AWS High-Availability Web Architecture with Bastion Host, ALB and EFS

⚙️ Overview:

Built a secure and highly available AWS architecture where a private Linux EC2 hosts a static website, accessible only through an Application Load Balancer, with administrative access controlled via a bastion host. Shared storage is provided using Amazon EFS.

🏛️ Objectives:

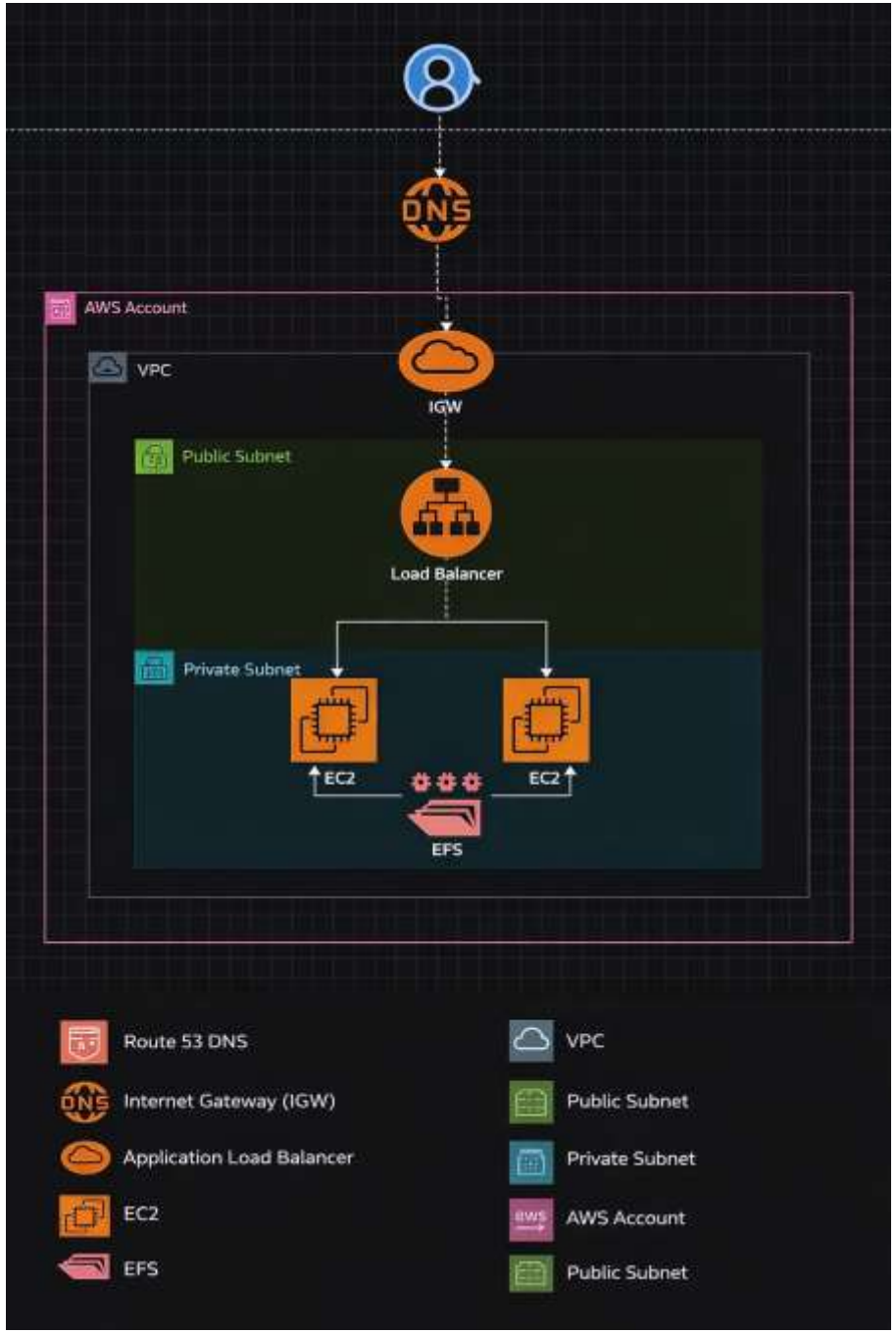
- Secure access to private EC2 instances
- High availability for web applications
- Network isolation using public and private subnets
- Shared and persistent storage.

