specific instances based on rules (e.g., based on path patterns or host headers).
- **Tags**: Optionally, add tags to your load balancer for easier management and identification.

Creation of Application Load Balancer

7. Register Targets: After creating the load balancer, register the targets (such as EC2 instances or IP addresses) with the target group associated with the load balancer. This tells the load balancer where to forward incoming requests.

8. Review and Create: Review your load balancer configuration and click "Create" to create the ALB.

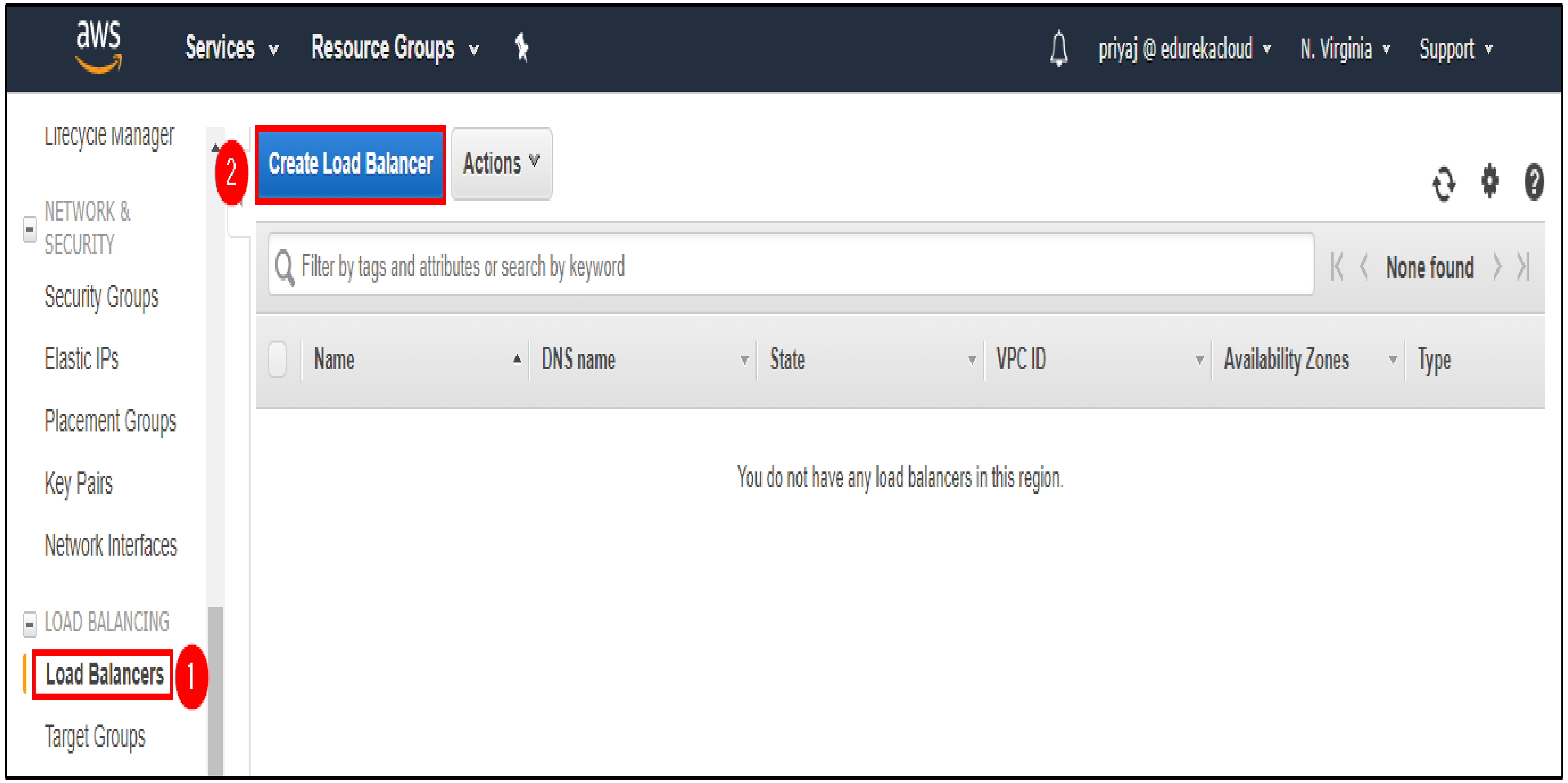
9. Wait for Creation: It may take a few minutes for the ALB to be created. Once created, you'll see it listed in the Load Balancers dashboard with its DNS name.

10. DNS Configuration: Use the DNS name provided by AWS to point your domain to the ALB if you're using a custom domain.

After following these steps, your Application Load Balancer will be ready to distribute incoming application traffic across multiple targets, providing high availability, fault tolerance, and scalability for your applications running on AWS.

