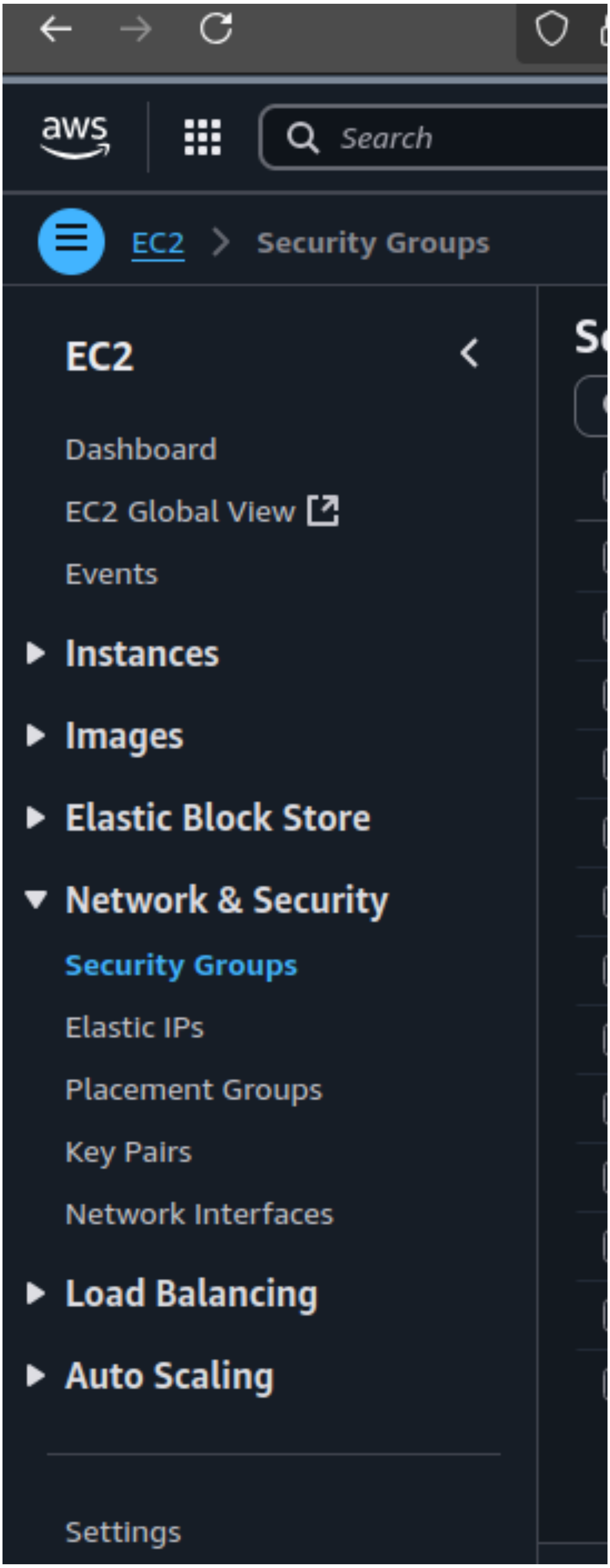
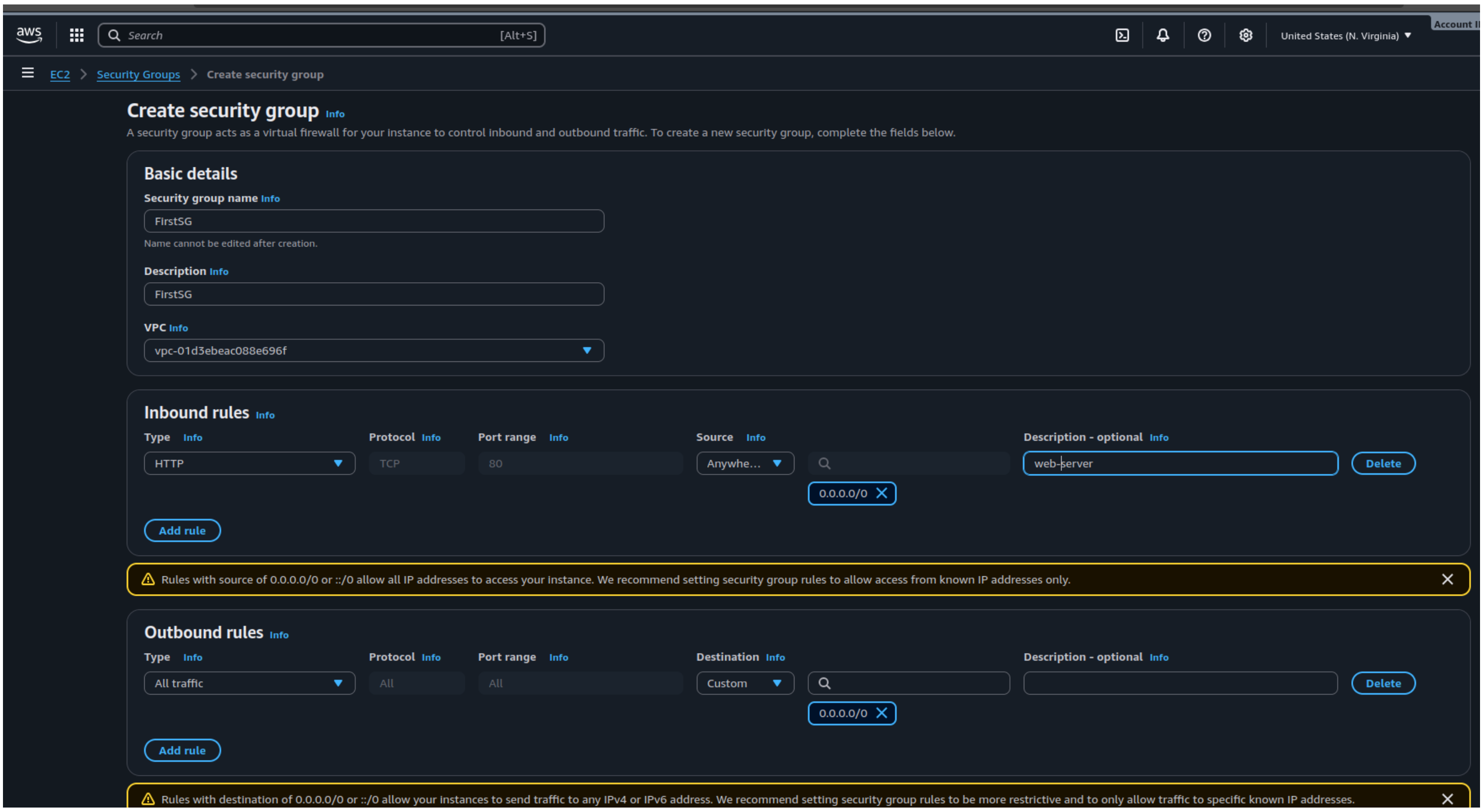
Creating a basic EC2 instance

