Placement Empowerment Program

Cloud Computing and DevOps Centre

Secure Access with a Bastion HostSet up a bastion host in a public subnet to securely access instances in a private subnet.

Name: **Kevin Christopher Ruban**
Department: **AI-ML**

Introduction

In cloud environments, securing access to private instances is crucial. A **Bastion Host** (or Jump Box) is a special-purpose instance that acts as a secure gateway to access EC2 instances in a private subnet. Instead of exposing private instances directly to the internet, users connect to the Bastion Host first and then access the private instances from there.
This setup **enhances security** by limiting direct SSH access to private instances and applying strict security controls.

Overview

We will set up a **Bastion Host** in a **public subnet** that provides controlled SSH access to instances inside a **private subnet**.

What We Will Do?

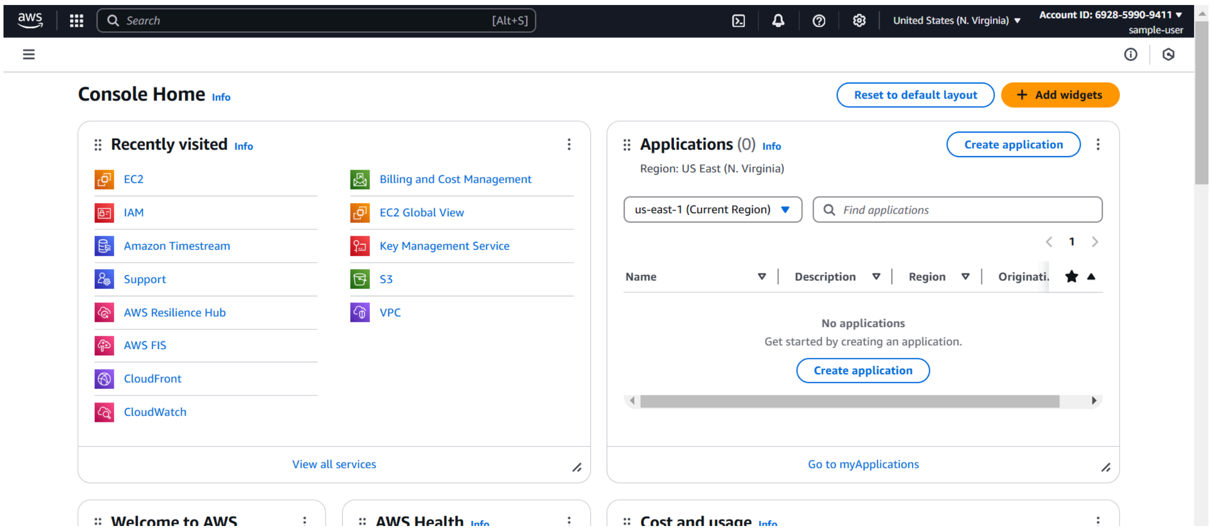
1. **Create a VPC** with a **Public** and **Private Subnet.**
2. **Set Up a Bastion Host** in the Public Subnet.
3. **Launch a Private EC2 Instance** in the Private Subnet.
4. **Configure Secure SSH Access** via the Bastion Host.
5. **Enhance Security** by restricting SSH access and

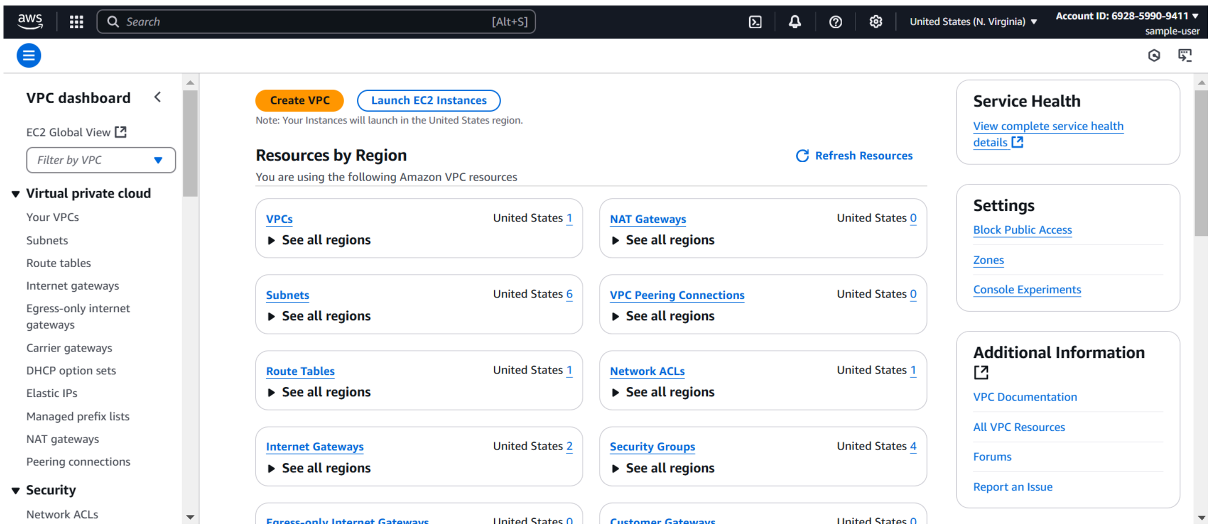
considering AWS Systems Manager as an alternative.