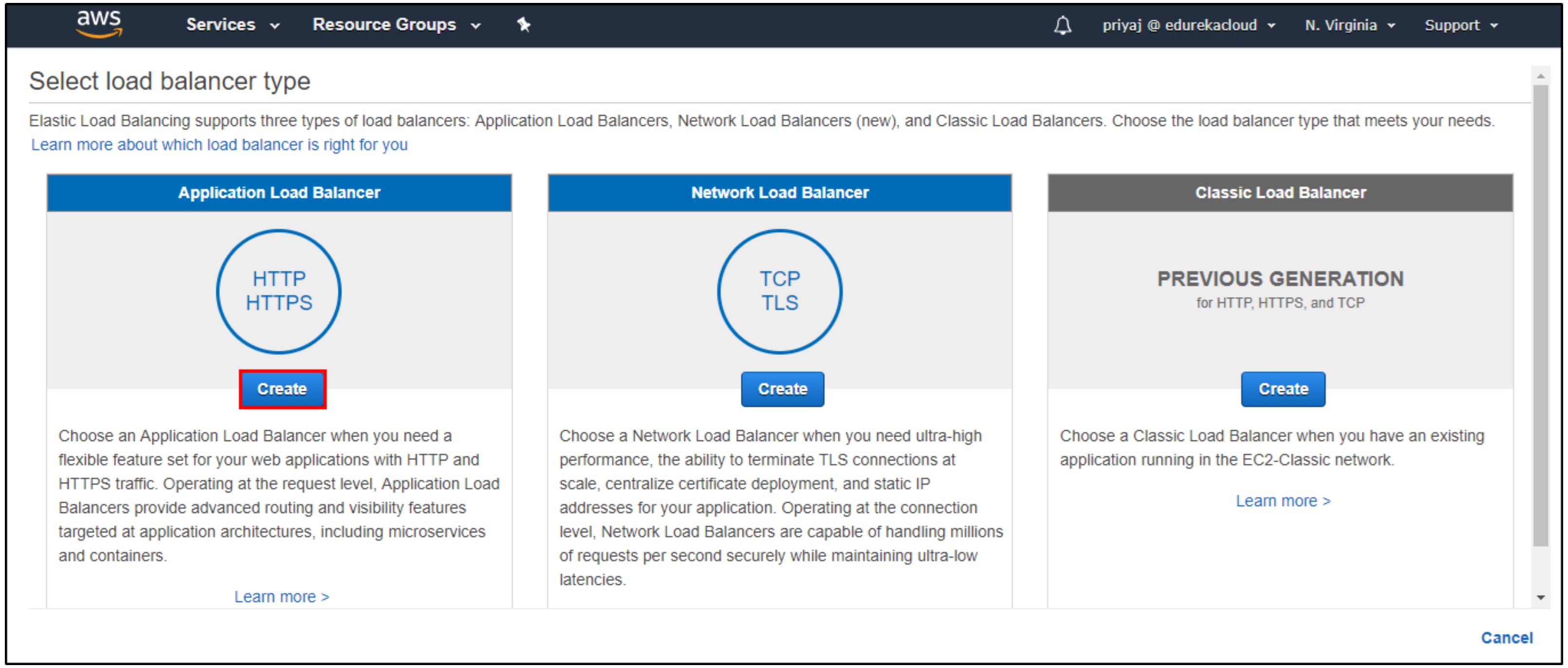
Creation of Application Load Balancer

Step 1: In the navigation pane, under Load Balancing, choose Load Balancers and create load balancer.



You do not have any load balancers in this region.

Step 2 : Select the “Load balancer type“. In this we are going to use HTTP/HTTPS



Select load balancer type

Elastic Load Balancing supports three types of load balancers: Application Load Balancers, Network Load Balancers (new), and Classic Load Balancers. Choose the load balancer type that meets your needs.
Learn more about which load balancer is right for you

Application Load Balancer

HTTP
HTTPS

Create

Choose an Application Load Balancer when you need a flexible feature set for your web applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.

Learn more >

Network Load Balancer

TCP
TLS

Create

Choose a Network Load Balancer when you need ultra-high performance, the ability to terminate TLS connections at scale, centralize certificate deployment, and static IP addresses for your application. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.

Classic Load Balancer

PREVIOUS GENERATION
for HTTP, HTTPS, and TCP

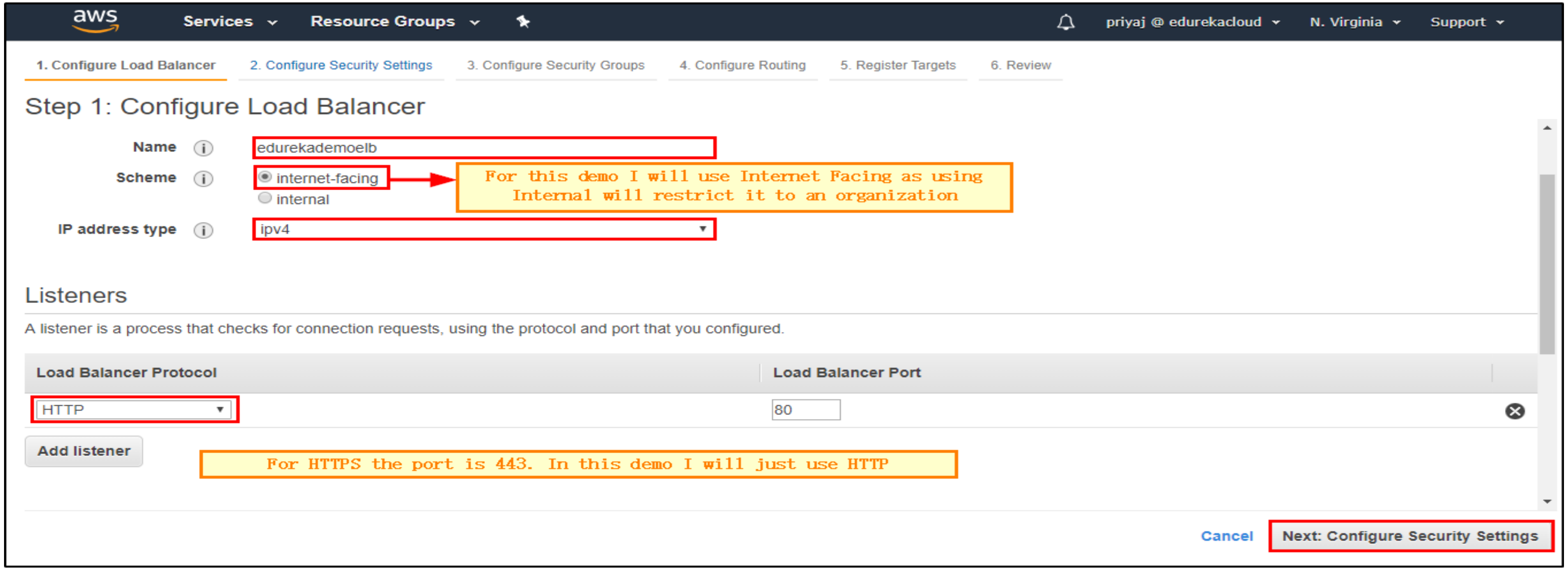
Create

Choose a Classic Load Balancer when you have an existing application running in the EC2-Classic network.