✨ Key Features:

•Bastion host for secure SSH/RDP access
•Private EC2 with no public IP
•ALB for web traffic routing
•Amazon EFS for shared storage
•VPC-based network isolation

💻 Tools & Technologies

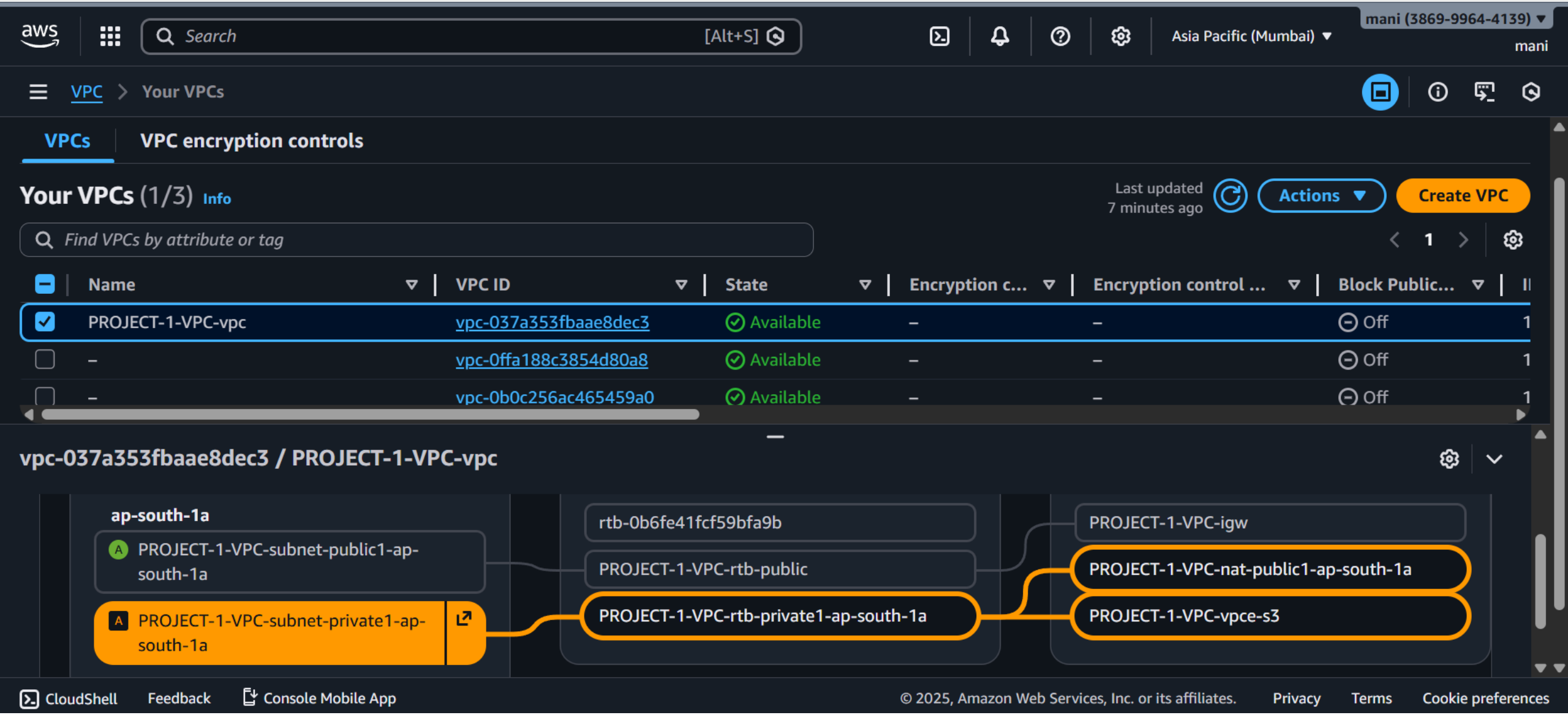
- Amazon EC2, Amazon VPC
- Application Load Balancer
- Amazon EFS
- AWS IAM
- Linux, Windows

