Deploy Linux Server Using AWS

Greg Hamelin

AWS – Amazon Web Services

1. My choice - Amazon AWS.

2. I chose Amazon AWS because it is so popular and their free tier is so generous that it is not only a great choice for my resume, but also a great choice because I would likely use this for some of my own personal projects.

3. Amazon AWS started out as a way to help solve some of the problems Amazon's software engineers were running into. Software teams were spending large amounts of time building out infrastructure. Multiple teams were faced with building infrastructure to support what they were working on and all that time and disparate efforts were inefficient, expensive and time consuming. Amazon decided to create their own internal cloud which software teams could share to build their projects and not have to deal with the difficult task of building and maintaining infrastructure. There is a great article on Fortune magazine's site where they interviewed Adam Selipsky who is the CEO for AWS. Amazon Web Services has grown much since it started and accounts for a large majority of internet infrastructure as well as a majority share of Amazon's revenue.

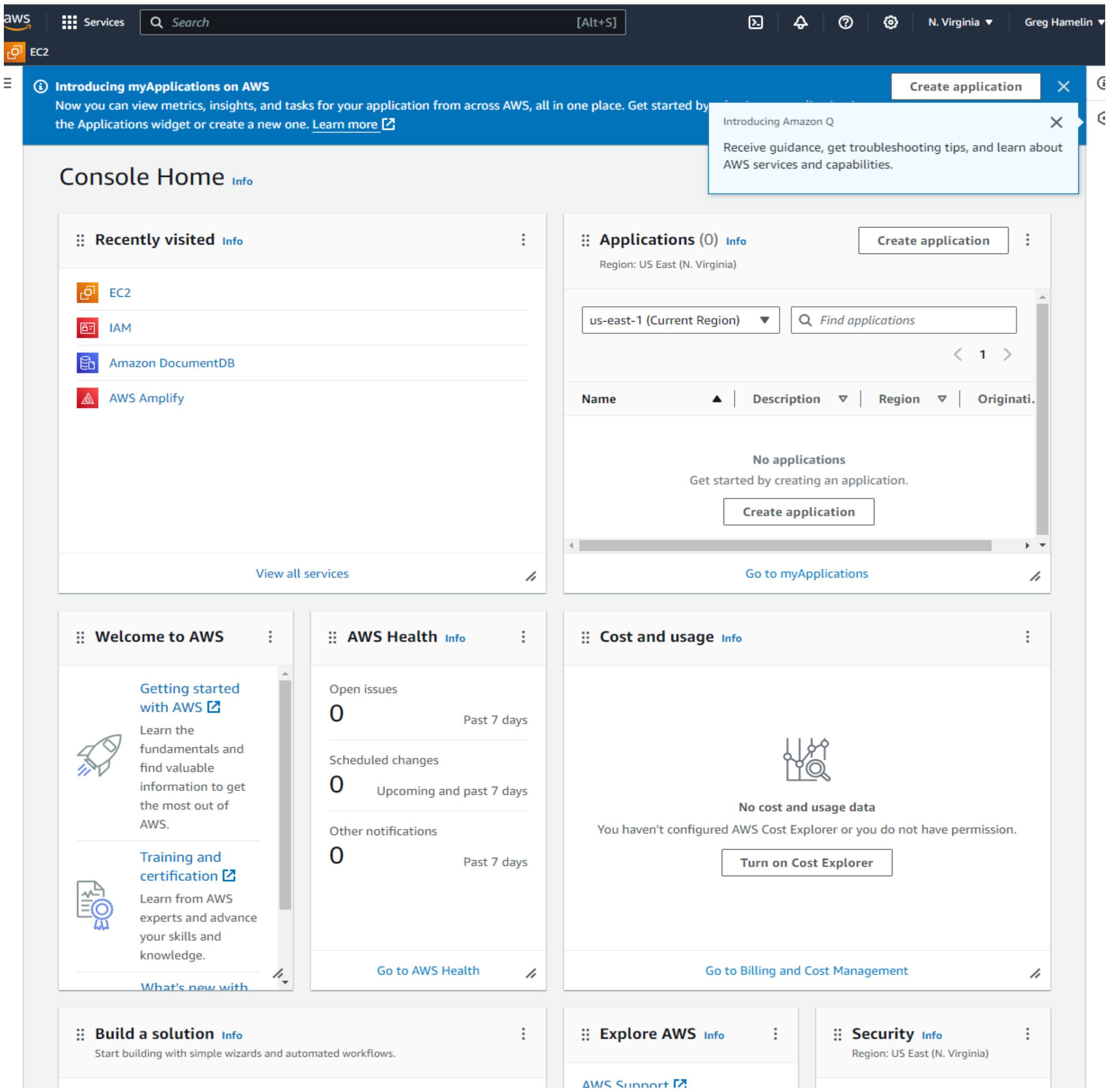
4. I already had experience using AWS from a class I took this semester so it was a no brainer and I thought it was a great way to test my familiarity with the AWS console.

5. It did work the way I expected, but if I had to learn everything from scratch I think it could have been much more daunting and time consuming. I don't normally use cloud providers so setting up a Linux VM and connecting to it on my local machine is very simple. With AWS I have to be aware of not just the basics like OS type, resources for the VM but also know how to connect to it via SSH using the public IP and a PPK file to handle authentication.

6. Challenges were making sure I took my time and captured each step in my instructions as well as screenshots. I wanted my guide to be nearly fool proof even if the person following my directions was not tech savvy.

Step 1. Access Amazon Web Services

- Log in to AWS – If you have an Amazon account already you can use that to setup a free AWS account which has a very generous free tier offering.
    - We will need to setup an EC2 instance.
    - EC2 is Amazon's Elastic Compute Cloud.
    - This is the part of AWS where you would create a virtual machine.
    - AWS calls the virtual machine an "instance"



Introducing myApplications on AWS
Now you can view metrics, insights, and tasks for your application from across AWS, all in one place. Get started by the Applications widget or create a new one. Learn more

Introducing Amazon Q
Receive guidance, get troubleshooting tips, and learn about AWS services and capabilities.

Console Home Info

Recently visited: EC2, IAM, Amazon DocumentDB, AWS Amplify

Applications (0) — Region: US East (N. Virginia)
No applications
Get started by creating an application.

Getting started with AWS
Learn the fundamentals and find valuable information to get the most out of AWS.

Training and certification
Learn from AWS experts and advance your skills and knowledge.

AWS Health
Open issues 0 Past 7 days
Scheduled changes 0 Upcoming and past 7 days
Other notifications 0 Past 7 days

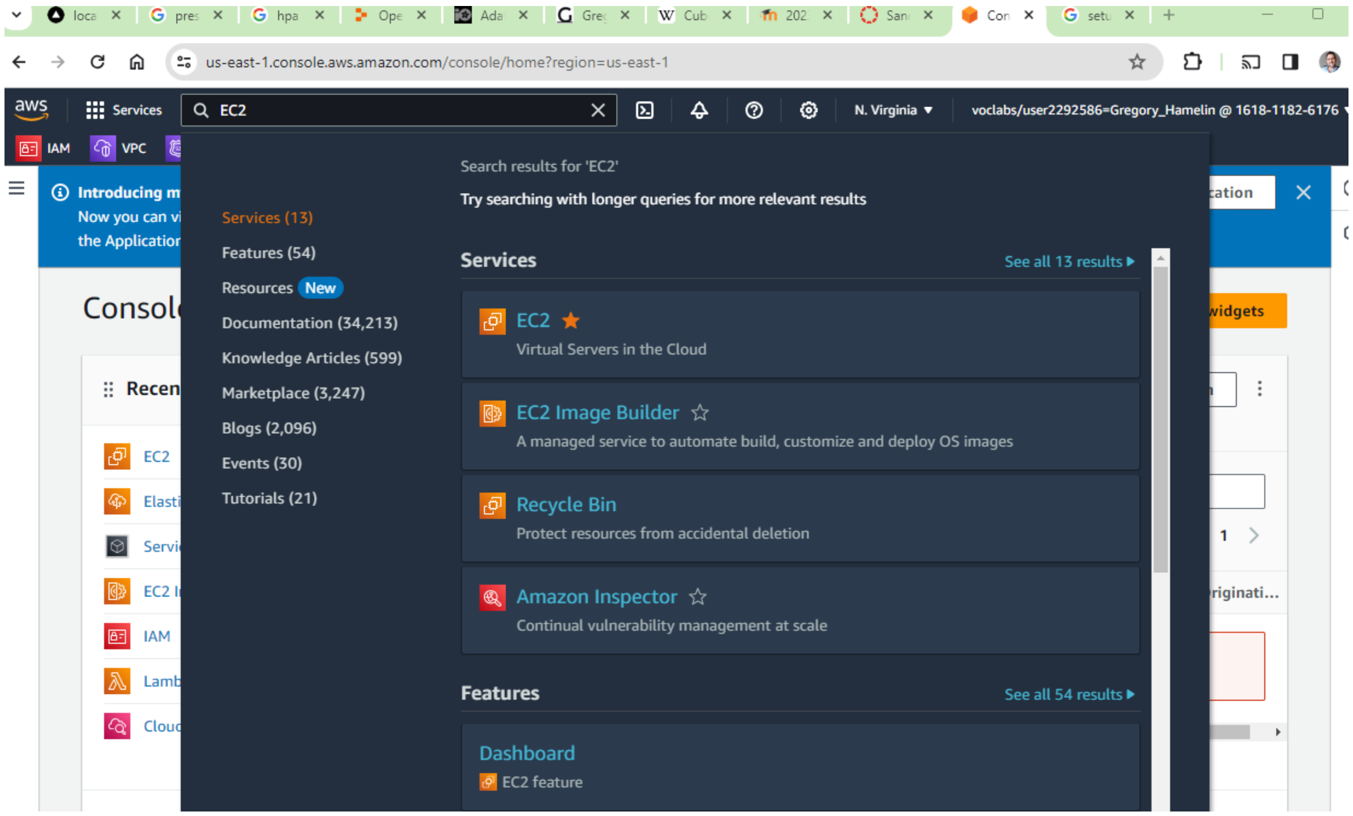
Cost and usage
No cost and usage data
You haven't configured AWS Cost Explorer or you do not have permission.