Learn more >

Cancel

Step 3: You have to "Configure your load balancer". Add name, Schema and IP address type(which is IPV4) in this case.



Step 1: Configure Load Balancer

Name: edurekademoelb

Scheme: ● internet-facing ○ internal

For this demo I will use Internet Facing as using Internal will restrict it to an organization

IP address type: ipv4

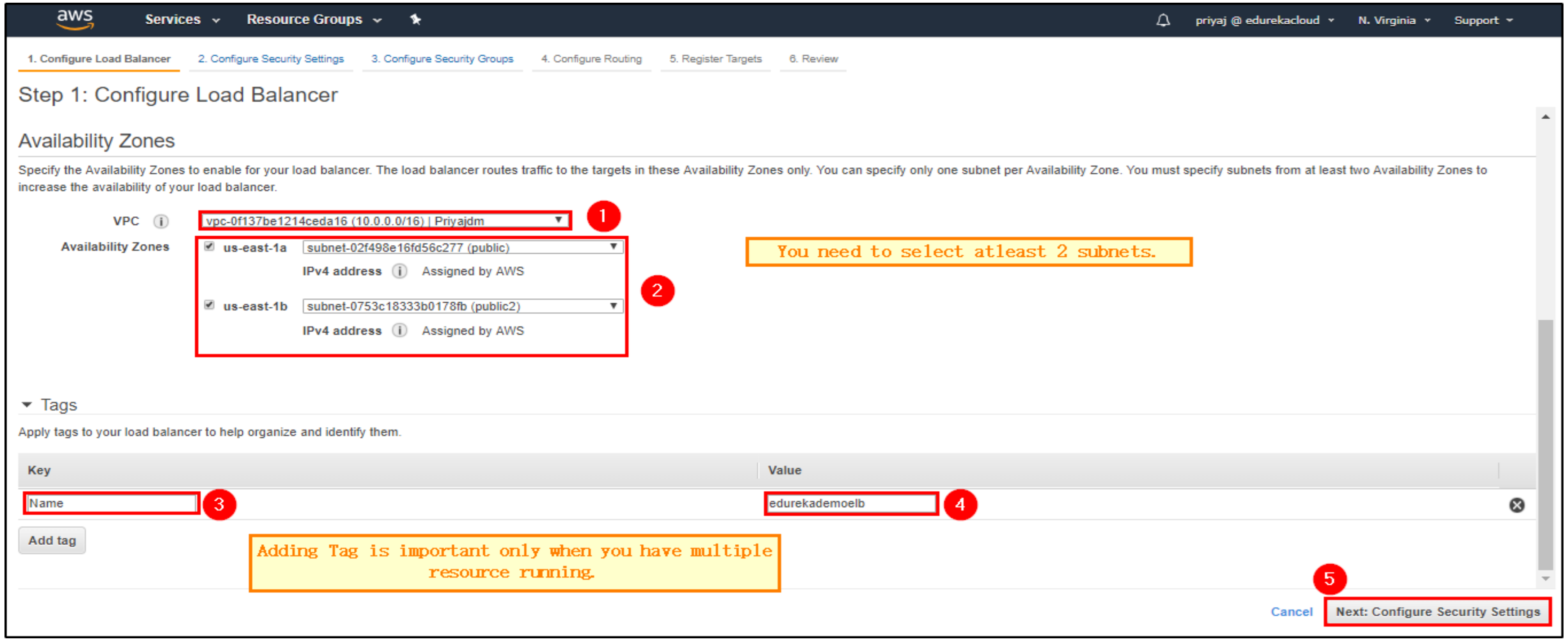
Listeners

A listener is a process that checks for connection requests, using the protocol and port that you configured.

Load Balancer Protocol	Load Balancer Port
HTTP	80

For HTTPS the port is 443. In this demo I will just use HTTP

Step 4: Select your VPC for “Configure Load Balancer“. Add your VPC and Availability Zones.



Step 1: Configure Load Balancer

Availability Zones

Specify the Availability Zones to enable for your load balancer. The load balancer routes traffic to the targets in these Availability Zones only. You can specify only one subnet per Availability Zone. You must specify subnets from at least two Availability Zones to increase the availability of your load balancer.

VPC: vpc-0f137be1214ceda16 (10.0.0.0/16) | Priyajdm

Availability Zones:
- ☑ us-east-1a: subnet-02f498e16fd56c277 (public); IPv4 address: Assigned by AWS
- ☑ us-east-1b: subnet-0753c18333b0178fb (public2); IPv4 address: Assigned by AWS

You need to select atleast 2 subnets.