Creating a Security group:
Under EC2, go to network and security tab and choos security group.



Click on the orange tab to create a security group.
Then name it something in relation to what the project is.
I use the name for the description also.
Inbound rules is http with source of Anywhere ipv4.



EC2 > Security Groups > Create security group

Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name Info: FirstSG

Name cannot be edited after creation.

Description Info: FirstSG

VPC Info: vpc-01d3ebeac088e696f

Inbound rules Info

Type	Protocol	Port range	Source		Description - optional
HTTP	TCP	80	Anywhe...	0.0.0.0/0	web-server

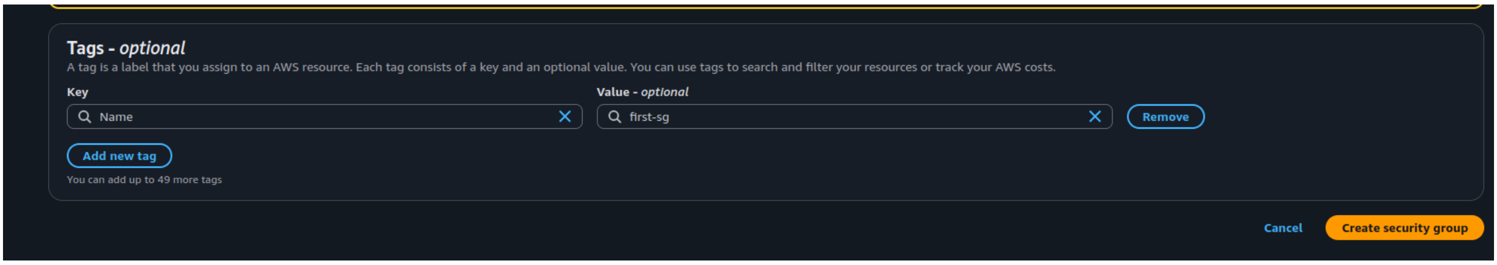
Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Outbound rules Info

Type	Protocol	Port range	Destination		Description - optional
All traffic	All	All	Custom	0.0.0.0/0	

Rules with destination of 0.0.0.0/0 or ::/0 allow your instances to send traffic to any IPv4 or IPv6 address. We recommend setting security group rules to be more restrictive and to only allow traffic to specific known IP addresses.

I add a tag and then press the orang tab to create the security group.