Build a solution
Start building with simple wizards and automated workflows.

Security — Region: US East (N. Virginia)

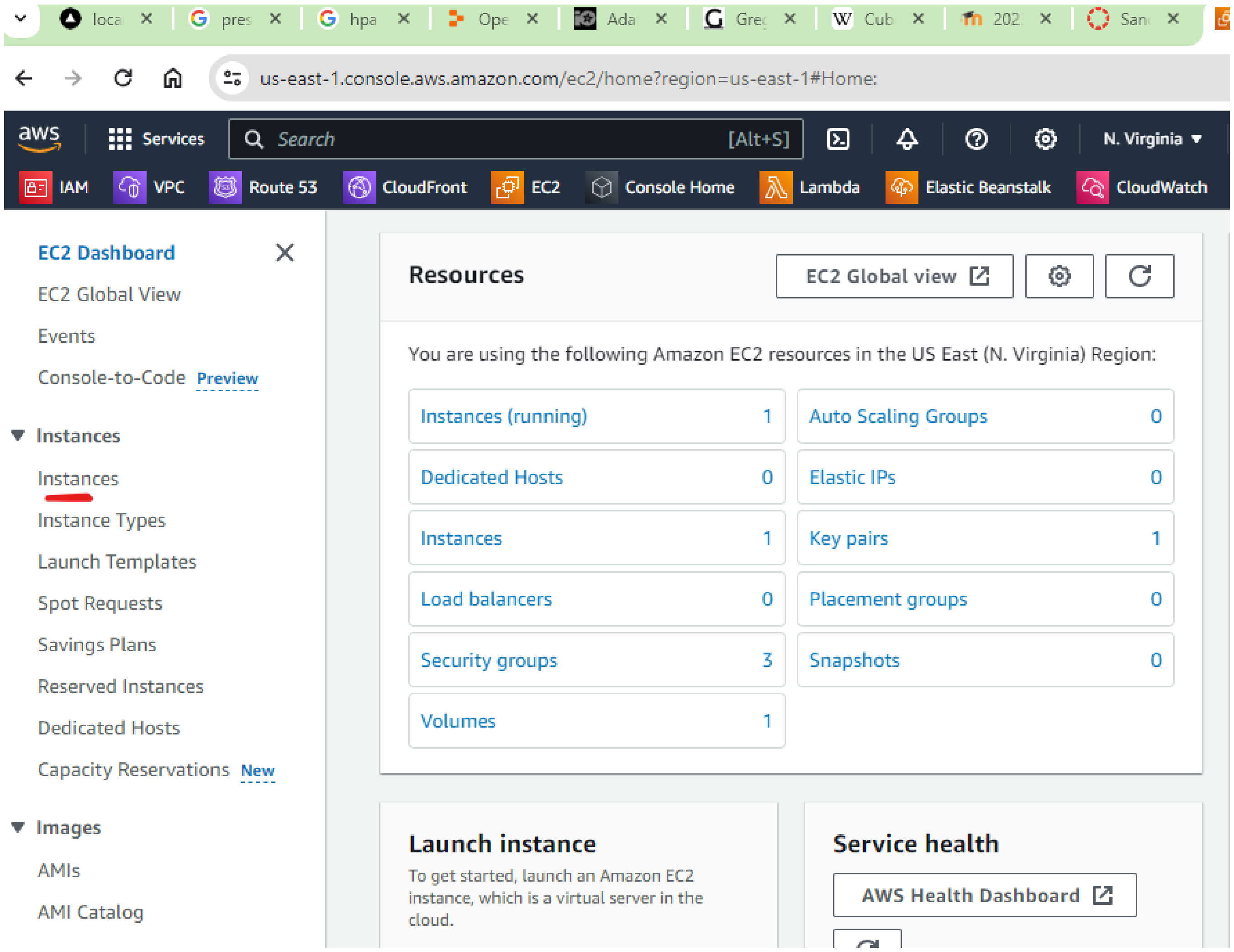
Step 2. Open the EC2 Dashboard

- Go to the search bar and type in EC2, select EC2 to bring you to the EC2 dashboard. I also recommend clicking on the start next to EC2 in the search window because it will be added to your favorites bar at the top of the AWS console and you won't need to search for it.
- You don't need to change your region, but if you are planning on setting up additional resources beyond the EC2 instance, you will need to make sure those resources are also setup in the same region so they can be made available to the EC2 instance.


us-east-1.console.aws.amazon.com/console/home?region=us-east-1
Services
EC2
N. Virginia
voclabs/user2292586=Gregory_Hamelin @ 1618-1182-6176
IAM
VPC
Search results for 'EC2'
Try searching with longer queries for more relevant results
Services (13)
Features (54)
Resources New
Documentation (34,213)
Knowledge Articles (599)
Marketplace (3,247)
Blogs (2,096)
Events (30)
Tutorials (21)
Services
See all 13 results
EC2
Virtual Servers in the Cloud
EC2 Image Builder
A managed service to automate build, customize and deploy OS images
Recycle Bin
Protect resources from accidental deletion
Amazon Inspector
Continual vulnerability management at scale
Features
See all 54 results
Dashboard
EC2 feature

Step 3. Navigate to Instances

- Once you have the EC2 dashboard pulled up, navigate to the Instances dashboard by clicking on "Instances" in the left hand menu.


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Home:

Resources

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

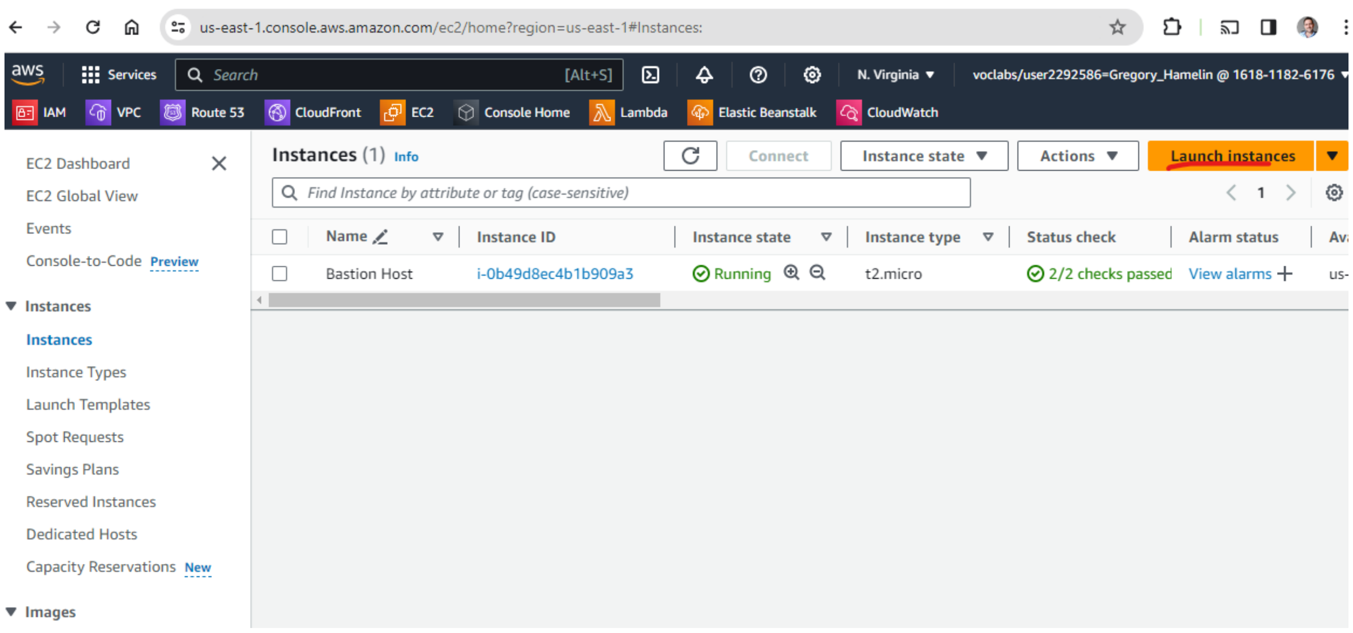
Instances (running)	1	Auto Scaling Groups	0
Dedicated Hosts	0	Elastic IPs	0
Instances	1	Key pairs	1
Load balancers	0	Placement groups	0
Security groups	3	Snapshots	0
Volumes	1		