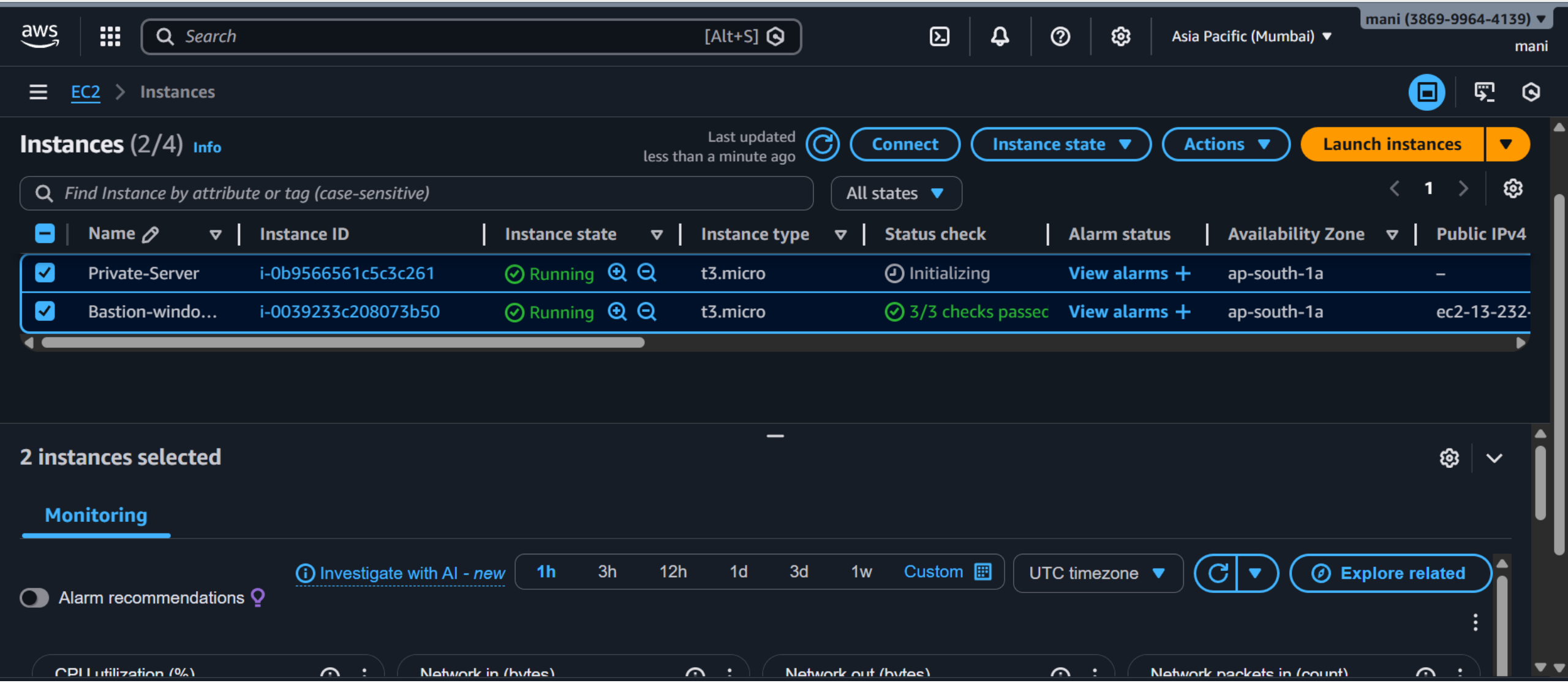


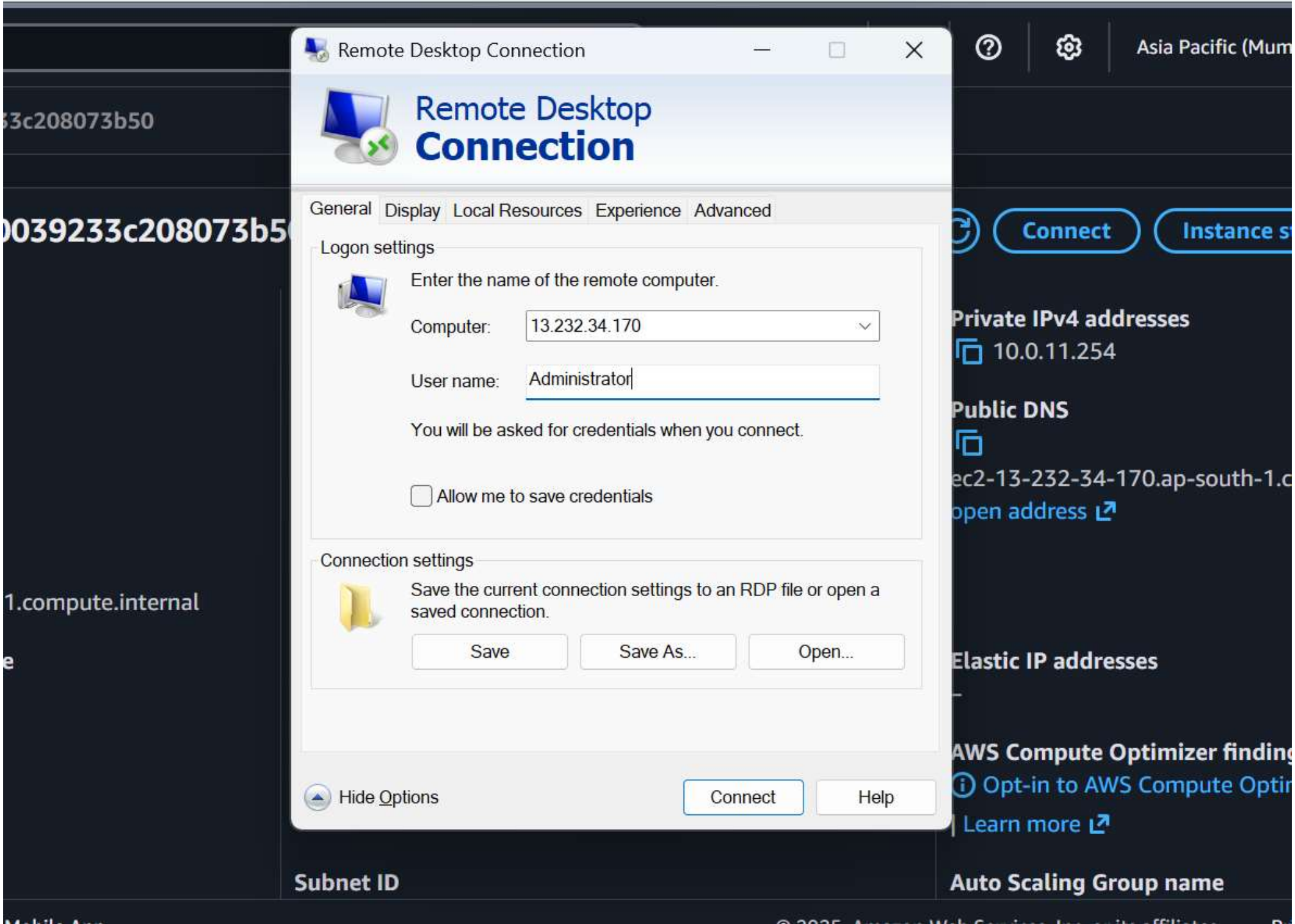
Architecture Flow:

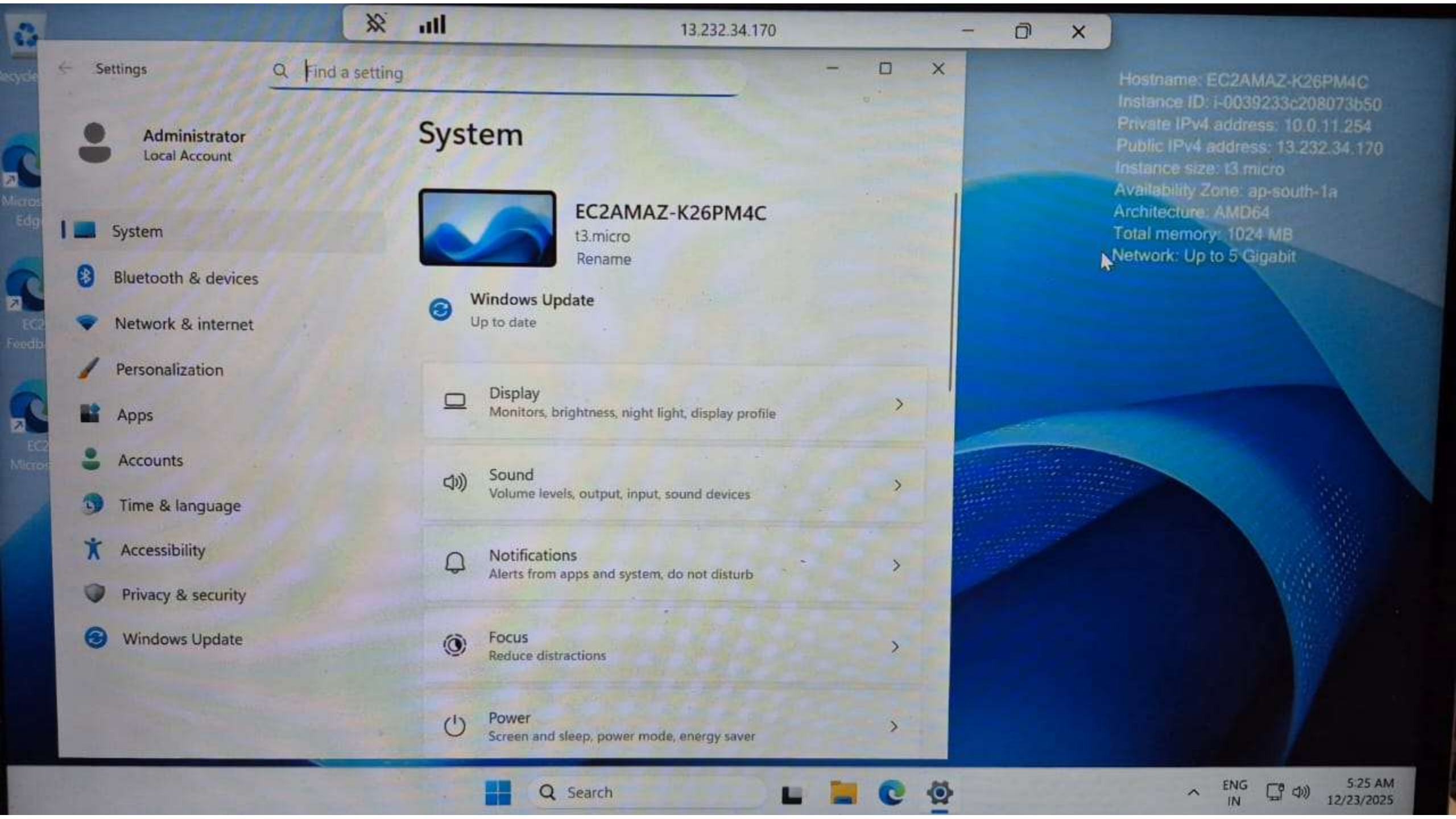
1. User accesses website via ALB DNS
2. ALB routes traffic to private EC2
3. Admin access via bastion host only
4. Files shared using Amazon EFS