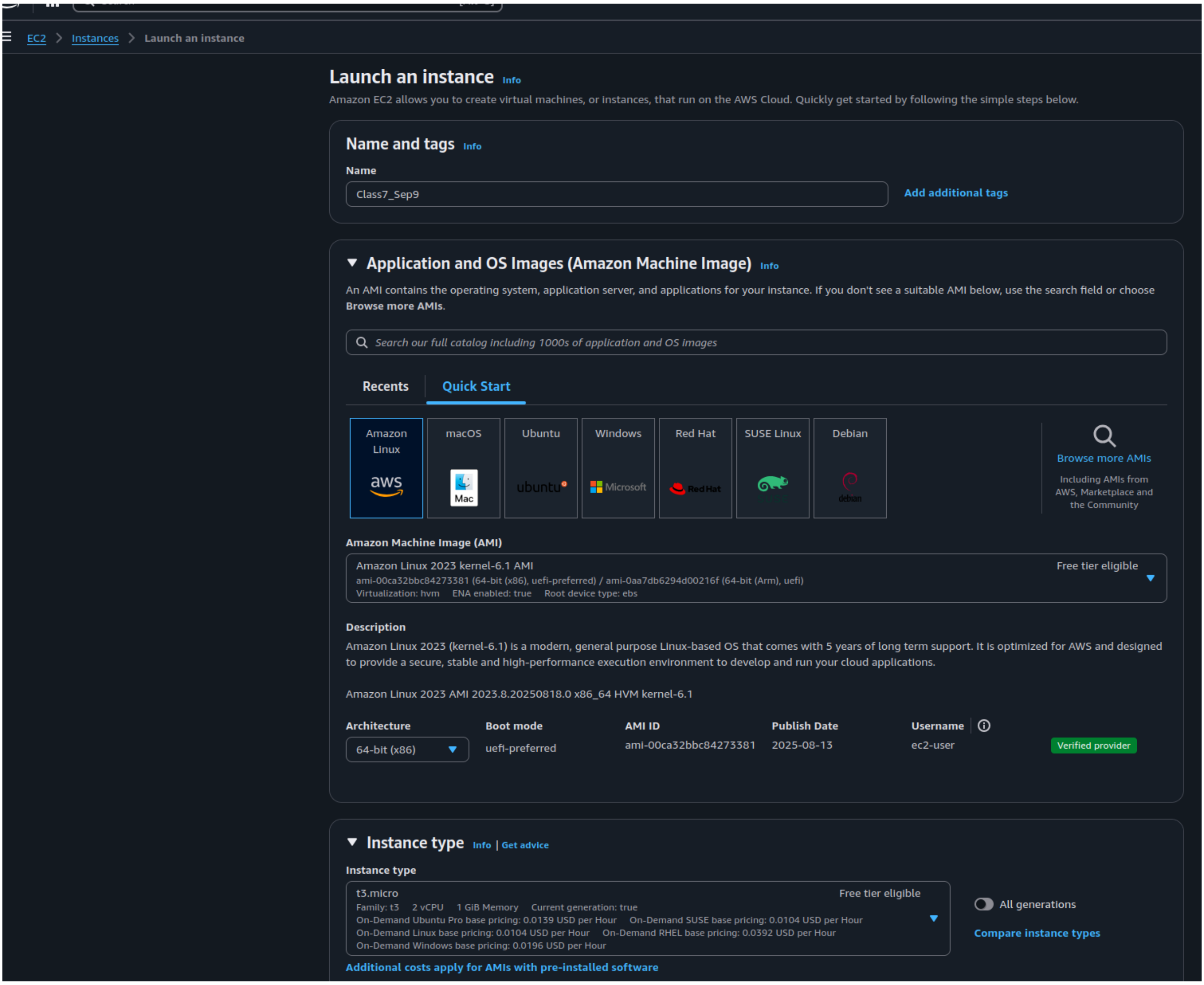
Tags - *optional*

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*
Name	first-sg

You can add up to 49 more tags

Under Instances in EC2, click on launch instance.
Give your instance a name. I named mine Class7_Sep9.
Leave the Image on default Amazon Linux and default Instance type.



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: Class7_Sep9

▼ Application and OS Images (Amazon Machine Image) Info

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose **Browse more AMIs**.

Search our full catalog including 1000s of application and OS images
Recents
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE Linux
Debian
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI — Free tier eligible
ami-00ca32bbc84273381 (64-bit (x86), uefi-preferred) / ami-0aa7db6294d00216f (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2023 (kernel-6.1) is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

Amazon Linux 2023 AMI 2023.8.20250818.0 x86_64 HVM kernel-6.1

Architecture	Boot mode	AMI ID	Publish Date	Username	
64-bit (x86)	uefi-preferred	ami-00ca32bbc84273381	2025-08-13	ec2-user	Verified provider