Launch instance

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

Service health

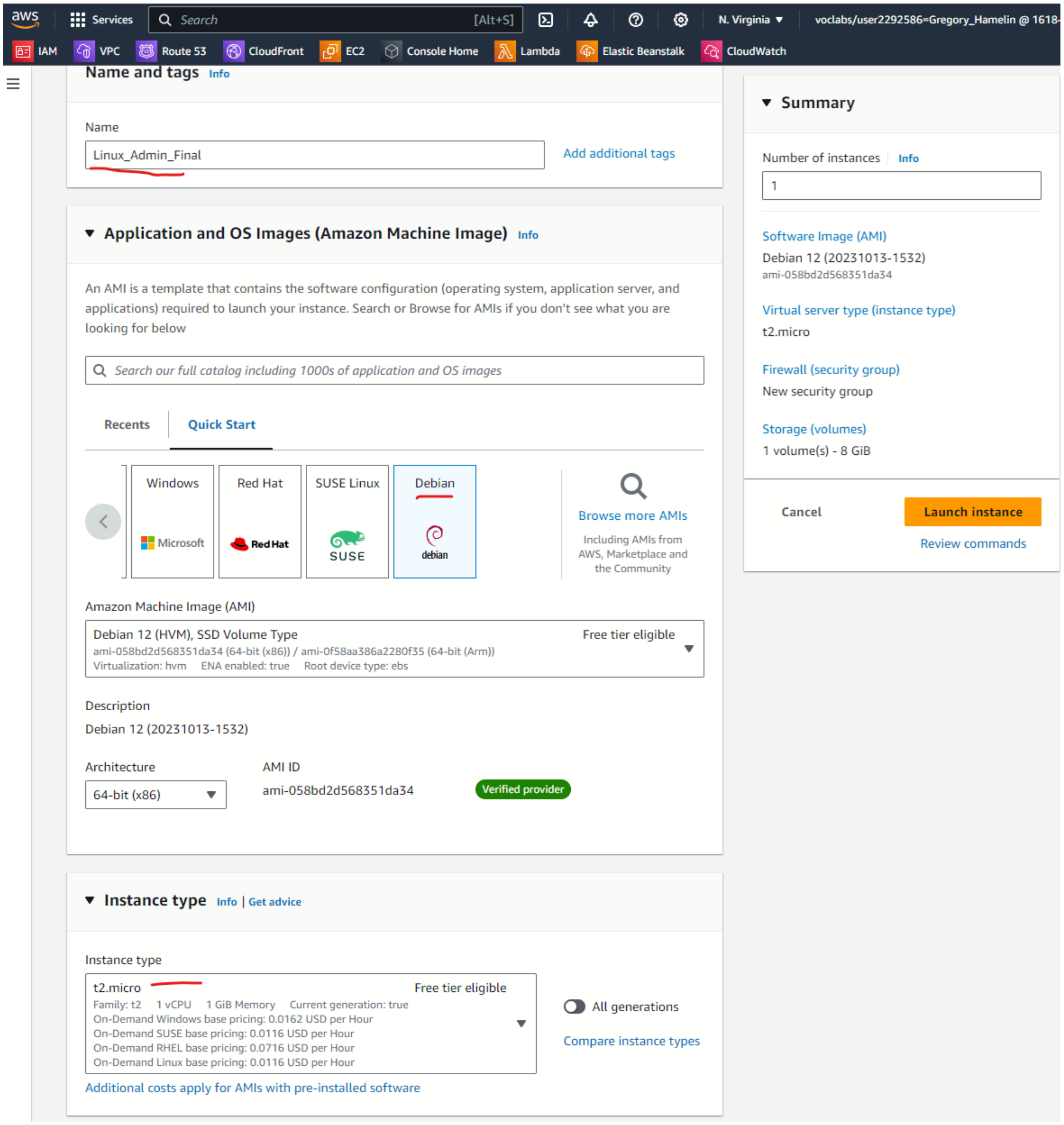
- From there, click on the "Launch Instance" button. AWS uses the term launch interchangeably with create. Launching is what we do to create a new instance. We can create new instances from an existing image provided by AWS or we can use another instance and clone it so to speak.



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Av
Bastion Host	i-0b49d8ec4b1b909a3	Running	t2.micro	2/2 checks passed	View alarms	us-

Step 4. Launch a New Instance

- Under the "Name and tags" section you are going to specify the name of the machine. For my example, my machine name will be "Linux_Admin_Final". Amazon also allows you to add additional tags that can be useful when using the command line to interact with AWS. In our case, we only need the name tag, which will be two components, the tag name also known as a key and the value which will hold the name of our machine.
- Under the section "Application and OS Images(Amazon Machine Image)" choose Debian under the "quick start" options.
- For this example, we're going with good ol' Debian which is the base distribution many of the most popular Linux distributions were forked. I believe Ubuntu among many others originated as a fork of the Debian project.
- Under the section "Instance type" we will leave it at the default t2.micro because it is Free tier eligible.



Services | Search [Alt+S] | N. Virginia | voclabs/user2292586=Gregory_Hamelin @ 1618-

IAM | VPC | Route 53 | CloudFront | EC2 | Console Home | Lambda | Elastic Beanstalk | CloudWatch

Name and tags Info

Name: Linux_Admin_Final

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

Recents | Quick Start

Windows | Red Hat | SUSE Linux | Debian

Browse more AMIs — Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Debian 12 (HVM), SSD Volume Type — Free tier eligible
ami-058bd2d568351da34 (64-bit (x86)) / ami-0f58aa386a2280f35 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description: Debian 12 (20231013-1532)

Architecture: 64-bit (x86)

AMI ID: ami-058bd2d568351da34 — Verified provider

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

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

Summary

Number of instances Info: 1

Software Image (AMI): Debian 12 (20231013-1532) ami-058bd2d568351da34

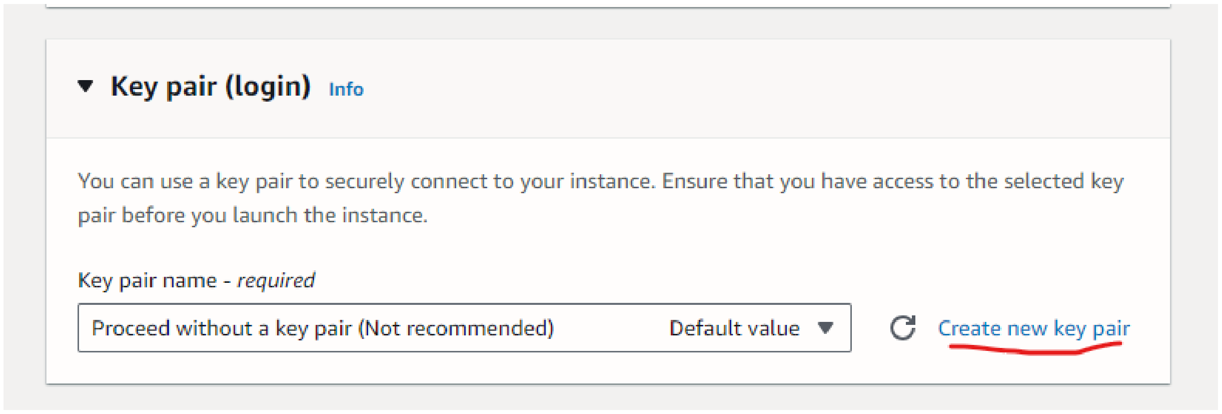
Virtual server type (instance type): t2.micro

Firewall (security group): New security group

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

Cancel | Launch instance | Review commands

- Under the section "Key Pair(login)" click on the "Create new key pair" link. This will allow us to create a new key pair that we can use to connect to our new Linux server using a .PPK file. This file will be used with PuTTY to remotely connect to the AWS Debian Linux instance we are creating. Once you create the new key pair, a .PPK file will automatically download via your web browser. Make sure you know where this file is because we're going to need it later to connect to our machine via PuTTY.