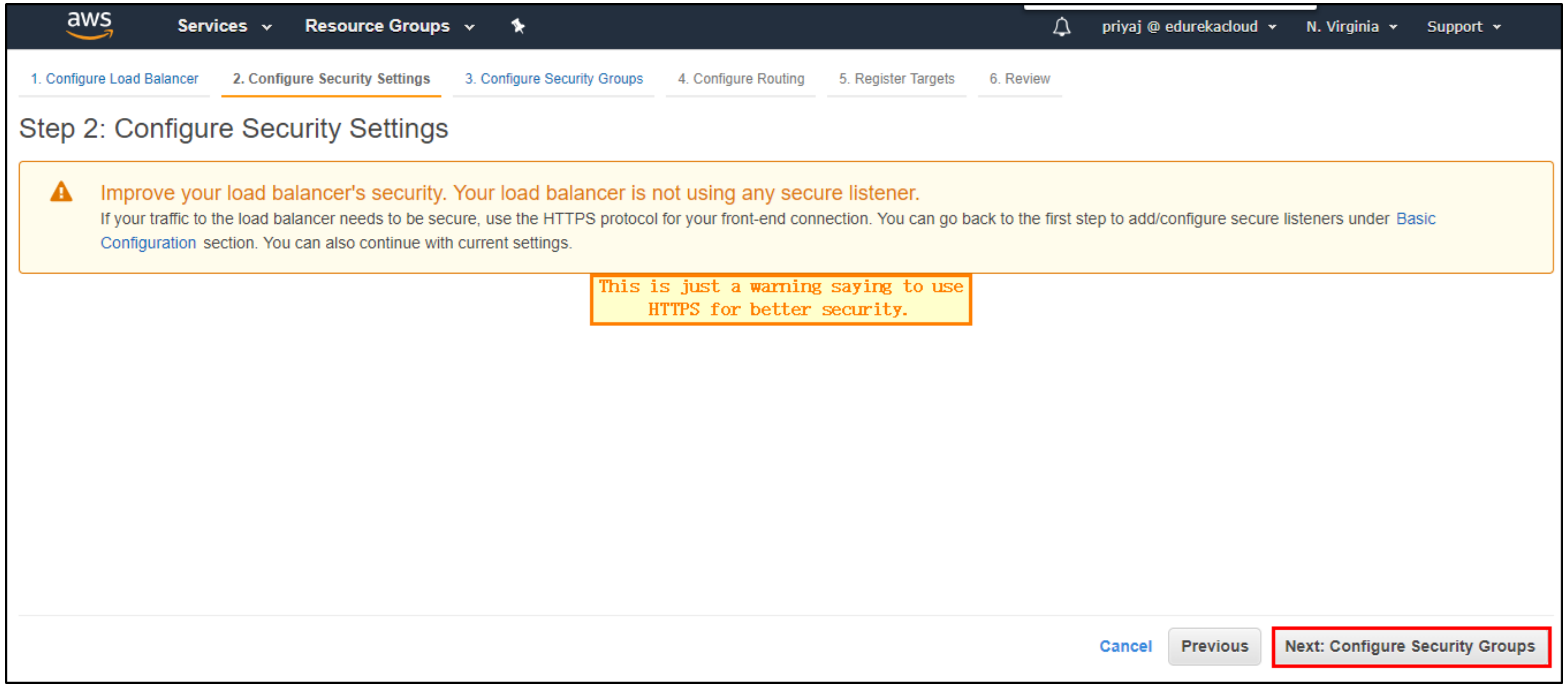
Tags

Apply tags to your load balancer to help organize and identify them.

Key	Value
Name	edurekademoelb

Adding Tag is important only when you have multiple resource running.

Step 5: Now you have to "*Configure Security Settings*". You'll be prompted a warning as displayed in the picture below. This warning is just to let you know that you should use HTTPS instead of HTTP.