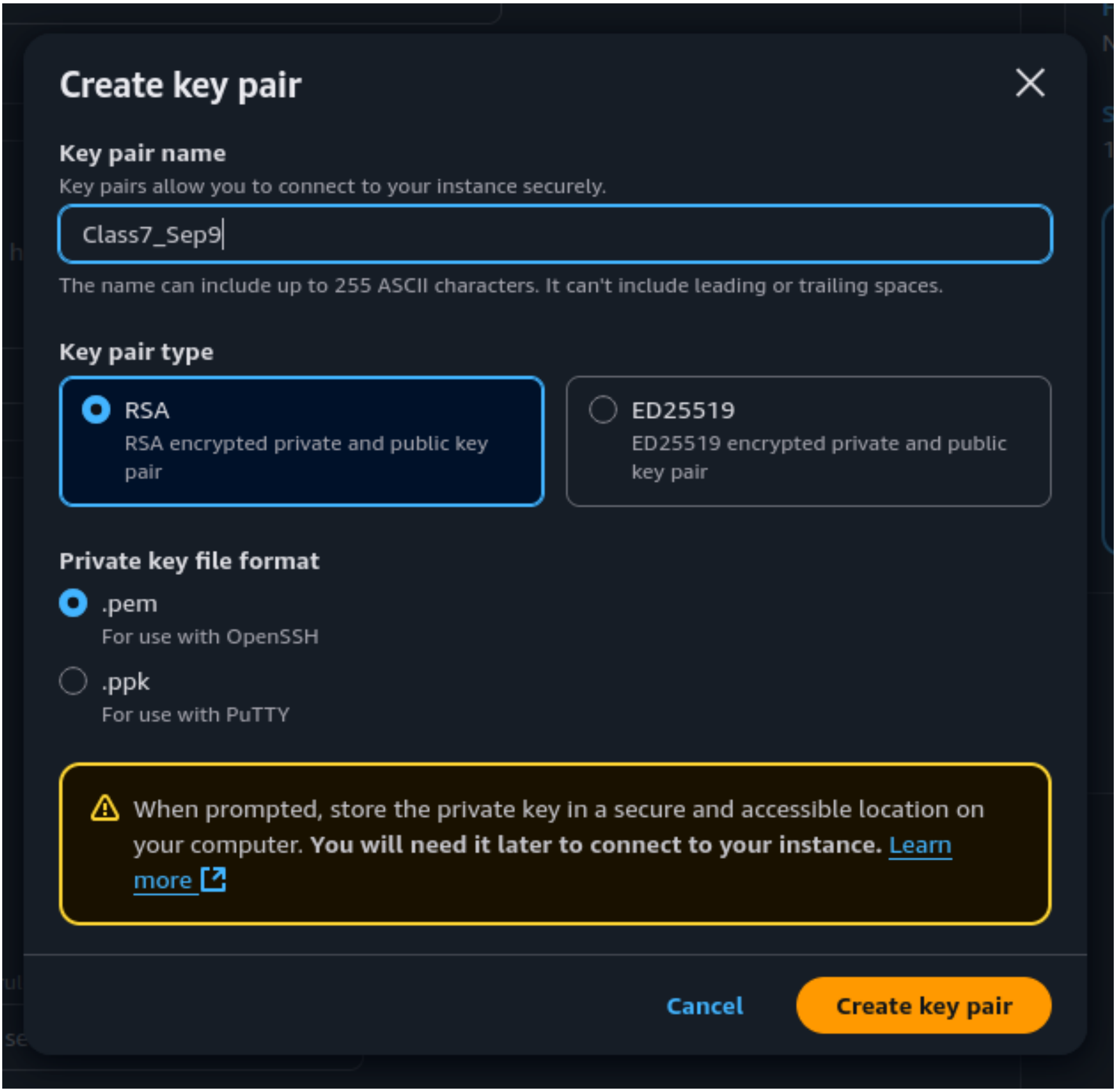
▼ Instance type Info | Get advice

Instance type

t3.micro — Free tier eligible
Family: t3 2 vCPU 1 GiB Memory Current generation: true
On-Demand Ubuntu Pro base pricing: 0.0139 USD per Hour On-Demand SUSE base pricing: 0.0104 USD per Hour
On-Demand Linux base pricing: 0.0104 USD per Hour On-Demand RHEL base pricing: 0.0392 USD per Hour
On-Demand Windows base pricing: 0.0196 USD per Hour

○ All generations

Under Key pair if this is your first time create a new key pair.



Create key pair

Key pair name

Key pairs allow you to connect to your instance securely.

Class7_Sep9

The name can include up to 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

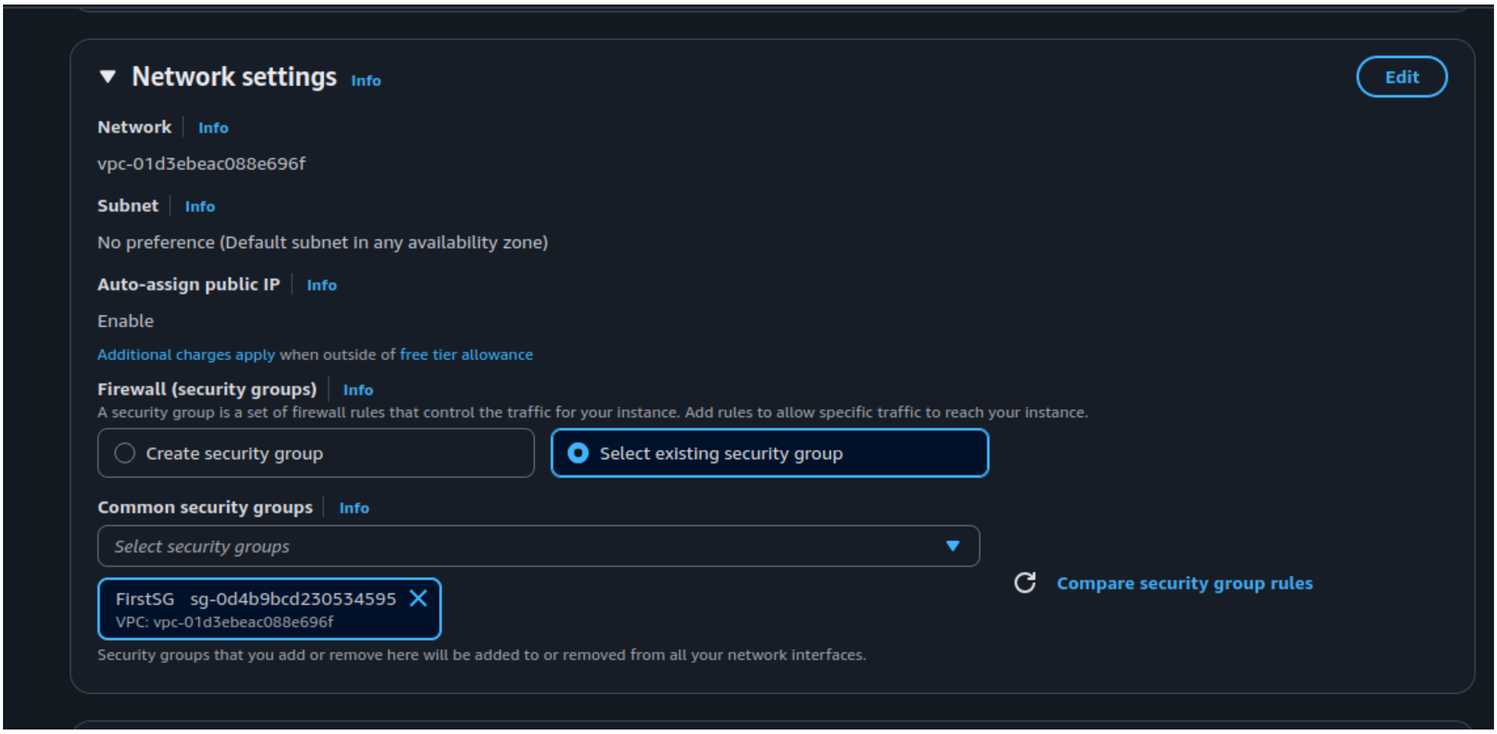
● .pem
For use with OpenSSH

○ .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

I used the same naming convention to keep it simple.