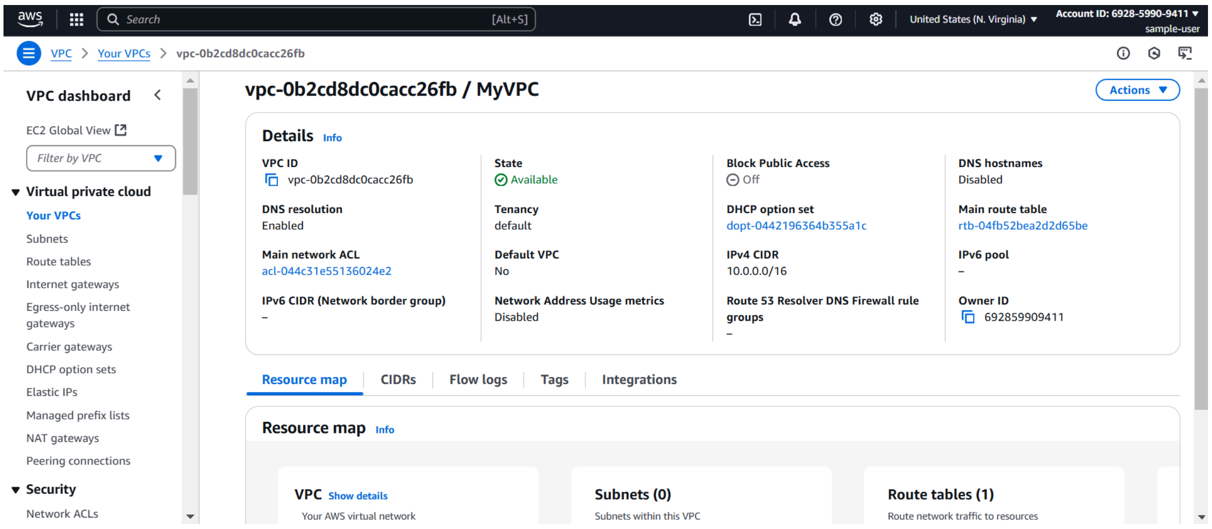
Step 1:
Create a VPC with Public and Private Subnets

1.1 Create a VPC

- Go to AWS Console → VPC Dashboard.
- Click Create VPC and name it MyVPC.
- Set IPv4 CIDR Block: 10.0.0.0/16.
- Click Create VPC.






vpc-0b2cd8dc0cacc26fb / MyVPC

Details

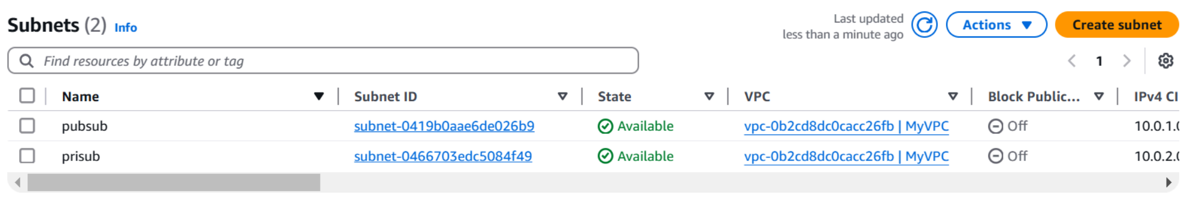
VPC ID vpc-0b2cd8dc0cacc26fb	State Available	Block Public Access Off	DNS hostnames Disabled
DNS resolution Enabled	Tenancy default	DHCP option set dopt-0442196364b355a1c	Main route table rtb-04fb52bea2d2d65be
Main network ACL acl-044c31e55136024e2	Default VPC No	IPv4 CIDR 10.0.0.0/16	IPv6 pool –
IPv6 CIDR (Network border group) –	Network Address Usage metrics Disabled	Route 53 Resolver DNS Firewall rule groups –	Owner ID 692859909411