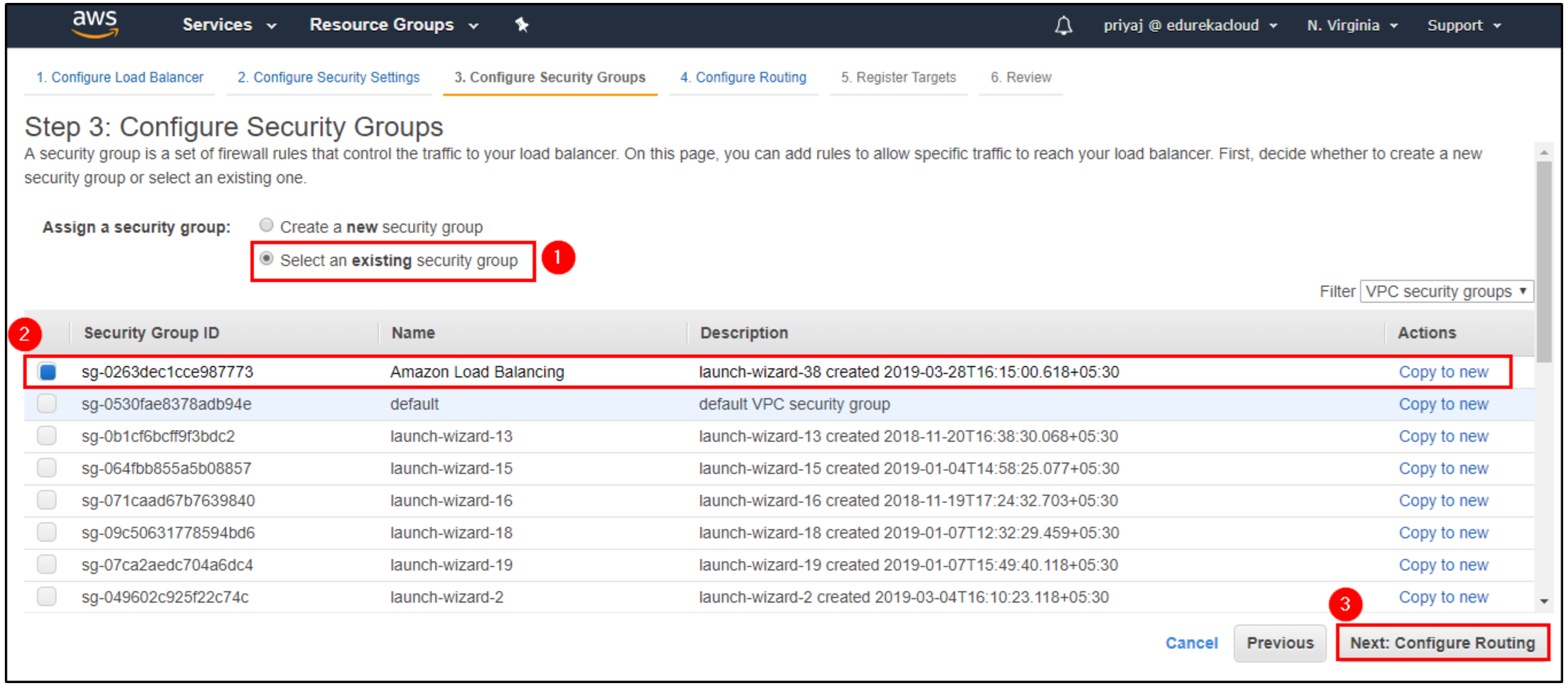


Step 2: Configure Security Settings

Improve your load balancer's security. Your load balancer is not using any secure listener.
If your traffic to the load balancer needs to be secure, use the HTTPS protocol for your front-end connection. You can go back to the first step to add/configure secure listeners under Basic Configuration section. You can also continue with current settings.

This is just a warning saying to use HTTPS for better security.

Step 6: To configure Security Groups, start by assigning a security group.



Step 3: Configure Security Groups

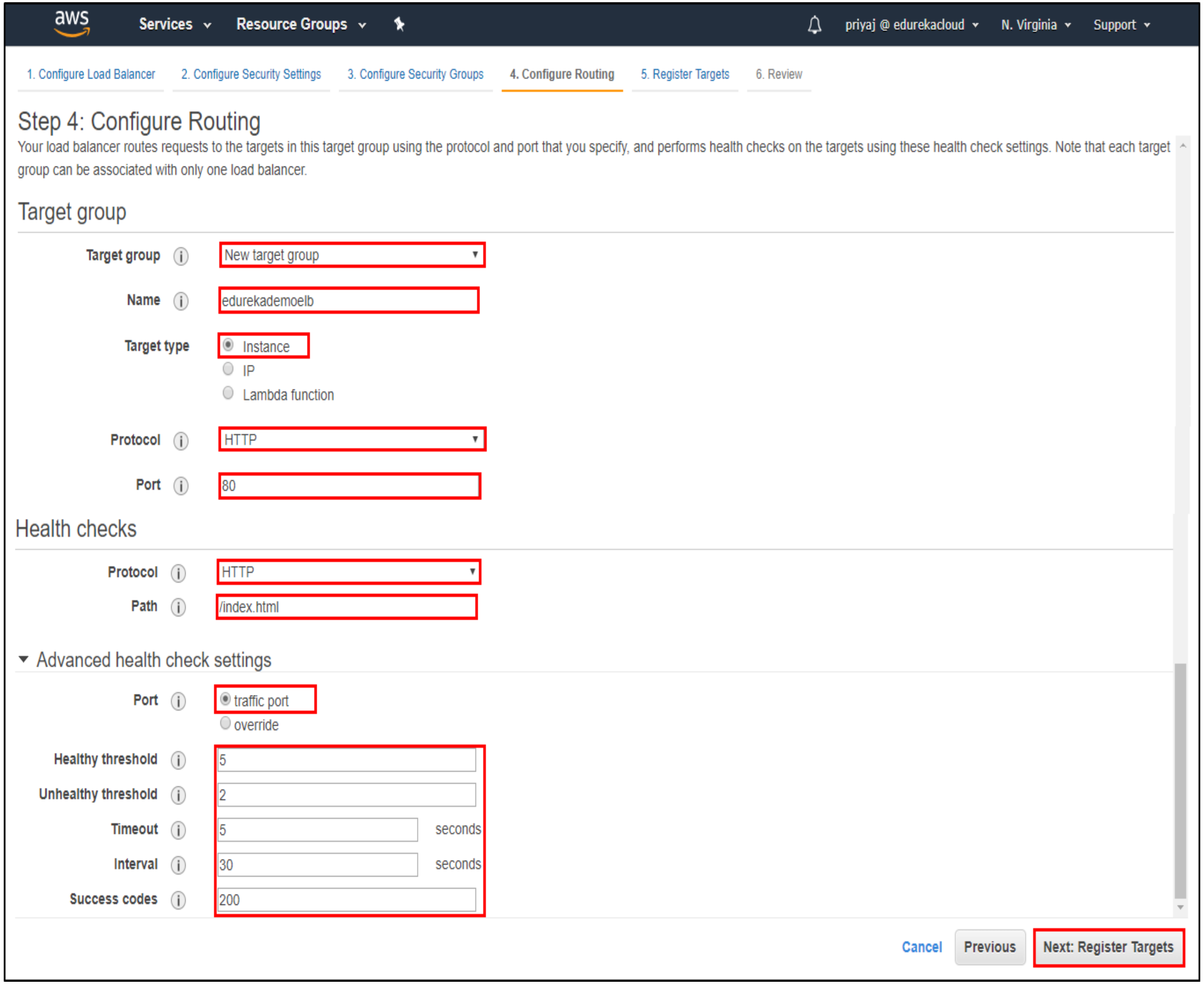
A security group is a set of firewall rules that control the traffic to your load balancer. On this page, you can add rules to allow specific traffic to reach your load balancer. First, decide whether to create a new security group or select an existing one.