In the Network Settings we will choose a security group that we previously created before launching the EC2 instance.



▼ Network settings Info

Network | Info

vpc-01d3ebeac088e696f

Subnet | Info

No preference (Default subnet in any availability zone)

Auto-assign public IP | Info

Enable

Additional charges apply when outside of free tier allowance

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

Common security groups | Info

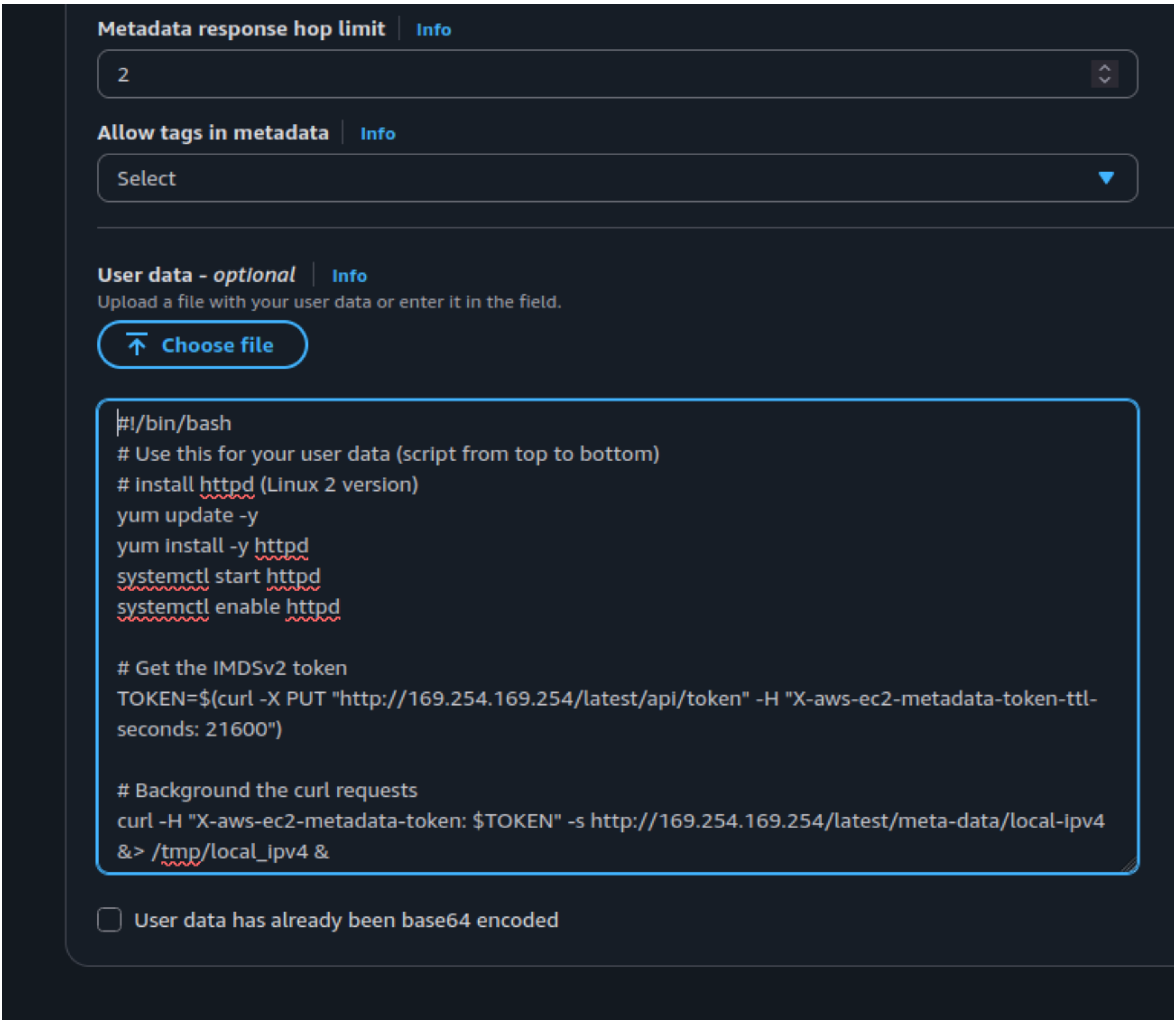
Select security groups ▼

FirstSG sg-0d4b9bcd230534595 ✕
VPC: vpc-01d3ebeac088e696f

Security groups that you add or remove here will be added to or removed from all your network interfaces.

I named it the same as the instance with SG at the end. I also added a inbound rule for http as the type, source is anywhere ipv4.

I then go into the advanced details and paste the startup.txt script into the user data before launching the instance for the first time.