1.2 Create a Public Subnet

- Go to **Subnets → Create Subnet**.
- Select **MyVPC** and set CIDR block 10.0.1.0/24.
- Enable **Auto-Assign Public IP**.

1.3 Create a Private Subnet

- Repeat the same process, but use CIDR block 10.0.2.0/24.
- **Do not enable** Auto-Assign Public IP.


Subnets (2)

Name	Subnet ID	State	VPC	Block Public...	IPv4 CI
pubsub	subnet-0419b0aae6de026b9	Available	vpc-0b2cd8dc0cacc26fb | MyVPC	Off	10.0.1.(
prisub	subnet-0466703edc5084f49	Available	vpc-0b2cd8dc0cacc26fb | MyVPC	Off	10.0.2.(

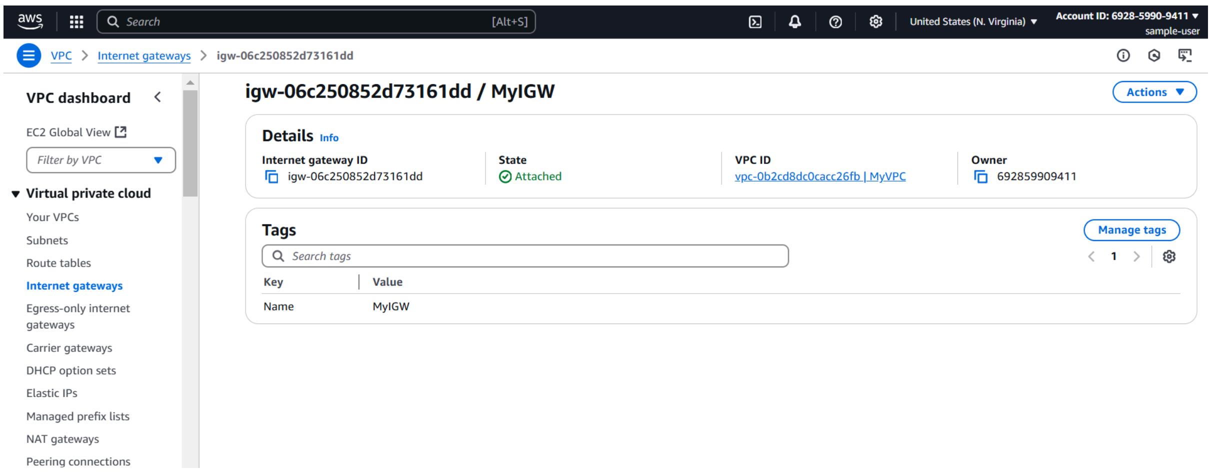
Info
Last updated less than a minute ago
Actions
Create subnet
Find resources by attribute or tag
1

Step 2:
Configure Public Subnet for **Internet Access**
2.1 **Create** an **Internet Gateway (IGW)**

- Go to **Internet Gateways** → Click **Create**

Internet Gateway.

- Name it **MyIGW**, attach it to **MyVPC**.