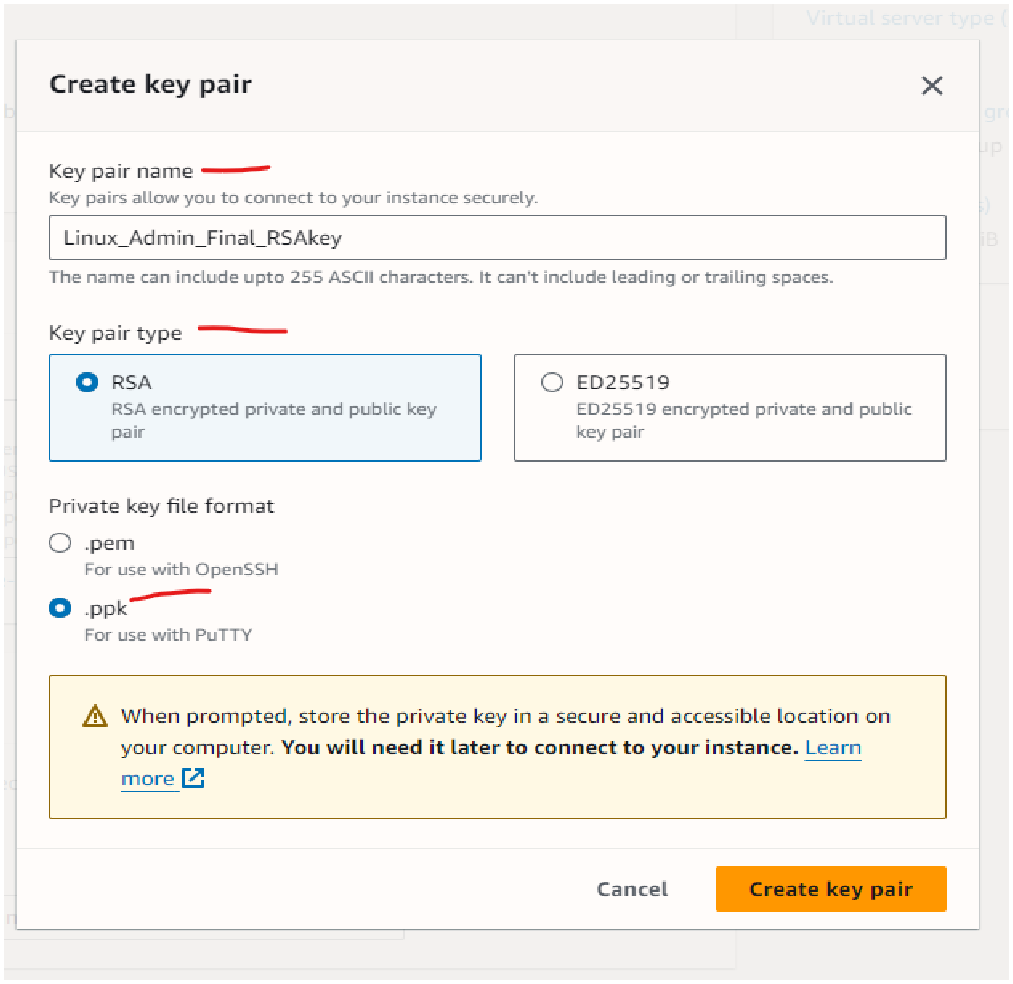
Key pair (login) Info

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*

Proceed without a key pair (Not recommended) Default value

Create new key pair



Create key pair

Key pair name

Key pairs allow you to connect to your instance securely.

Linux_Admin_Final_RSAkey

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

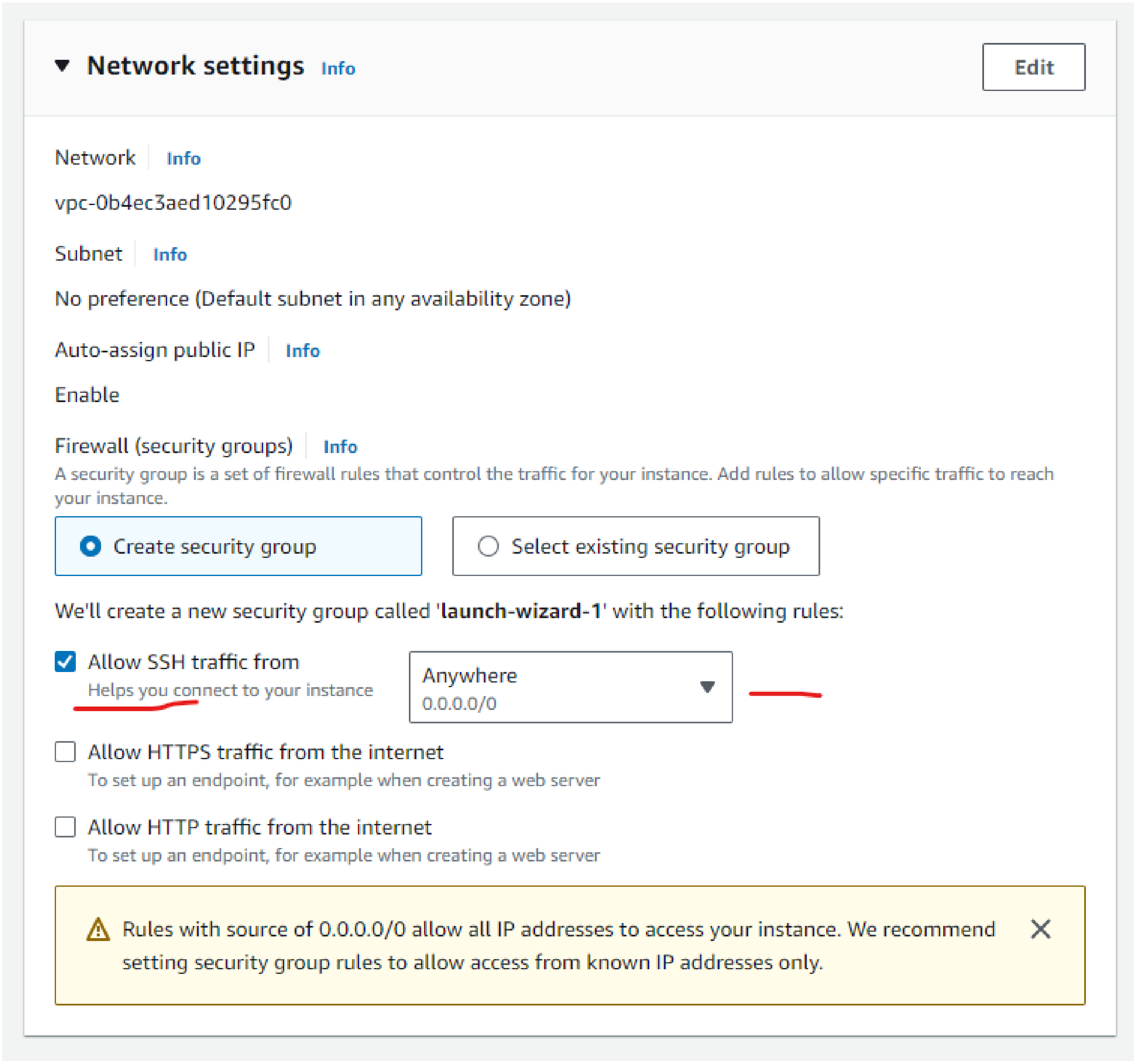
Private key file format

.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

- Under the “Network settings" section you can leave all the fields as the defaults, but this is the section where you can allow HTTPS traffic to your instance or restrict SSH access to your instance to your specific public IP. We won't be using this because I do not have a static IP address reserved with my ISP which means my public IP changes periodically. AWS has a warning about leaving the device accessible from any IP, but since this will not be a production machine and will hold no critical data, I will not worry about it.



▼ Network settings Info

Network Info

vpc-0b4ec3aed10295fc0

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:

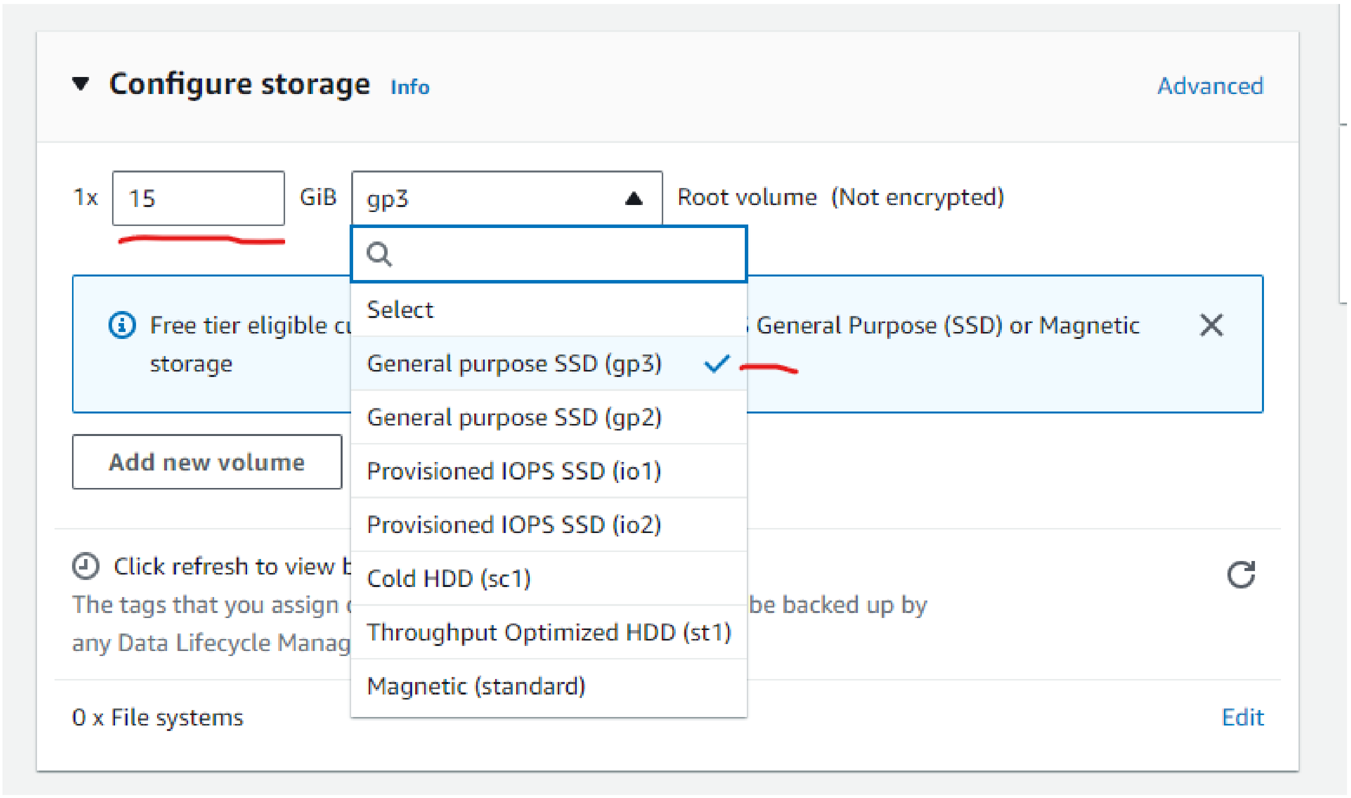
☑ Allow SSH traffic from — Anywhere 0.0.0.0/0
Helps you connect to your instance