Metadata response hop limit | Info
2
Allow tags in metadata | Info
Select
User data - *optional* | Info
Upload a file with your user data or enter it in the field.
Choose file

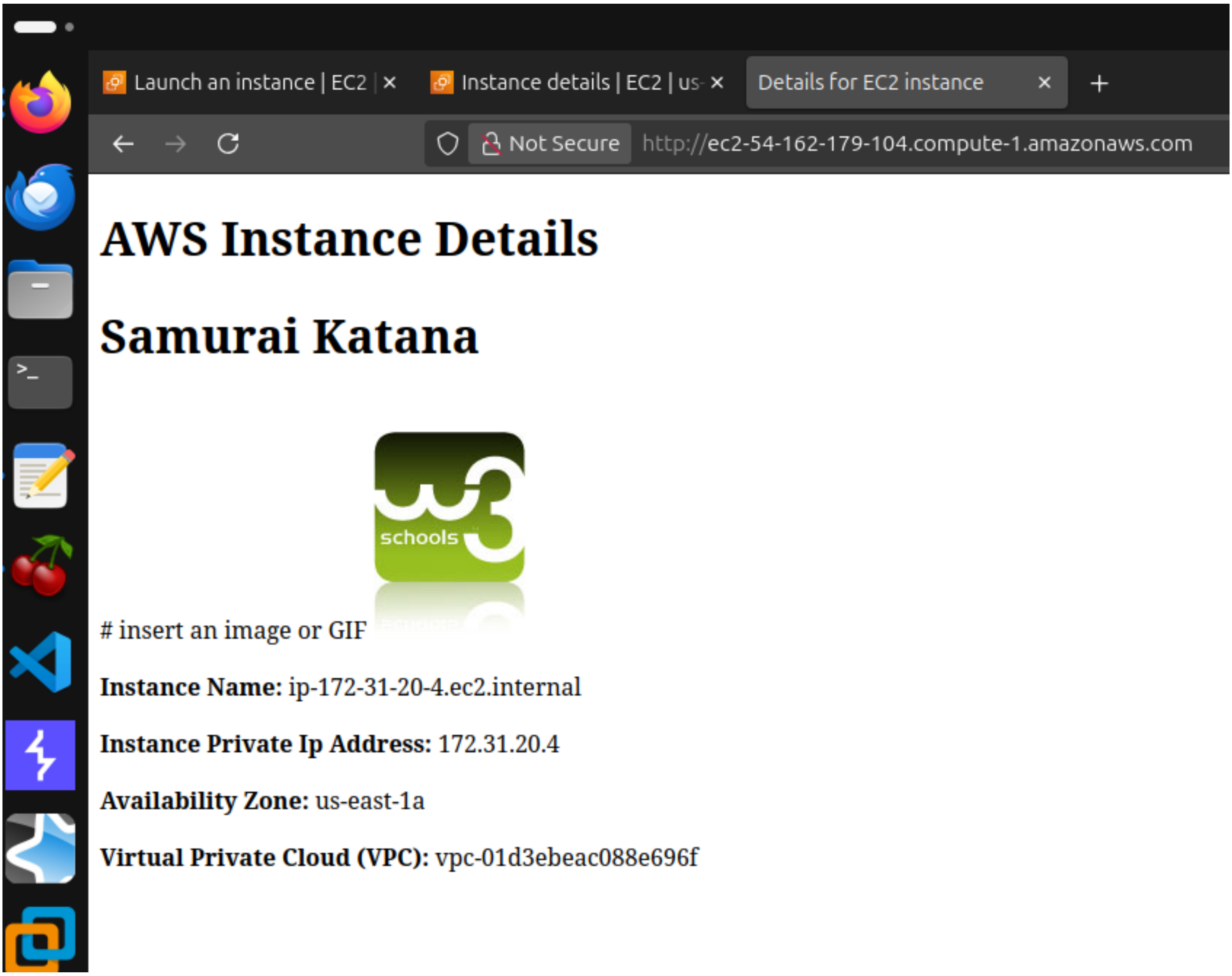
```
#!/bin/bash
# Use this for your user data (script from top to bottom)
# install httpd (Linux 2 version)
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd

# Get the IMDSv2 token
TOKEN=$(curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600")

# Background the curl requests
curl -H "X-aws-ec2-metadata-token: $TOKEN" -s http://169.254.169.254/latest/meta-data/local-ipv4 &> /tmp/local_ipv4 &
```

□ User data has already been base64 encoded

Now i launch the instance
After the instance launches i verify by going to the public dns on the browser.
http://


Launch an instance | EC2 |
Instance details | EC2 | us-
Details for EC2 instance
Not Secure
http://ec2-54-162-179-104.compute-1.amazonaws.com
schools