igw-06c250852d73161dd / MyIGW

Details Info

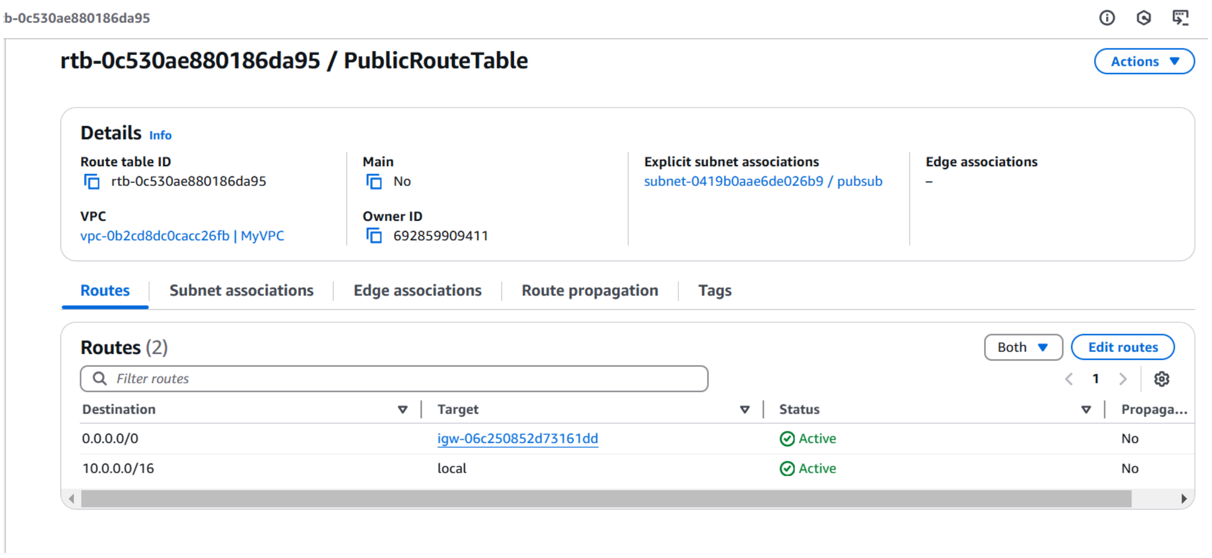
Internet gateway ID	State	VPC ID	Owner
igw-06c250852d73161dd	Attached	vpc-0b2cd8dc0cacc26fb | MyVPC	692859909411

Tags

Key	Value
Name	MyIGW

2.2 Update Public Route Table

- Go to **Route Tables → Create Route Table →** Name it **PublicRouteTable**.
- Associate it with **PublicSubnet**.
- Add a route:
  - **Destination:** 0.0.0.0/0
  - **Target: Internet Gateway (MyIGW)**



rtb-0c530ae880186da95 / PublicRouteTable

Details Info

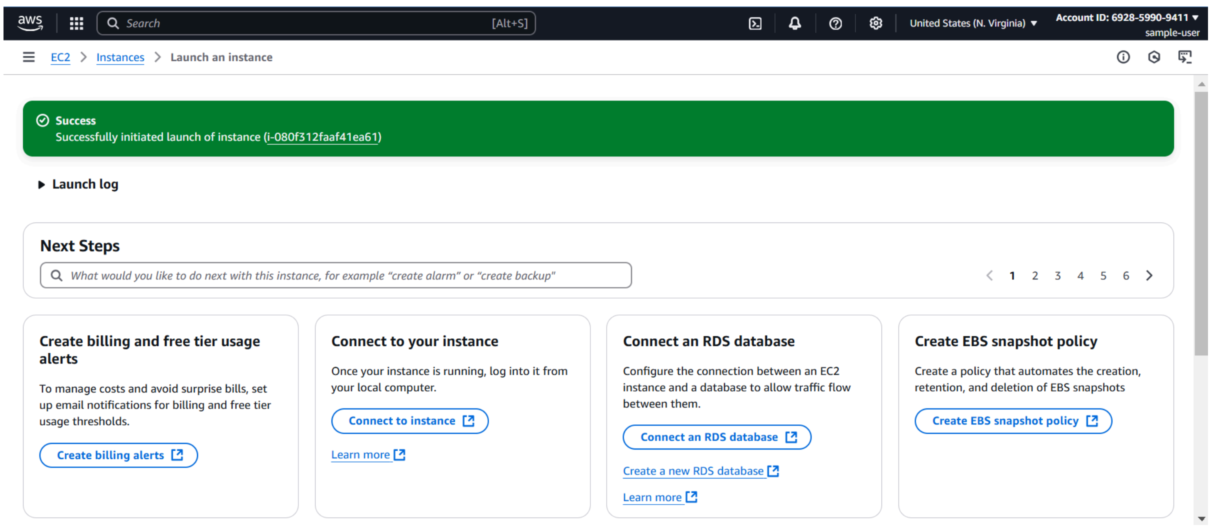
Route table ID: rtb-0c530ae880186da95
Main: No
Explicit subnet associations: subnet-0419b0aae6de026b9 / pubsub
Edge associations: –
VPC: vpc-0b2cd8dc0cacc26fb | MyVPC
Owner ID: 692859909411