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

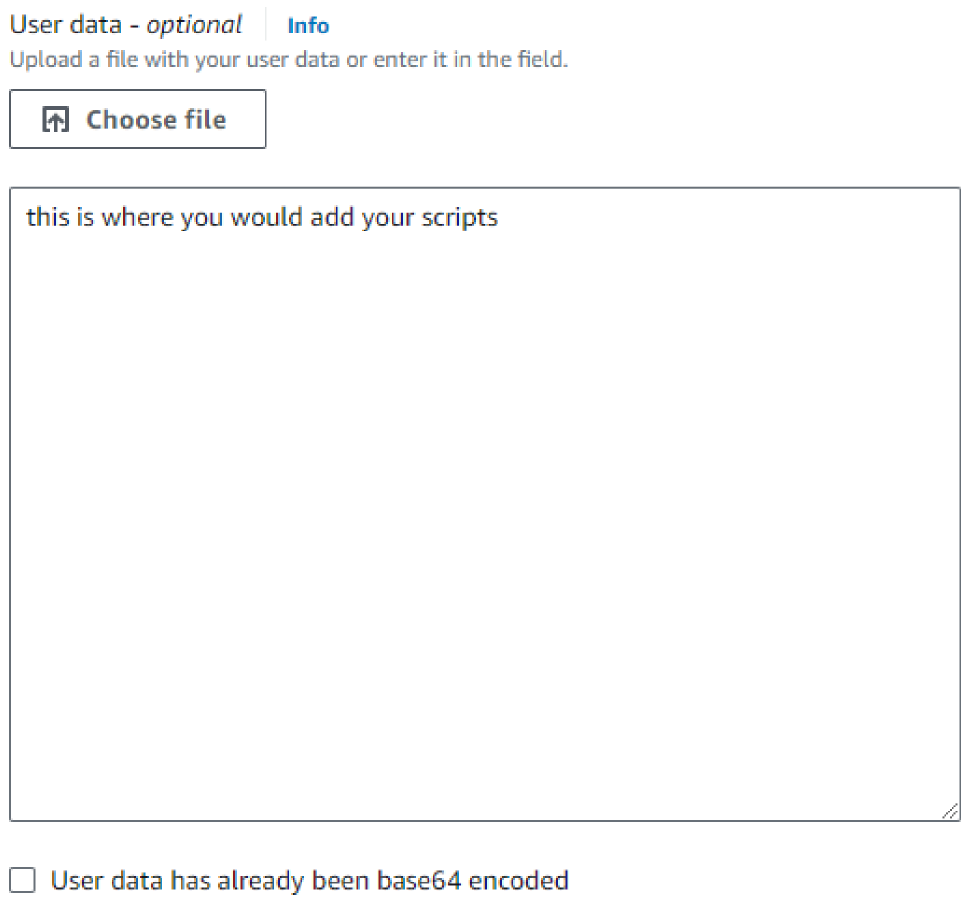
☐ Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

⚠ 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.

- Under "Configure storage" we will also leave those settings as the default because we aren't using this machine in production and only needs minimal resources. This is the section where you could add additional storage volumes that are directly mounted to the machine. Amazon also offers S3(Simple Storage Service) storage which are designed for certain use cases instead of adding a separate volume on the machine instance itself which uses Amazon's Elastic Block Store or Amazon EBS. I'm going to increase the storage to 15GB and leave the drive type as the default "gp3". You can specify different kinds of drives based on how much throughput you need for your specific use case. In this case we don't need a high-performance root volume for our Linux machine so we will stick with the "gp3" general purpose SSD.



- We won't be changing anything under the "Advanced details" section, but this is where you can make a lot more configuration specifications as well as setting specific commands to be run when the machine starts using the "User data" section.

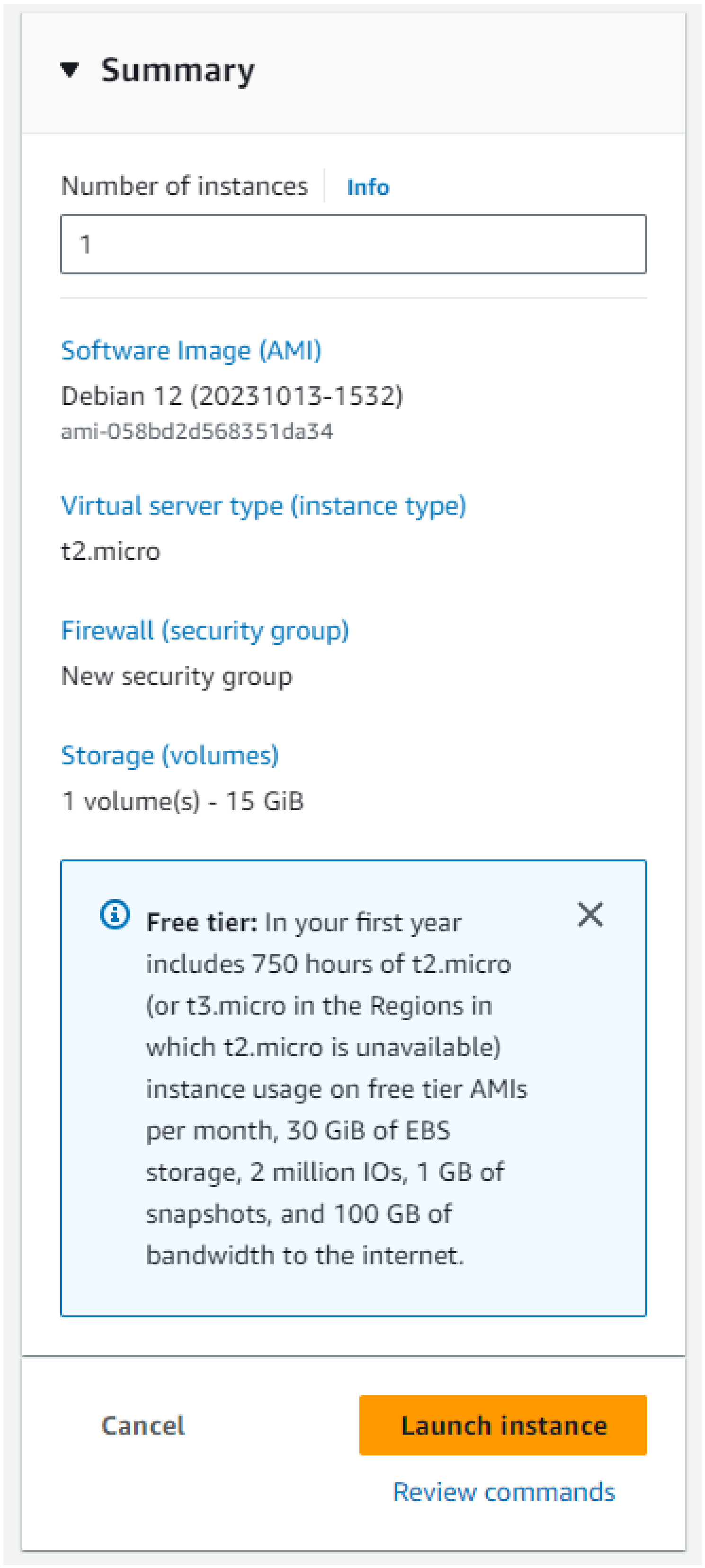


User data - *optional* Info
Upload a file with your user data or enter it in the field.

this is where you would add your scripts

☐ User data has already been base64 encoded

- Review the information under the "Summary" section and click launch instance to create our new Debian amazon machine instance (AMI).