AWS Instance Details

Samurai Katana

insert an image or GIF

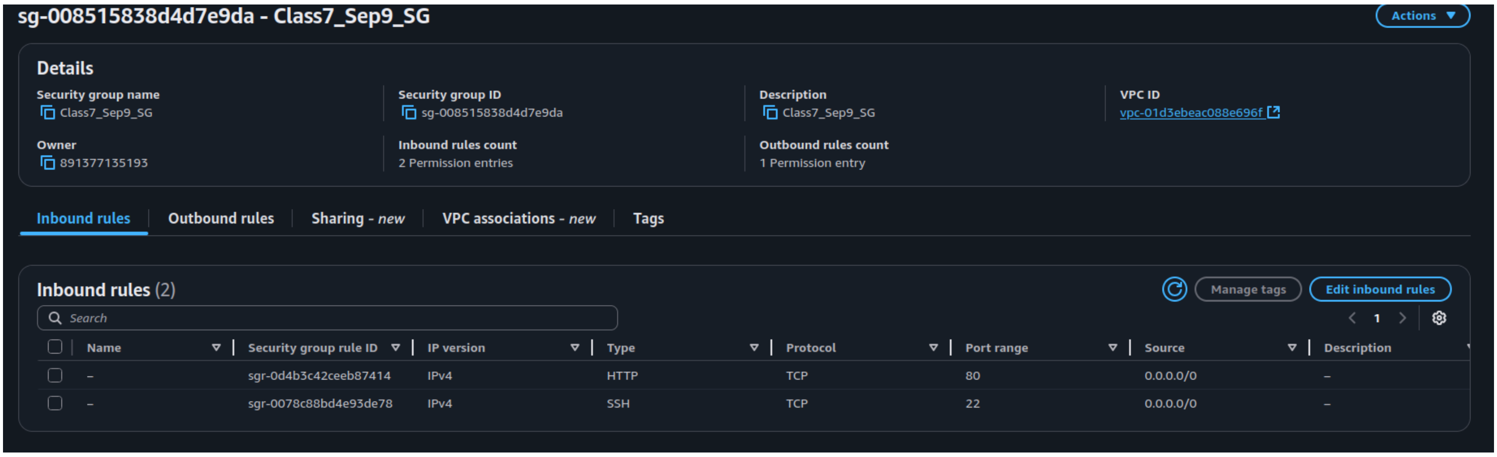
Instance Name: ip-172-31-20-4.ec2.internal

Instance Private Ip Address: 172.31.20.4

Availability Zone: us-east-1a

Virtual Private Cloud (VPC): vpc-01d3ebeac088e696f

In order to do the second part of the homework, i enabled ssh on the security group so that i can ssh into the VM.



sg-008515838d4d7e9da - Class7_Sep9_SG

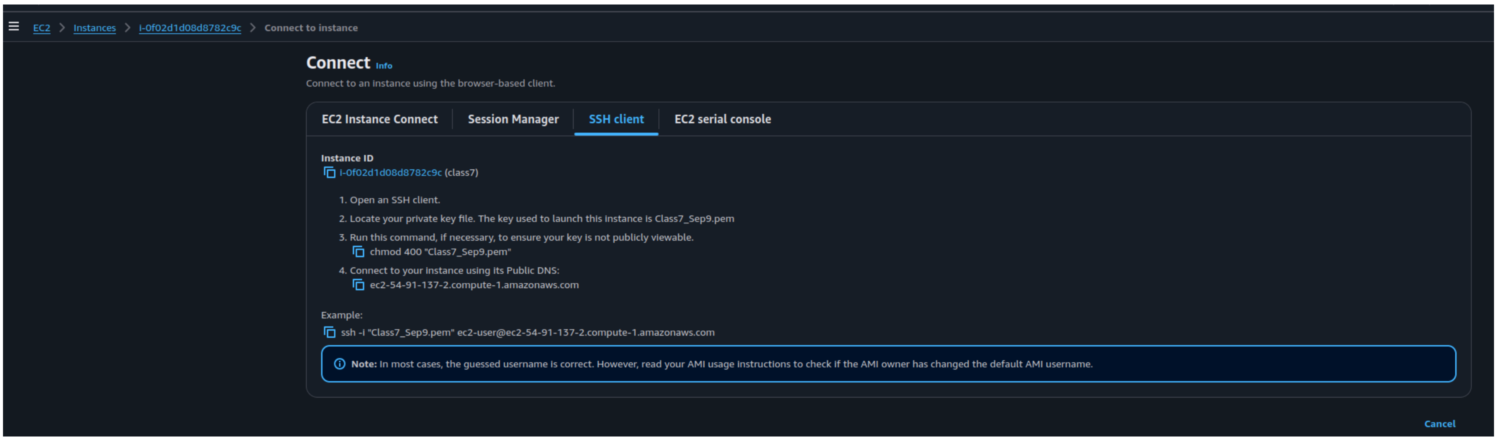
Details

Security group name: Class7_Sep9_SG
Security group ID: sg-008515838d4d7e9da
Description: Class7_Sep9_SG
VPC ID: vpc-01d3ebeac088e696f
Owner: 891377135193
Inbound rules count: 2 Permission entries
Outbound rules count: 1 Permission entry

Inbound rules (2)

□	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source	Description
□	–	sgr-0d4b3c42ceeb87414	IPv4	HTTP	TCP	80	0.0.0.0/0	–
□	–	sgr-0078c88bd4e93de78	IPv4	SSH	TCP	22	0.0.0.0/0	–

After enabling ssh i connected to the instance according to AWs instructions:



Connect to an instance using the browser-based client.