- **Work flow:**
- Create a **VPC**
- Create **Public Subnet** and **Private Subnet**
- Attach **Internet Gateway** to VPC
- Create **Public Route Table → IGW**
- Create **Security Groups** (Bastion, ALB, Private EC2)
- Launch **Bastion EC2** in Public Subnet (with Public IP)
- Launch **Private EC2** in Private Subnet (No Public IP)
- Access Private EC2 **via Bastion only**
- Install **Static Website** on Private EC2
- Create **Target Group** and register Private EC2
- Create **Application Load Balancer** in Public Subnets
- Access website using **ALB DNS name**

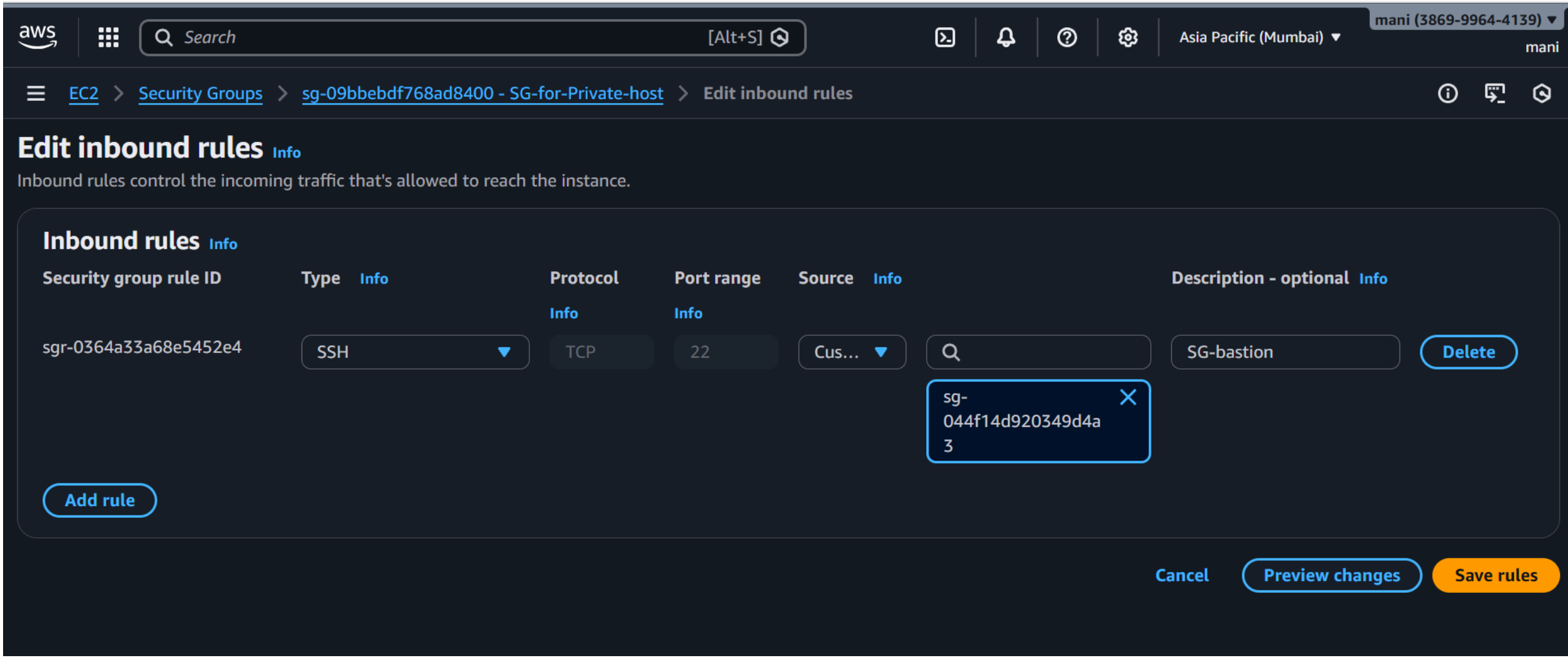

aws
Search
[Alt+S]
Asia Pacific (Mumbai)
mani (3869-9964-4139)
mani
VPC
Your VPCs
VPCs
VPC encryption controls
Your VPCs (1/3) Info
Last updated
7 minutes ago
Actions
Create VPC
Find VPCs by attribute or tag
1
Name
VPC ID
State
Encryption c...
Encryption control ...
Block Public...
PROJECT-1-VPC-vpc
vpc-037a353fbaae8dec3
Available
–
–
Off
–
vpc-0ffa188c3854d80a8
Available
–
–
Off
–
vpc-0b0c256ac465459a0
Available
–
–
Off
vpc-037a353fbaae8dec3 / PROJECT-1-VPC-vpc
ap-south-1a
PROJECT-1-VPC-subnet-public1-ap-south-1a
PROJECT-1-VPC-subnet-private1-ap-south-1a
rtb-0b6fe41fcf59bfa9b
PROJECT-1-VPC-rtb-public
PROJECT-1-VPC-rtb-private1-ap-south-1a
PROJECT-1-VPC-igw
PROJECT-1-VPC-nat-public1-ap-south-1a
PROJECT-1-VPC-vpce-s3
CloudShell
Feedback