Number of instances | Info

1

Software Image (AMI)

Debian 12 (20231013-1532)
ami-058bd2d568351da34

Virtual server type (instance type)

t2.micro

Firewall (security group)

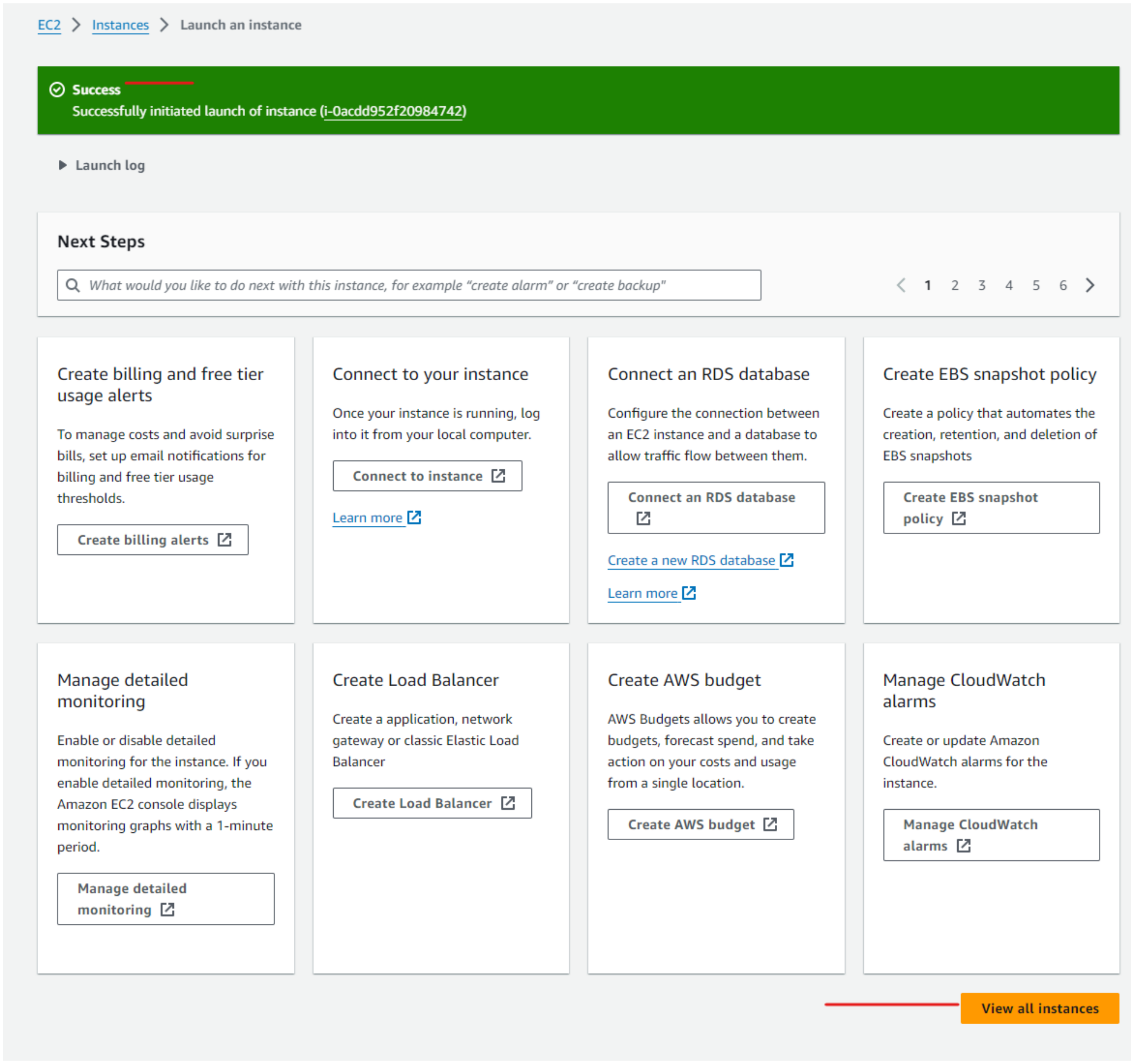
New security group

Storage (volumes)

1 volume(s) - 15 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 snapshots, and 100 GB of bandwidth to the internet.

- You should see a message displayed at the top of the screen saying "Success" informing you that your instance has been launched. From here we're going to click the "View all instances" button at the bottom of the screen.


EC2 > Instances > Launch an instance

Success
Successfully initiated launch of instance (i-0acdd952f20984742)

▶ Launch log

Next Steps

What would you like to do next with this instance, for example "create alarm" or "create backup"

< 1 2 3 4 5 6 >

Create billing and free tier usage alerts
To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.
Create billing alerts

Connect to your instance
Once your instance is running, log into it from your local computer.
Connect to instance
Learn more

Connect an RDS database
Configure the connection between an EC2 instance and a database to allow traffic flow between them.
Connect an RDS database
Create a new RDS database
Learn more

Create EBS snapshot policy
Create a policy that automates the creation, retention, and deletion of EBS snapshots
Create EBS snapshot policy

Manage detailed monitoring
Enable or disable detailed monitoring for the instance. If you enable detailed monitoring, the Amazon EC2 console displays monitoring graphs with a 1-minute period.
Manage detailed monitoring

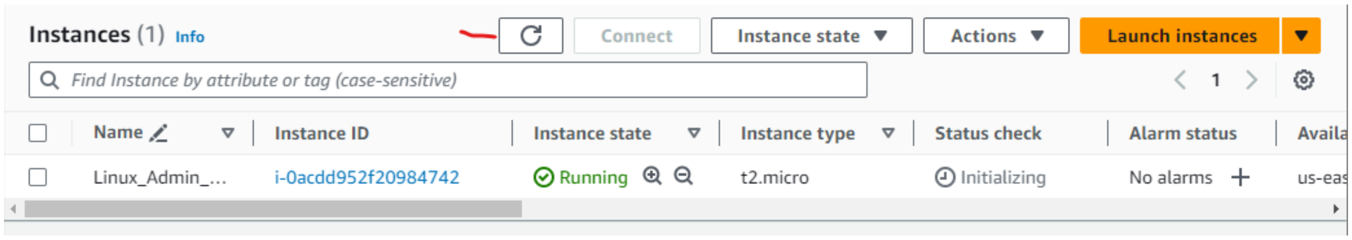
Create Load Balancer
Create a application, network gateway or classic Elastic Load Balancer
Create Load Balancer

Create AWS budget
AWS Budgets allows you to create budgets, forecast spend, and take action on your costs and usage from a single location.
Create AWS budget

Manage CloudWatch alarms
Create or update Amazon CloudWatch alarms for the instance.
Manage CloudWatch alarms

View all instances

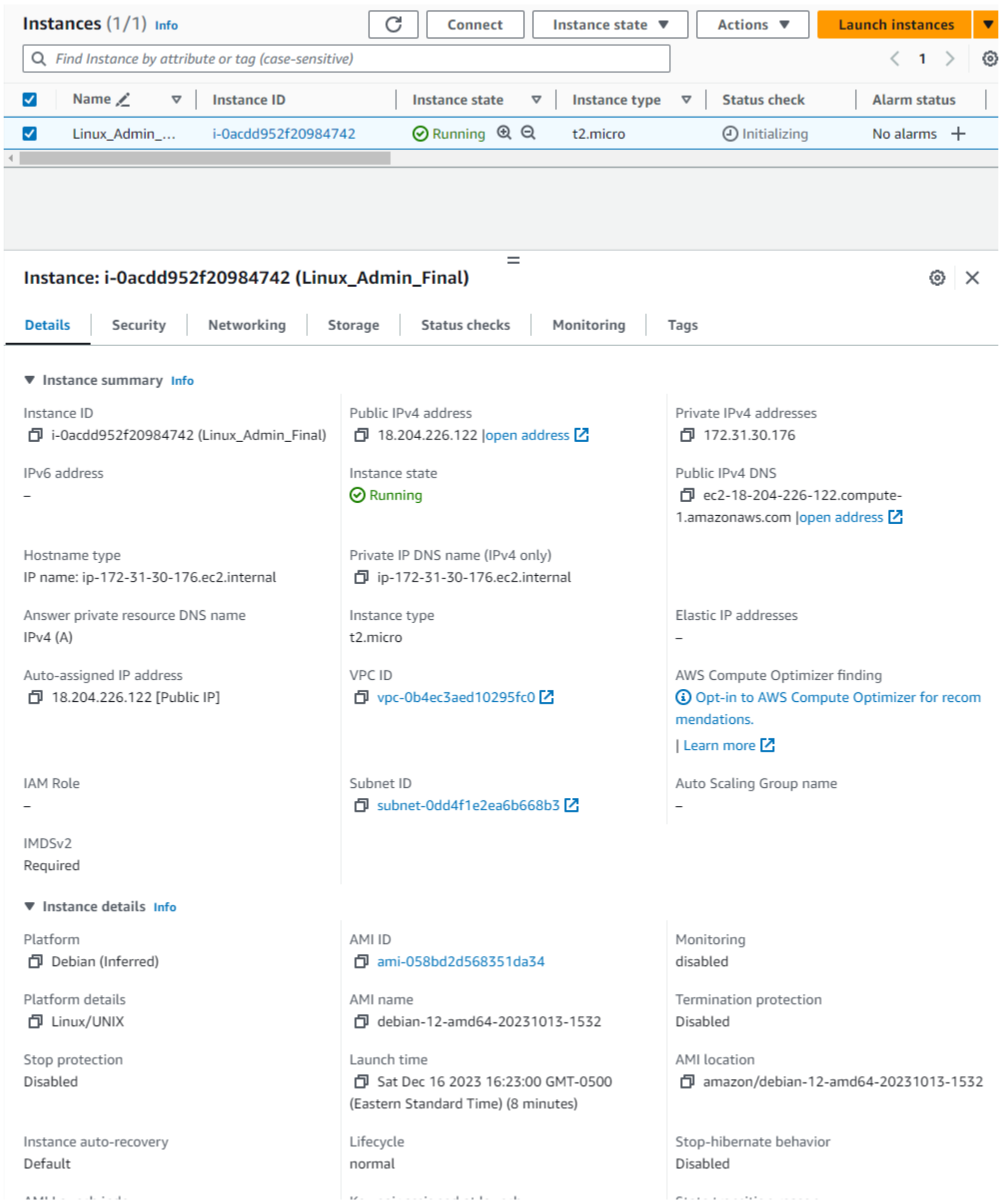
- You may need to refresh your instances list to display the new machine we created. Click on the circle icon in the instances dashboard to refresh.