Instance ID
i-0f02d1d08d8782c9c (class7)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is Class7_Sep9.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
   `chmod 400 "Class7_Sep9.pem"`
4. Connect to your instance using its Public DNS:
   `ec2-54-91-137-2.compute-1.amazonaws.com`

Example:

```
ssh -i "Class7_Sep9.pem" ec2-user@ec2-54-91-137-2.compute-1.amazonaws.com
```

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

So i first change the permissions of the .pem that i recieved that is currently on my computer with the command:

```
chmod 400 “Class7_Sep9.pem”
```

I then ssh into the VM:

```
ssh -i "Class7_Sep9.pem" ec2-user@ec2-54-91-137-2.compute-1.amazonaws.com
```

```
[ec2-user@ip-172-31-23-54 html]$ 
```

Now that i am into the VM i change to the directory that the webserver file is in.

```
cd /var/www/html/
```

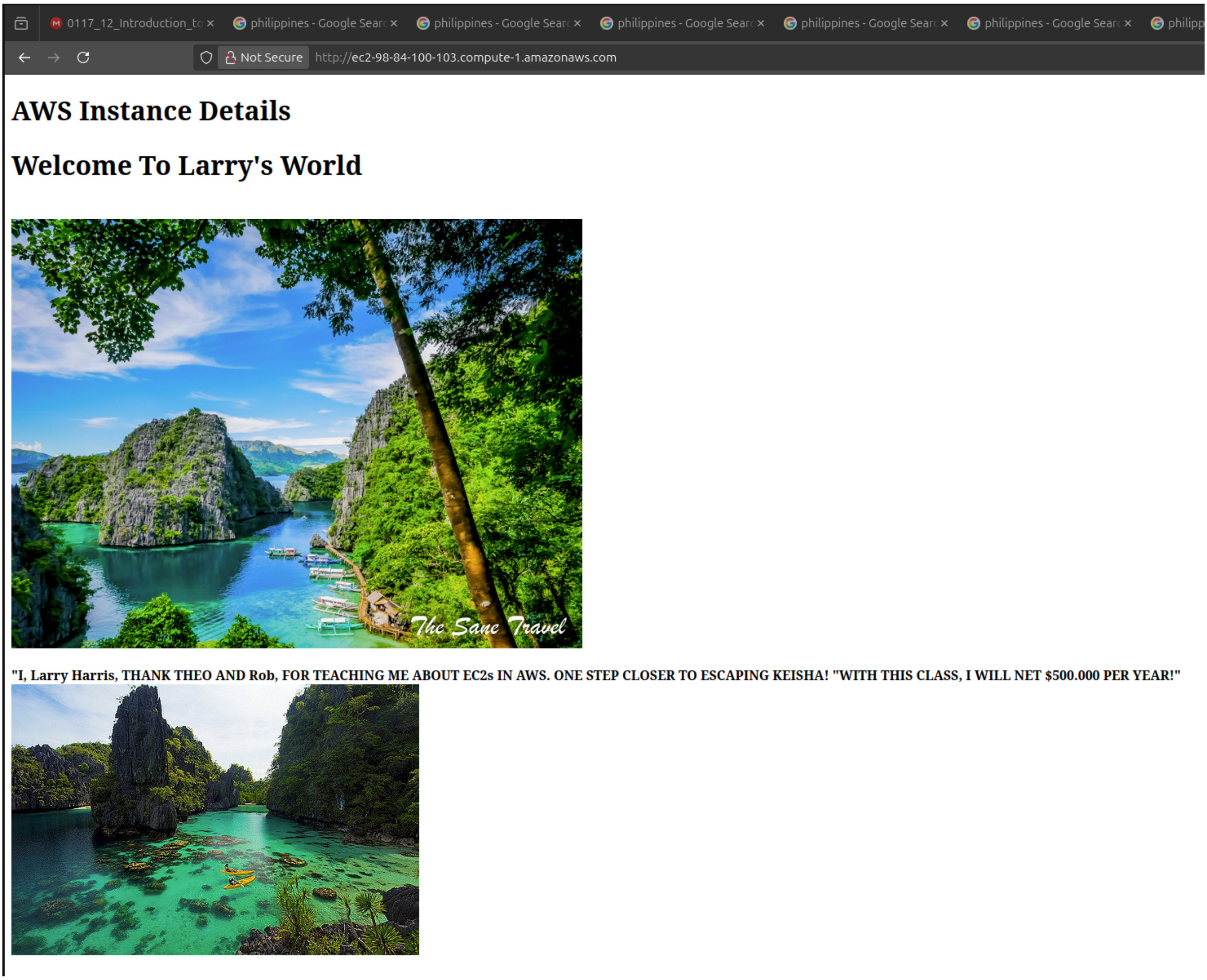
Then:

```
sudo nano index.html
```

Paste the following into the editor:

```
<html lang=\"en\" class=\"h-100\">
<head>
<title>Details for EC2 instance</title>
</head>
<body>
<div>
<h1>WINNERS WIN!!!</h1>
<br>
<img src="https://www.nowtravelasia.com/wp-content/uploads/2018/08/trip-asia-travel.jpg"
<br>
<p><b>"I, Larry Harris, THANK THEO AND Rob, FOR TEACHING ME ABOUT EC2s IN AWS. ONE STEP CLOSER TO ESCAPING KEISHA! "WITH THIS CLASS, I WILL NET $500.000 PER YEAR!"</b</p>
</div>
</body>
</html>
```

save it and then refresh the website to see the update.


0117_12_Introduction_to
philippines - Google Searc
philippines - Google Searc
philippines - Google Searc
philippines - Google Searc
philippines - Google Searc
philipp
Not Secure
http://ec2-98-84-100-103.compute-1.amazonaws.com
The Sane Travel

AWS Instance Details

Welcome To Larry's World

"I, Larry Harris, THANK THEO AND Rob, FOR TEACHING ME ABOUT EC2s IN AWS. ONE STEP CLOSER TO ESCAPING KEISHA! "WITH THIS CLASS, I WILL NET $500.000 PER YEAR!"