Console Mobile App
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Search
[Alt+S]
Asia Pacific (Mumbai)
mani (3869-9964-4139)
mani
EC2
Instances
Instances (2/4) Info
Last updated
less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4
Private-Server
i-0b9566561c5c3c261
Running
t3.micro
Initializing
View alarms
ap-south-1a
–
Bastion-windo...
i-0039233c208073b50
Running
t3.micro
3/3 checks passec
View alarms
ap-south-1a
ec2-13-232-
2 instances selected
Monitoring
Investigate with AI - new
1h
3h
12h
1d
3d
1w
Custom
UTC timezone
Explore related
Alarm recommendations
CPU utilization (%)
Network in (bytes)
Network out (bytes)
Network packets in (count)

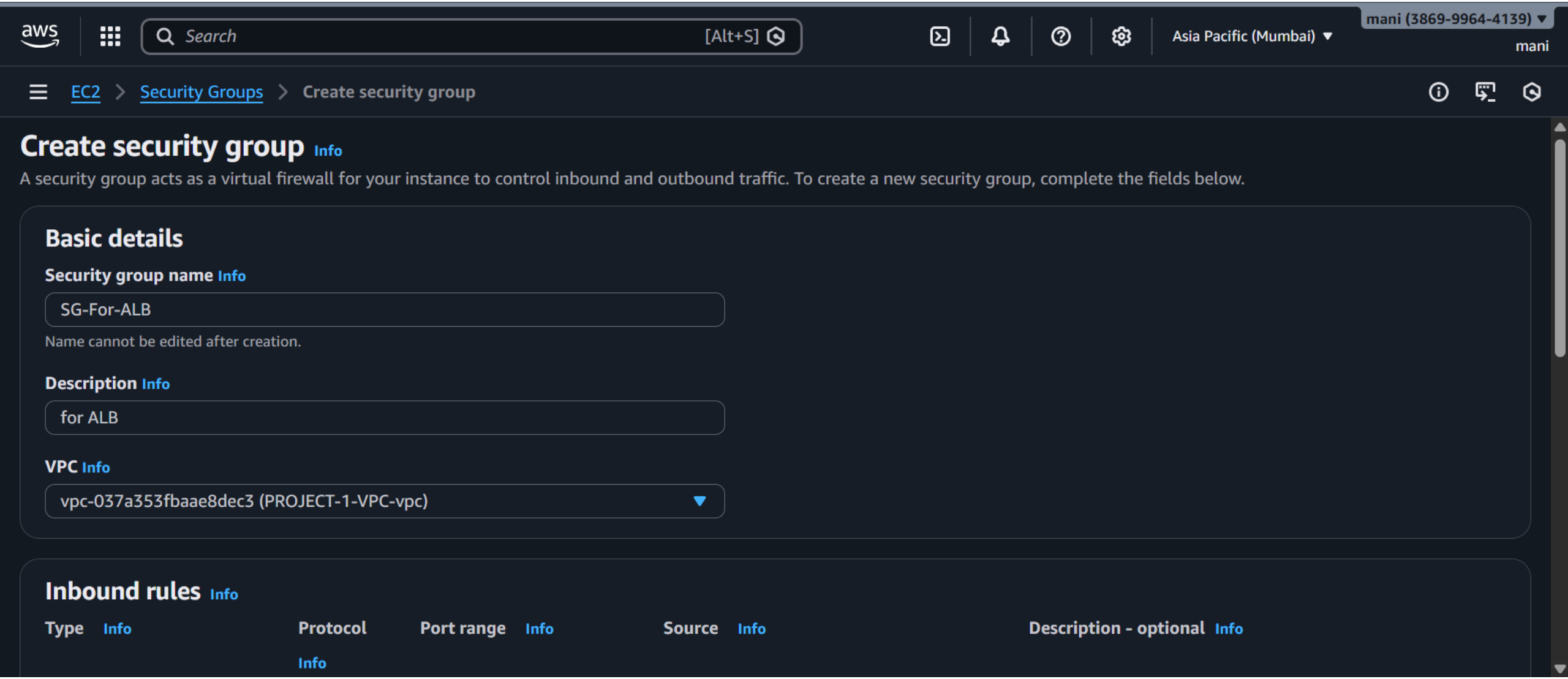

Remote Desktop Connection
Remote Desktop
Connection
General
Display
Local Resources
Experience
Advanced
Logon settings
Enter the name of the remote computer.
Computer:
13.232.34.170
User name:
Administrator
You will be asked for credentials when you connect.
Allow me to save credentials
Connection settings
Save the current connection settings to an RDP file or open a saved connection.
Save
Save As...
Open...
Hide Options
Connect
Help
Connect
Private IPv4 addresses
10.0.11.254
Public DNS
ec2-13-232-34-170.ap-south-1.c
open address
Elastic IP addresses
AWS Compute Optimizer finding
Opt-in to AWS Compute Optir
Learn more
Subnet ID
Auto Scaling Group name