Assign a security group: ○ Create a **new** security group
● Select an **existing** security group

Filter: VPC security groups

	Security Group ID	Name	Description	Actions
☑	sg-0263dec1cce987773	Amazon Load Balancing	launch-wizard-38 created 2019-03-28T16:15:00.618+05:30	Copy to new
□	sg-0530fae8378adb94e	default	default VPC security group	Copy to new
□	sg-0b1cf6bcff9f3bdc2	launch-wizard-13	launch-wizard-13 created 2018-11-20T16:38:30.068+05:30	Copy to new
□	sg-064fbb855a5b08857	launch-wizard-15	launch-wizard-15 created 2019-01-04T14:58:25.077+05:30	Copy to new
□	sg-071caad67b7639840	launch-wizard-16	launch-wizard-16 created 2018-11-19T17:24:32.703+05:30	Copy to new
□	sg-09c50631778594bd6	launch-wizard-18	launch-wizard-18 created 2019-01-07T12:32:29.459+05:30	Copy to new
□	sg-07ca2aedc704a6dc4	launch-wizard-19	launch-wizard-19 created 2019-01-07T15:49:40.118+05:30	Copy to new
□	sg-049602c925f22c74c	launch-wizard-2	launch-wizard-2 created 2019-03-04T16:10:23.118+05:30	Copy to new

Step 7: Now the main part, “Configure Routing“. Configure the routing as shown in the image.



Step 4: Configure Routing

Your load balancer routes requests to the targets in this target group using the protocol and port that you specify, and performs health checks on the targets using these health check settings. Note that each target group can be associated with only one load balancer.

Target group

Target group: New target group

Name: edurekademoelb

Target type: ● Instance ○ IP ○ Lambda function

Protocol: HTTP

Port: 80

Health checks

Protocol: HTTP

Path: /index.html

Advanced health check settings

Port: ● traffic port ○ override

Healthy threshold: 5

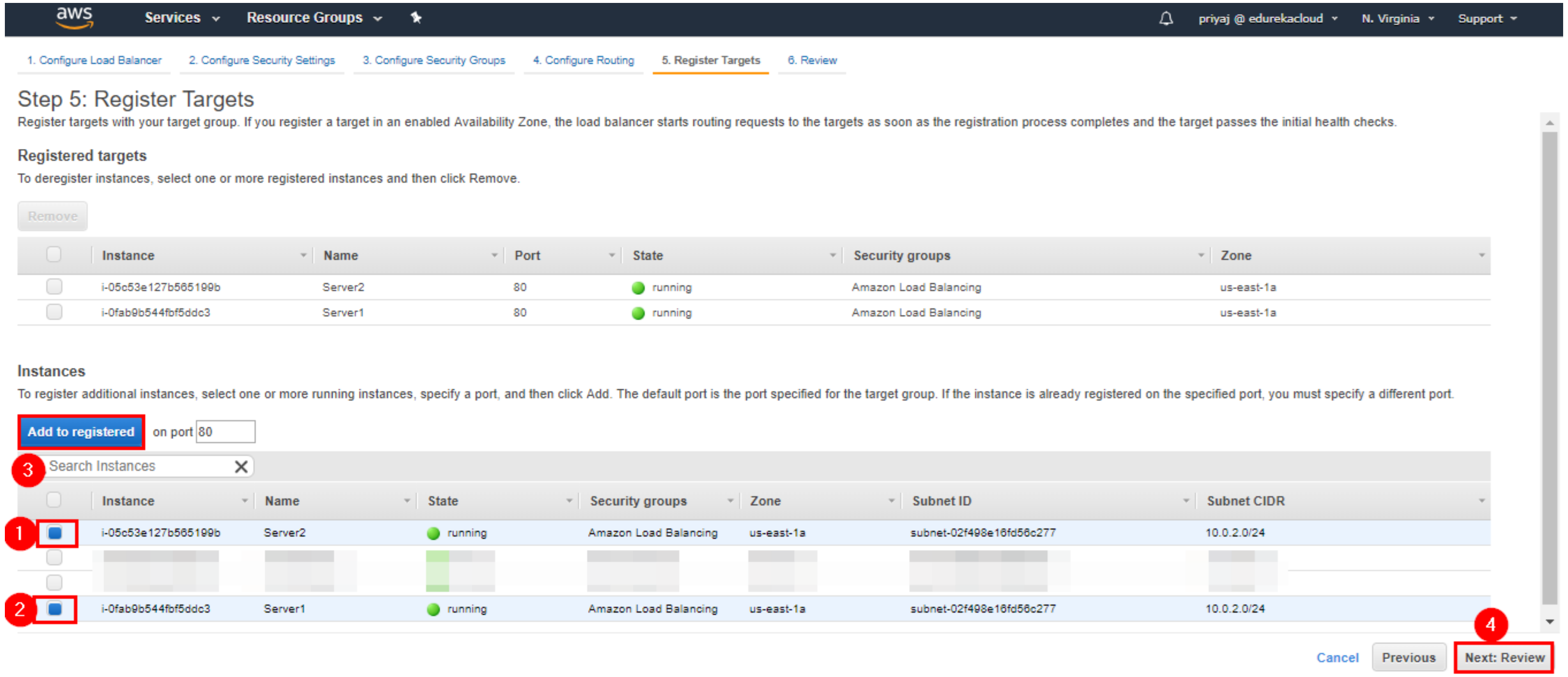
Unhealthy threshold: 2

Timeout: 5 seconds

Interval: 30 seconds

Success codes: 200

Step 8: Now, just add your instance, i.e. “Register Targets“..



Step 5: Register Targets

Register targets with your target group. If you register a target in an enabled Availability Zone, the load balancer starts routing requests to the targets as soon as the registration process completes and the target passes the initial health checks.

Registered targets

To deregister instances, select one or more registered instances and then click Remove.