Routes (2)

Destination	Target	Status	Propaga...
0.0.0.0/0	igw-06c250852d73161dd	Active	No
10.0.0.0/16	local	Active	No

Step 3:
Launch a Bastion Host (Public Subnet)

1. Go to **EC2 Dashboard → Launch Instance**.
2. Select **Amazon Linux 2** (or **Ubuntu**).
3. Choose **t2.micro (Free Tier Eligible)**.
4. Place it in **PublicSubnet** with **Auto-Assign Public IP enabled**.
5. Create a **Security Group (BastionSG)**:
   - Allow **SSH (Port 22) from Your IP** (xx.xx.xx.xx/32).
6. Create or use an **existing key pair** (e.g., bastion-key.pem).
7. Click **Launch**.



aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 6928-5990-9411
sample-user
EC2 > Instances > Launch an instance

Success
Successfully initiated launch of instance (i-080f312faaf41ea61)

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

Step 4:
Launch a Private EC2 Instance

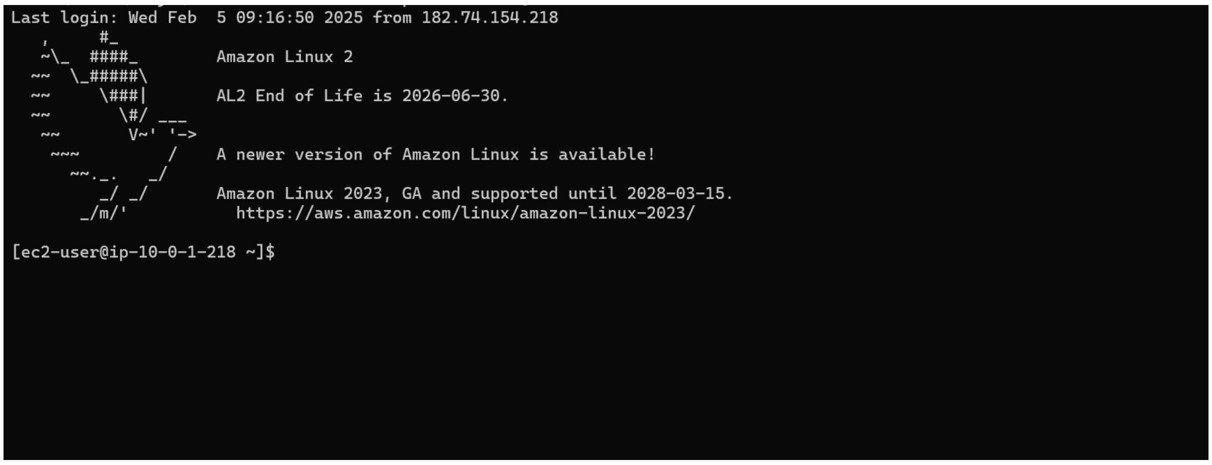
1. Go to **EC2 Dashboard → Launch Instance**.
2. Choose **Amazon Linux 2** (or **Ubuntu**).
3. Choose **t2.micro** and place it in **PrivateSubnet**.
4. **Disable Auto-Assign Public IP**.
5. Create a **Security Group (PrivateSG)**:
   - Allow **SSH (Port 22) only from Bastion Host's Security Group**.
6. Use the same **key pair** (bastion-key.pem).
7. Click **Launch**.

Step 5: Connect to the Private Instance Using the Bastion Host

5.1 Connect to the Bastion Host

```
ssh -i bastion-key.pem ec2-user@<bastion-public-ip>
```

(Replace <bastion-public-ip> with the actual Bastion Host public IP.)



```
Last login: Wed Feb  5 09:16:50 2025 from 182.74.154.218
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-10-0-1-218 ~]$
```

5.2 SSH from Bastion to Private Instance

1. Copy the bastion-key.pem file to the Bastion

Host:

```
scp -i bastion-key.pem bastion-key.pem
ec2-user@<bastion-public-ip>:~/
```