13.232.34.170
Settings
Find a setting
Administrator
Local Account
System
Bluetooth & devices
Network & internet
Personalization
Apps
Accounts
Time & language
Accessibility
Privacy & security
Windows Update
System
EC2AMAZ-K26PM4C
t3.micro
Rename
Windows Update
Up to date
Display
Monitors, brightness, night light, display profile
Sound
Volume levels, output, input, sound devices
Notifications
Alerts from apps and system, do not disturb
Focus
Reduce distractions
Power
Screen and sleep, power mode, energy saver
Hostname: EC2AMAZ-K26PM4C
Instance ID: i-0039233c208073b50
Private IPv4 address: 10.0.11.254
Public IPv4 address: 13.232.34.170
Instance size: t3.micro
Availability Zone: ap-south-1a
Architecture: AMD64
Total memory: 1024 MB
Network: Up to 5 Gigabit
Search
ENG
IN
5:25 AM
12/23/2025

Add SG of bastion host in the inbound rule of private server



SG for ALB


aws
Search
[Alt+S]
Asia Pacific (Mumbai)
mani (3869-9964-4139)
mani
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
SG-For-ALB
Name cannot be edited after creation.
Description Info
for ALB
VPC Info
vpc-037a353fbaae8dec3 (PROJECT-1-VPC-vpc)
Inbound rules Info
Type Info
Protocol Info
Port range Info
Source Info
Description - optional Info

Inbound rules Info

Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info	
HTTP	TCP	80	An... 0.0.0.0/0	to access the website	Delete