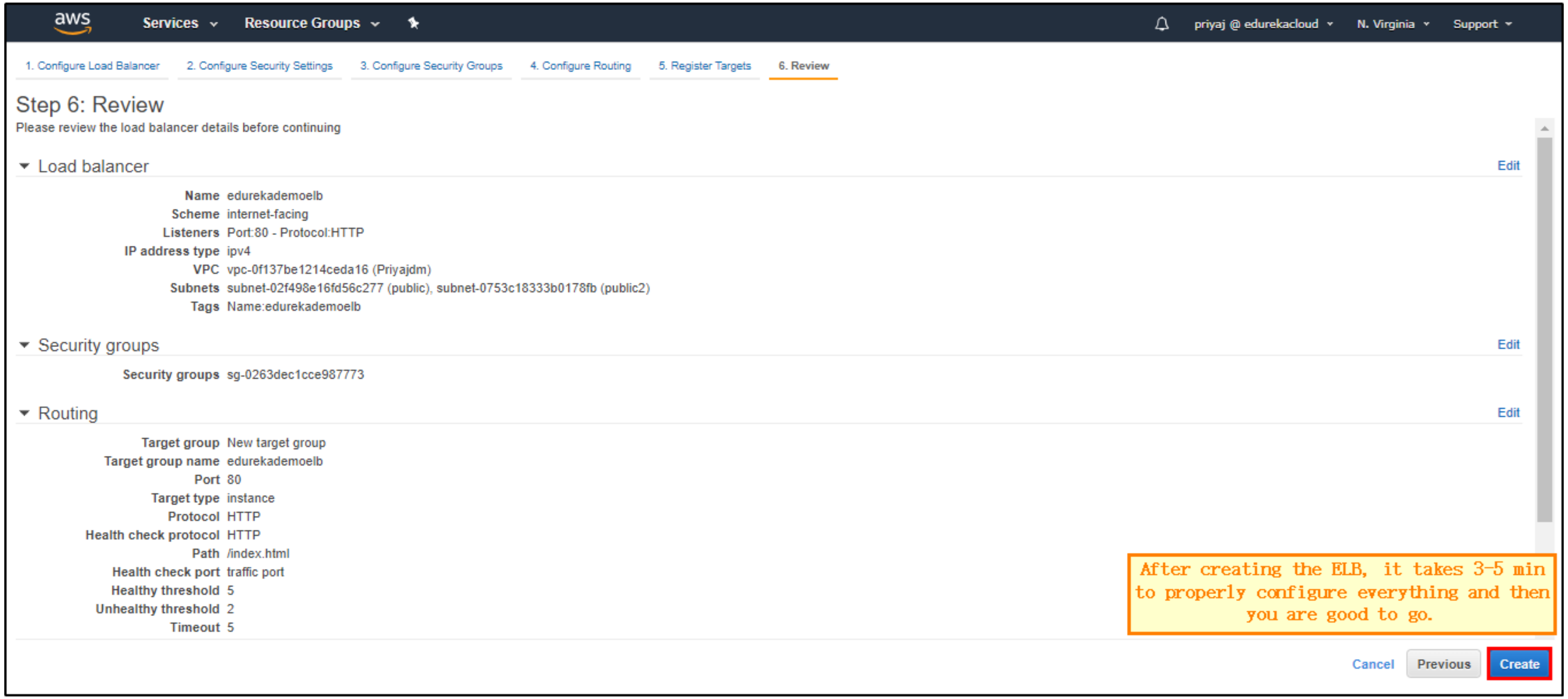
□	Instance	Name	Port	State	Security groups	Zone
□	i-05c53e127b565199b	Server2	80	● running	Amazon Load Balancing	us-east-1a
□	i-0fab9b544fbf5ddc3	Server1	80	● running	Amazon Load Balancing	us-east-1a

Instances

To register additional instances, select one or more running instances, specify a port, and then click Add. The default port is the port specified for the target group. If the instance is already registered on the specified port, you must specify a different port.

□	Instance	Name	State	Security groups	Zone	Subnet ID	Subnet CIDR
1 ☑	i-05c53e127b565199b	Server2	● running	Amazon Load Balancing	us-east-1a	subnet-02f498e16fd56c277	10.0.2.0/24
□	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
□	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
2 ☑	i-0fab9b544fbf5ddc3	Server1	● running	Amazon Load Balancing	us-east-1a	subnet-02f498e16fd56c277	10.0.2.0/24

Step 9: The final step "Review" all the settings if they're fine and Now, wait for 3-5 mins for the load balancer to configure.



Step 6: Review

Please review the load balancer details before continuing

Load balancer

Name	edurekademoelb
Scheme	internet-facing
Listeners	Port:80 - Protocol:HTTP
IP address type	ipv4
VPC	vpc-0f137be1214ceda16 (Priyajdm)
Subnets	subnet-02f498e16fd56c277 (public), subnet-0753c18333b0178fb (public2)
Tags	Name:edurekademoelb