2. Connect to the Bastion Host:

```
ssh -i bastion-key.pem ec2-user@<bastion-public-ip>
```

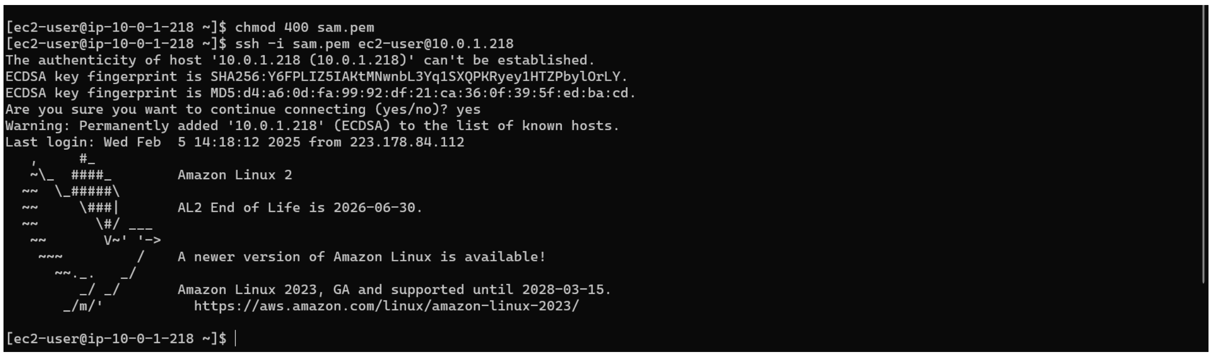
3. Change permissions for the key file:

```
chmod 400 bastion-key.pem
```

4. SSH into the Private Instance from the Bastion Host:

```
ssh -i bastion-key.pem ec2-user@<private-instance-ip>
```

(Replace <private-instance-ip> with the private IP of your instance.)



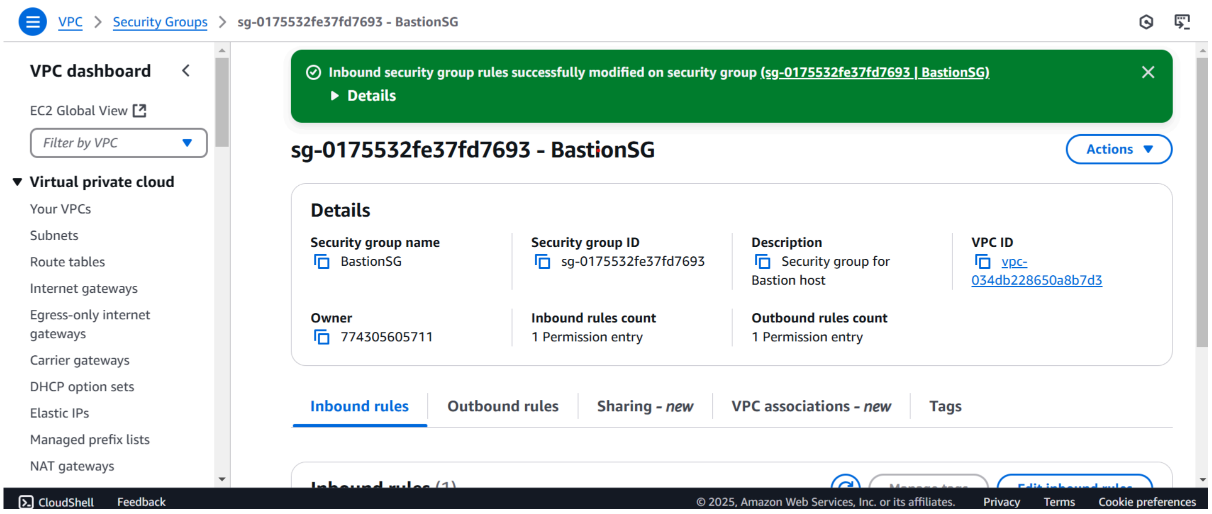
```
[ec2-user@ip-10-0-1-218 ~]$ chmod 400 sam.pem
[ec2-user@ip-10-0-1-218 ~]$ ssh -i sam.pem ec2-user@10.0.1.218
The authenticity of host '10.0.1.218 (10.0.1.218)' can't be established.
ECDSA key fingerprint is SHA256:Y6FPLIZ5IAKtMNwnbL3Yq1SXQPKRyey1HTZPbylOrLY.
ECDSA key fingerprint is MD5:d4:a6:0d:fa:99:92:df:21:ca:36:0f:39:5f:ed:ba:cd.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.1.218' (ECDSA) to the list of known hosts.
Last login: Wed Feb  5 14:18:12 2025 from 223.178.84.112
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-10-0-1-218 ~]$ |
```