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availa
Linux_Admin_...	i-0acdd952f20984742	Running	t2.micro	Initializing	No alarms	us-eas

Step 5. View Your New Instance

- Click on the check-box next to the instance name to display information about the instance in the screen below. You can find lots of helpful information for connecting to your machine information to connect your machine to other AWS resources like what specific VPC(virtual private cloud) where your machine instance is running. This is also where you can add or modify storage volumes under the "Storage" tab or make network interface changes under the "Networking" section. We don't need to make any changes in this example.



	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
✓	Linux_Admin_...	i-0acdd952f20984742	Running	t2.micro	Initializing	No alarms

Instance: i-0acdd952f20984742 (Linux_Admin_Final)

Instance summary Info

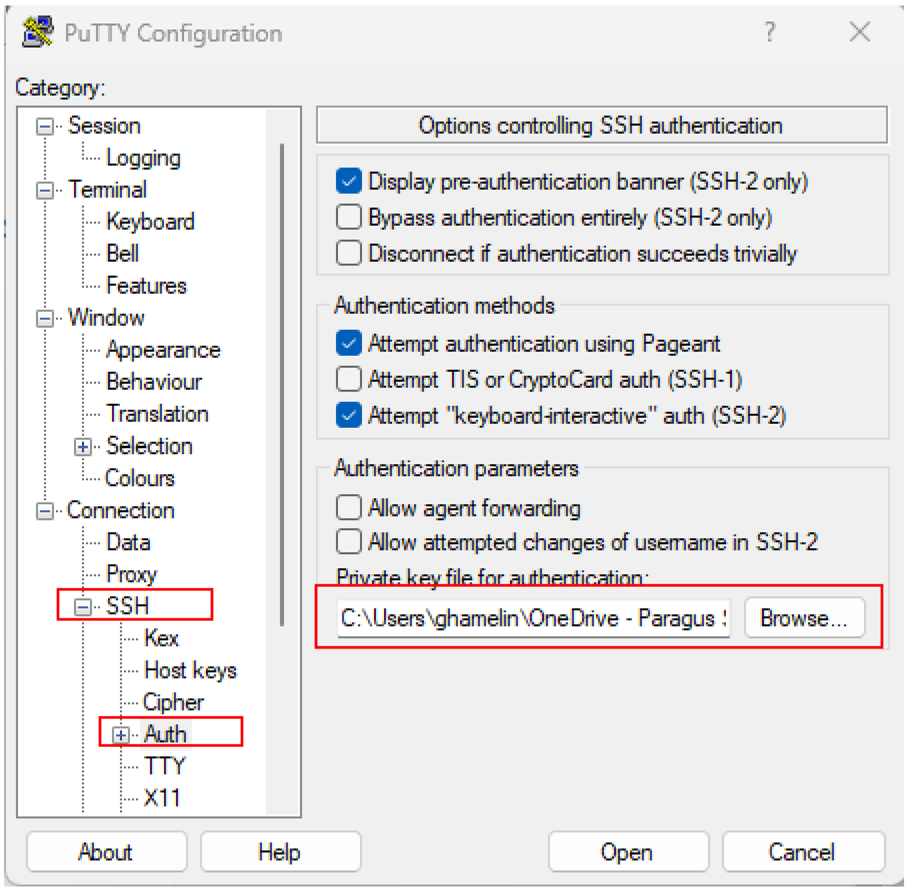
Instance ID: i-0acdd952f20984742 (Linux_Admin_Final)	Public IPv4 address: 18.204.226.122 |open address	Private IPv4 addresses: 172.31.30.176
IPv6 address: –	Instance state: Running	Public IPv4 DNS: ec2-18-204-226-122.compute-1.amazonaws.com |open address
Hostname type: IP name: ip-172-31-30-176.ec2.internal	Private IP DNS name (IPv4 only): ip-172-31-30-176.ec2.internal	
Answer private resource DNS name: IPv4 (A)	Instance type: t2.micro	Elastic IP addresses: –
Auto-assigned IP address: 18.204.226.122 [Public IP]	VPC ID: vpc-0b4ec3aed10295fc0	AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role: –	Subnet ID: subnet-0dd4f1e2ea6b668b3	Auto Scaling Group name: –
IMDSv2: Required		

Instance details Info

Platform: Debian (Inferred)	AMI ID: ami-058bd2d568351da34	Monitoring: disabled
Platform details: Linux/UNIX	AMI name: debian-12-amd64-20231013-1532	Termination protection: Disabled
Stop protection: Disabled	Launch time: Sat Dec 16 2023 16:23:00 GMT-0500 (Eastern Standard Time) (8 minutes)	AMI location: amazon/debian-12-amd64-20231013-1532
Instance auto-recovery: Default	Lifecycle: normal	Stop-hibernate behavior: Disabled

Step 6. Connect to Your New Linux Server Using PuTTY

- If you don't already have PuTTY installed on your computer you will need to install it. You can download PuTTY by going to **Putty.org**
- Once you have PuTTY installed we're going to add the .PPK file for authentication. Click on SSH on the left-hand menu, click on the plus "+" sign next to SSH and click on "Auth". Click on the "Browse..." button next to where it says "Private key file for authentication" to add our .PPK file to PuTTY.



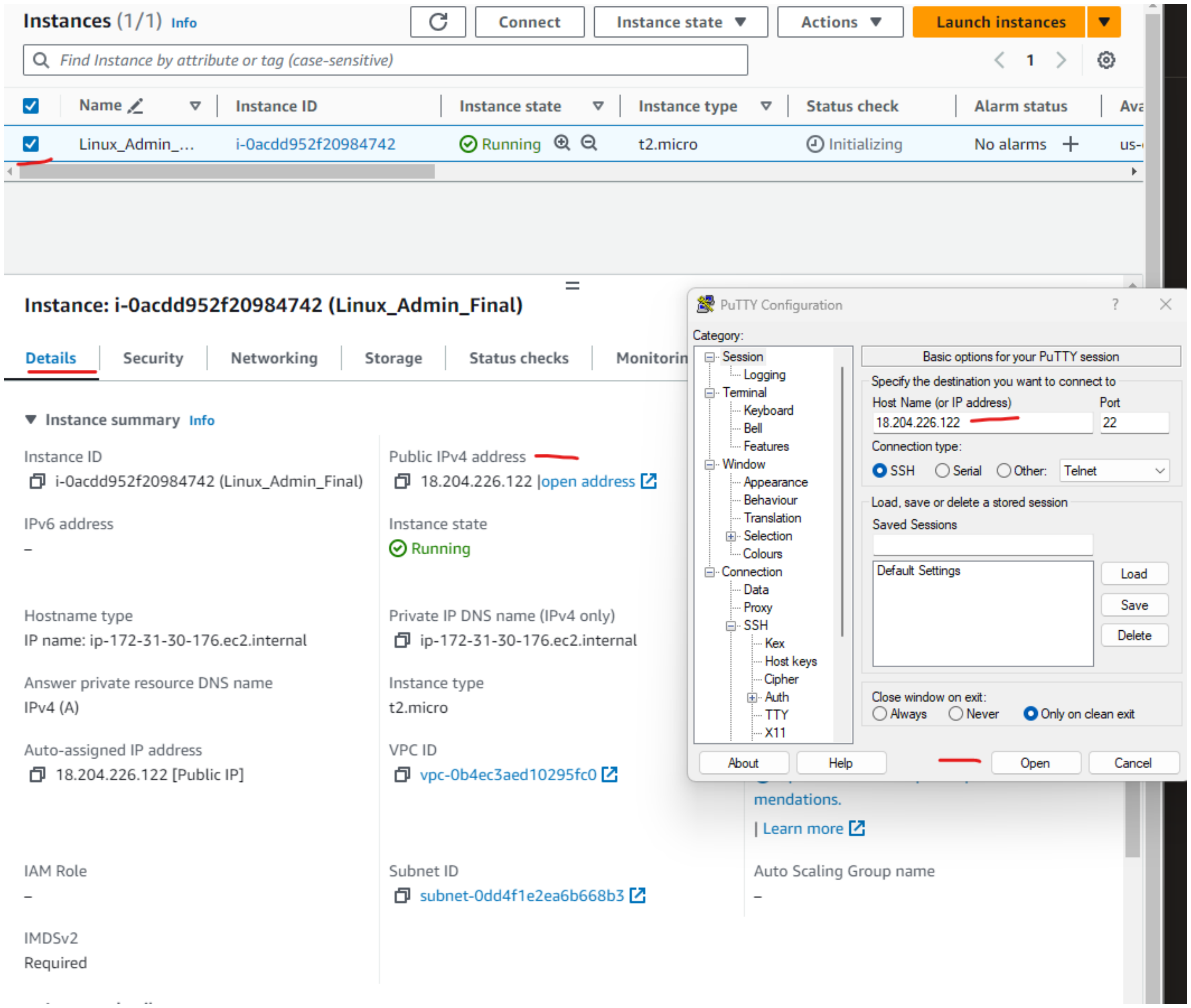
Options controlling SSH authentication

[✓] Display pre-authentication banner (SSH-2 only)
[ ] Bypass authentication entirely (SSH-2 only)
[ ] Disconnect if authentication succeeds trivially