Security groups

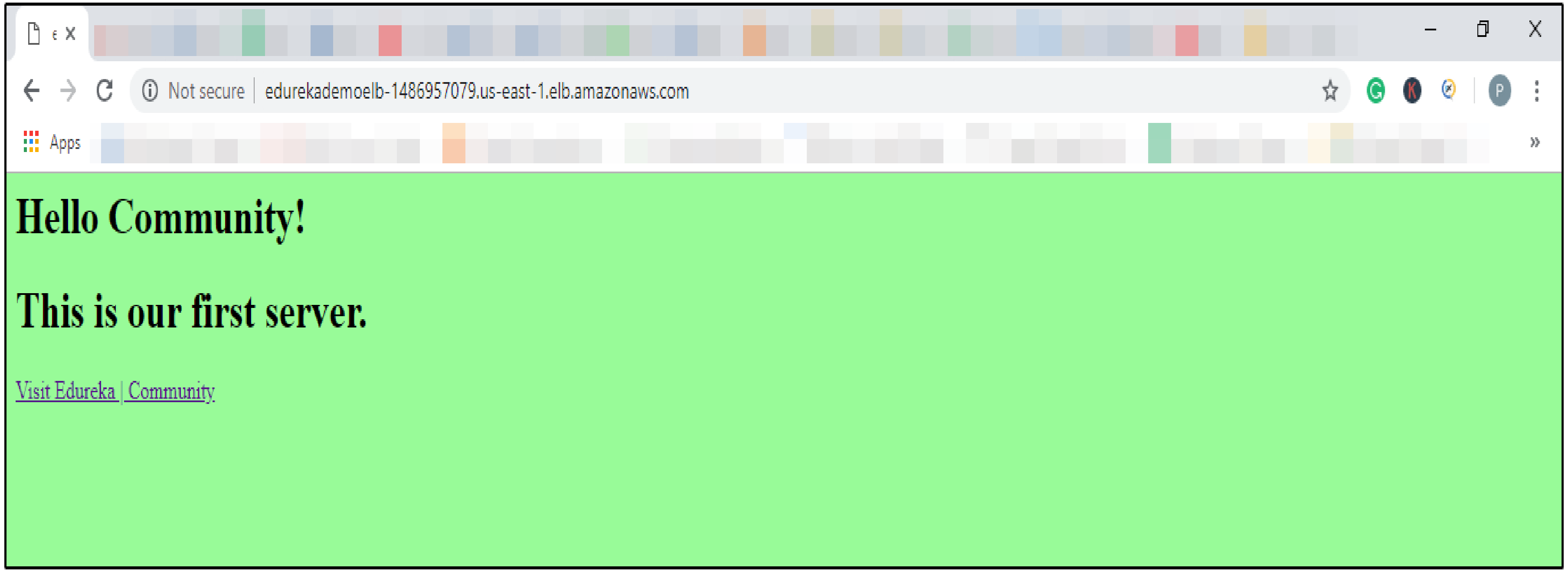
Security groups	sg-0263dec1cce987773

Routing

Target group	New target group
Target group name	edurekademoelb
Port	80
Target type	instance
Protocol	HTTP
Health check protocol	HTTP
Path	/index.html
Health check port	traffic port
Healthy threshold	5
Unhealthy threshold	2
Timeout	5

After creating the ELB, it takes 3-5 min to properly configure everything and then you are good to go.

Step 10: Select the DNS from the description. Paste it as a URL in your browser. It says “This is our First server”, indicating the web page got deployed on the first server. Hit “*Refresh*”

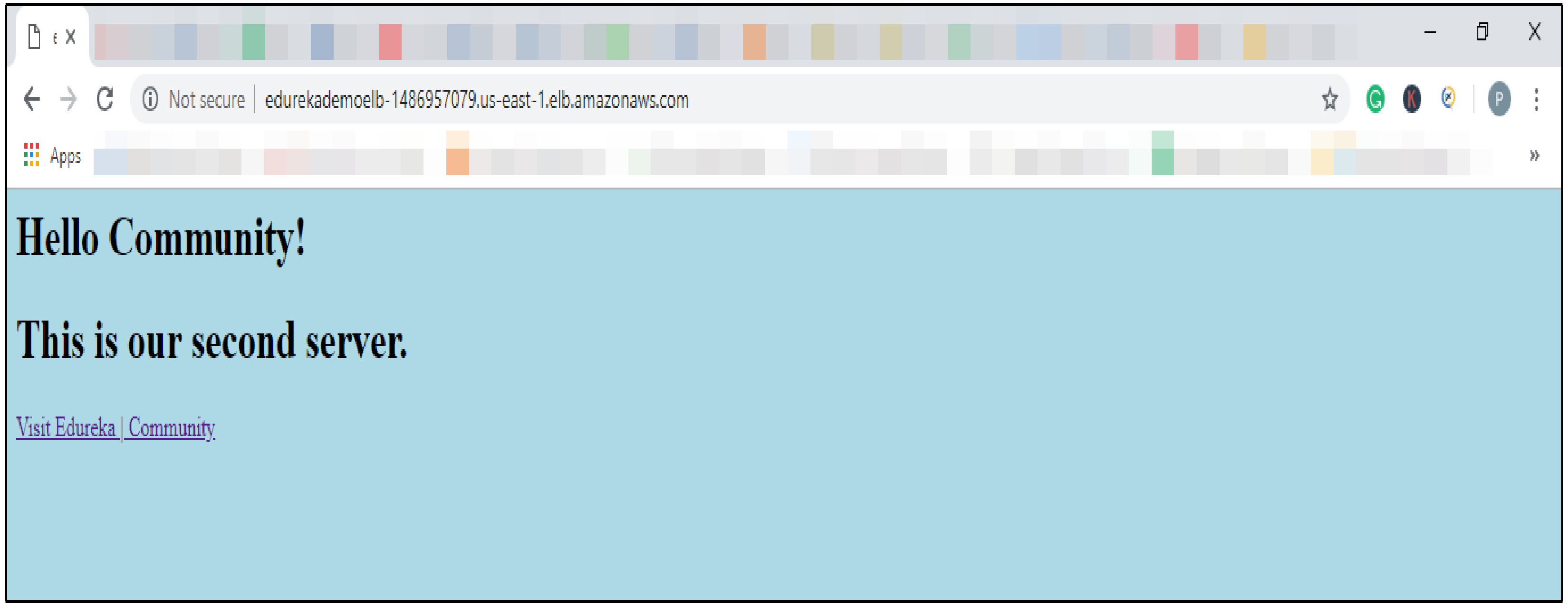


Hello Community!

This is our first server.

Visit Edureka | Community

- You’ll find another page(shown in the image below) being displayed which shows that the web page got deployed on the second server.
- You can see how Load balancer diverts the traffic to different servers to service the request from users.



Hello Community!

This is our second server.

Visit Edureka | Community

Conclusion

We learned

- Creation of an account on AWS.
- How to deploy HTML content on Amazon Linux 2 instance.
- About different types of AMI, Instance types, networking settings of EC2, storage, and user data.
- Deploying a website for admission portal on the EC2 Service.
- Configuring the Traffic rules of the Server for a specific need.
- Creating of Application Load Balancer

THANK

YOU FOR

LISTENING

ANY

QUESTION ?