Step 6: Secure Your Bastion Host

6.1 Restrict SSH Access

- **Go to Security Group (BastionSG)** → Edit Inbound Rules.
- **Allow SSH only from your IP address (xx.xx.xx.xx/32)** instead of allowing all (0.0.0.0/0)


VPC > Security Groups > sg-0175532fe37fd7693 - BastionSG

Inbound security group rules successfully modified on security group (sg-0175532fe37fd7693 | BastionSG)

Details

sg-0175532fe37fd7693 - BastionSG

Details

Security group name: BastionSG
Security group ID: sg-0175532fe37fd7693
Description: Security group for Bastion host
VPC ID: vpc-034db228650a8b7d3
Owner: 774305605711
Inbound rules count: 1 Permission entry
Outbound rules count: 1 Permission entry

6.2 Disable Password Authentication

1. Edit SSH config:

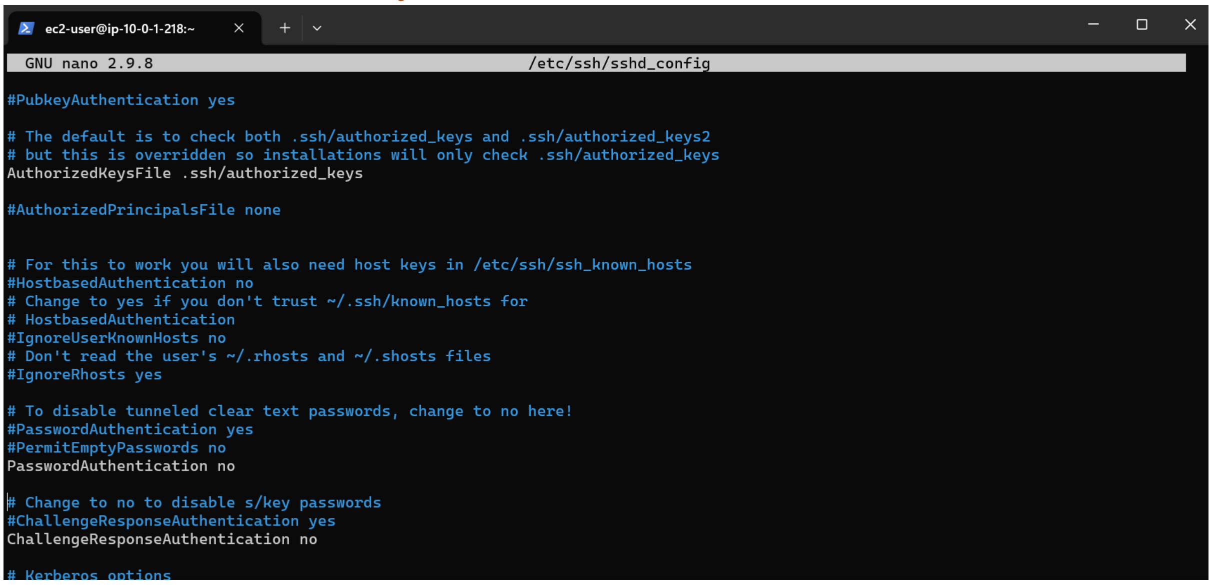
```
sudo nano /etc/ssh/sshd_config
```

2. Find and update these lines:

```
PasswordAuthentication no
PermitRootLogin no
```

1. Restart SSH service:

```
sudo systemctl restart sshd
```



```
  GNU nano 2.9.8                        /etc/ssh/sshd_config

#PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile .ssh/authorized_keys

#AuthorizedPrincipalsFile none


# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no
PasswordAuthentication no

# Change to no to disable s/key passwords
#ChallengeResponseAuthentication yes
ChallengeResponseAuthentication no

# Kerberos options
```

Step 7:
Alternative - Use AWS Systems Manager (SSM) Instead of SSH

1. **Attach SSM Managed Policy to EC2 IAM Role** (AmazonSSMManagedInstanceCore).
2. **Enable SSM Agent** (Pre-installed on Amazon Linux & Ubuntu).
3. Use **AWS Systems Manager > Session Manager** to connect to instances without SSH.

Conclusion

Using a Bastion Host significantly enhances security by acting as a controlled access point to private instances. This setup prevents direct internet exposure, enforces security group rules, and allows monitoring/logging of access.
For even better security, consider eliminating SSH and using AWS Systems Manager (SSM) Session Manager instead.