Authentication methods
[✓] Attempt authentication using Pageant
[ ] Attempt TIS or CryptoCard auth (SSH-1)
[✓] Attempt "keyboard-interactive" auth (SSH-2)

Authentication parameters
[ ] Allow agent forwarding
[ ] Allow attempted changes of username in SSH-2
Private key file for authentication:
C:\Users\ghamelin\OneDrive - Paragus !

- Once you've added your .PPK file, click on "Session" in the left hand menu. You will see a field for "Host Name (or IP address)" this is where we will put the Public IPv4 address for our newly created Debian server. To get the public address for our machine, we will go back to AWS, make sure our instance is selected and copy the IP address listed in the details section under "Public IPv4 address." Now paste this IP address in PuTTY under the Host Name field and click the "open" button to connect to our machine. You may be prompted to click accept because it's a new SSH connection, click accept and continue.



	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Ava
✓	Linux_Admin_...	i-0acdd952f20984742	Running	t2.micro	Initializing	No alarms	us-

Instance: i-0acdd952f20984742 (Linux_Admin_Final)

Instance summary Info

Instance ID
i-0acdd952f20984742 (Linux_Admin_Final)

Public IPv4 address
18.204.226.122 |open address

IPv6 address
–

Instance state
Running

Hostname type
IP name: ip-172-31-30-176.ec2.internal

Private IP DNS name (IPv4 only)
ip-172-31-30-176.ec2.internal

Answer private resource DNS name
IPv4 (A)

Instance type
t2.micro

Auto-assigned IP address
18.204.226.122 [Public IP]

VPC ID
vpc-0b4ec3aed10295fc0

IAM Role
–

Subnet ID
subnet-0dd4f1e2ea6b668b3

Auto Scaling Group name
–

IMDSv2
Required

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): 18.204.226.122
Port: 22

Connection type: [✓ SSH] Serial Other: Telnet

Load, save or delete a stored session

Saved Sessions

Default Settings

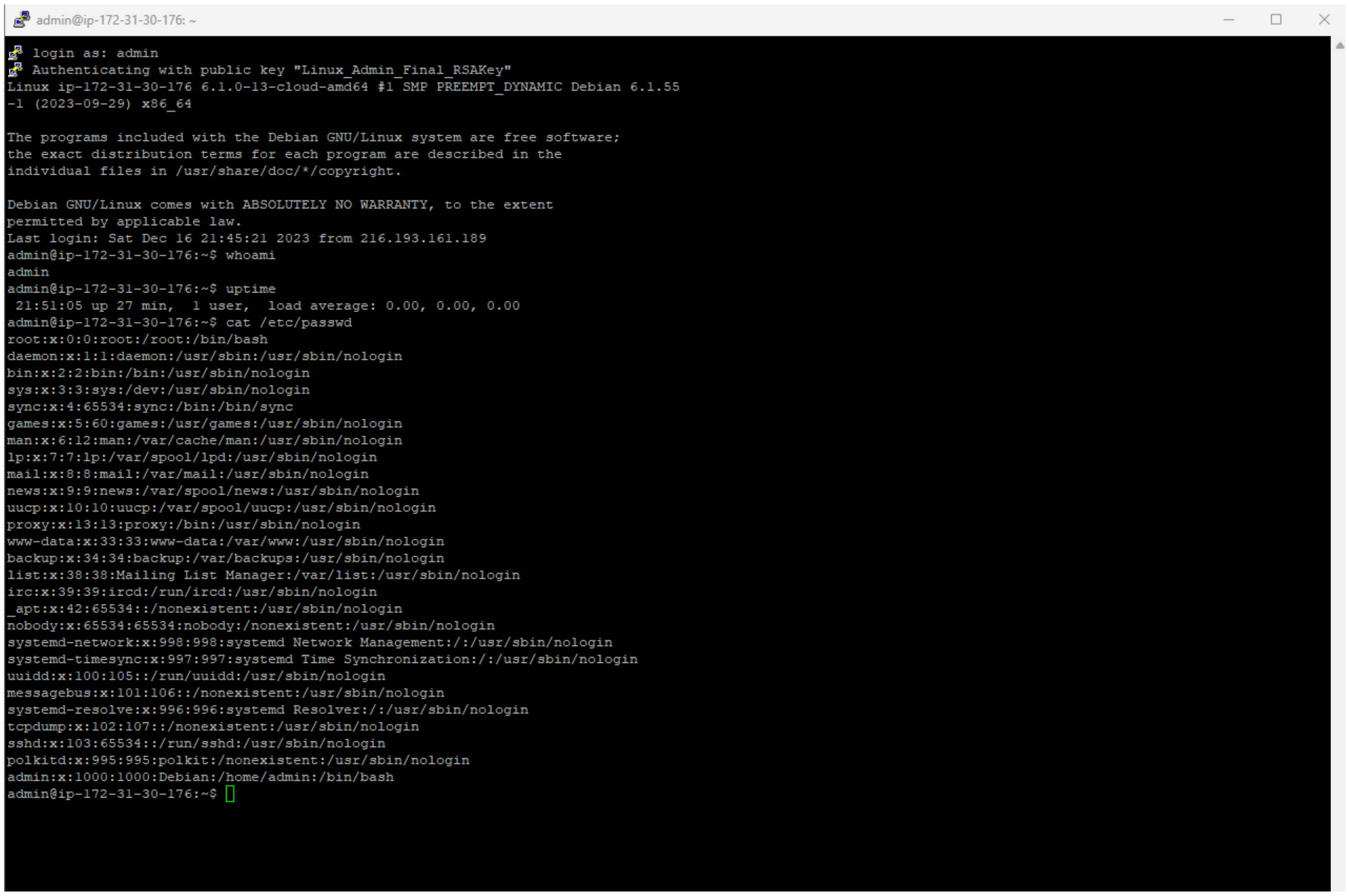
Close window on exit: Always Never [✓ Only on clean exit]

Step 7. Login to Your User Account

- Now you will be prompted, in your PuTTY terminal session, to specify your username. In this case we will type "admin" for our username and hit enter. Our PPK file will handle the rest of the authentication from there.
- You should see the generic login message and then you should see a terminal where we can now issue commands and look around the file system. If you want to exit, type "exit" and hit enter. The unfortunate default behavior for PuTTY is to close the application on exit which can be a little tedious. There may be a setting to change that, but I would recommend using an application like mRemote which allows you to setup multiple PuTTY, RDP and other connection files and manage them all from one pane. When you close the connection mRemote will stay open and can also handle multiple

simultaneous connections. mRemote is free as is PuTTY so I recommend checking it out.

- Once I was logged in, I ran a few basic commands including “cat /etc/passwd” to see the contents of the passwd file. We can our admin user at the bottom of the file.



```
login as: admin
Authenticating with public key "Linux_Admin_Final_RSAKey"
Linux ip-172-31-30-176 6.1.0-13-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.55
-1 (2023-09-29) x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Dec 16 21:45:21 2023 from 216.193.161.189
admin@ip-172-31-30-176:~$ whoami
admin
admin@ip-172-31-30-176:~$ uptime
 21:51:05 up 27 min,  1 user,  load average: 0.00, 0.00, 0.00
admin@ip-172-31-30-176:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
systemd-timesync:x:997:997:systemd Time Synchronization:/:/usr/sbin/nologin
uuidd:x:100:105::/run/uuidd:/usr/sbin/nologin
messagebus:x:101:106::/nonexistent:/usr/sbin/nologin
systemd-resolve:x:996:996:systemd Resolver:/:/usr/sbin/nologin
tcpdump:x:102:107::/nonexistent:/usr/sbin/nologin
sshd:x:103:65534::/run/sshd:/usr/sbin/nologin
polkitd:x:995:995:polkit:/nonexistent:/usr/sbin/nologin
admin:x:1000:1000:Debian:/home/admin:/bin/bash
admin@ip-172-31-30-176:~$
```