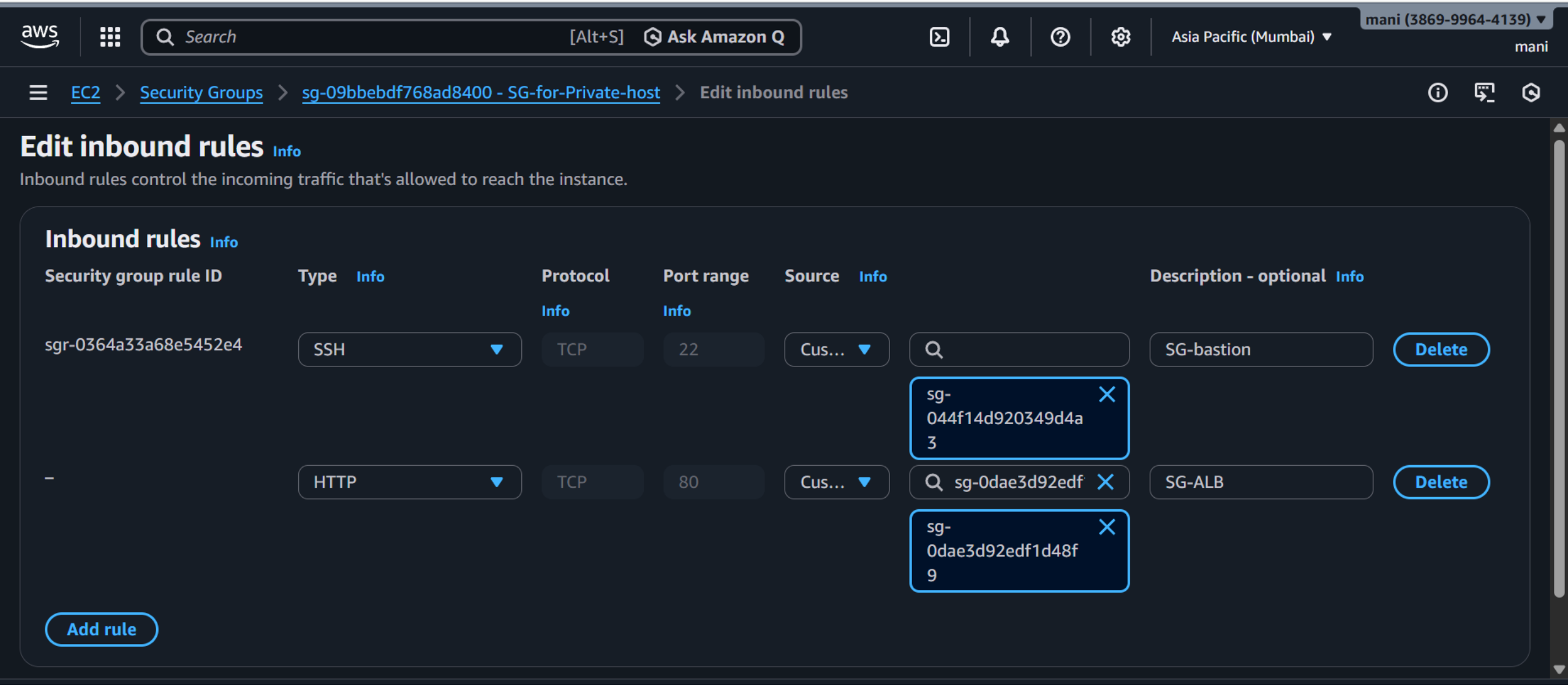
Add rule

Outbound rules Info

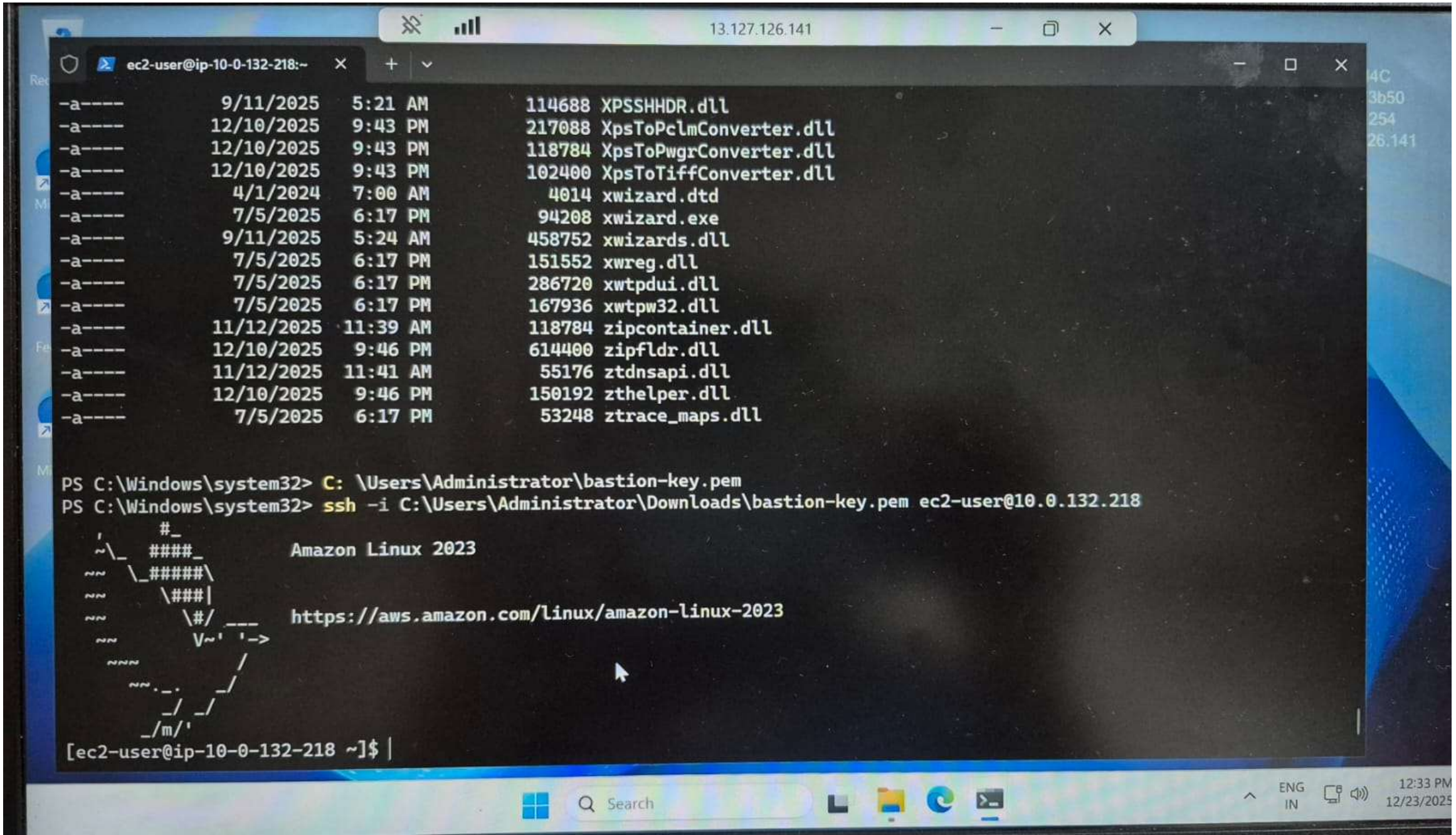
Type Info	Protocol Info	Port range Info	Destination Info	Description - optional Info	
All traffic	All	All	Cus... 0.0.0.0/0		Delete

Add rule

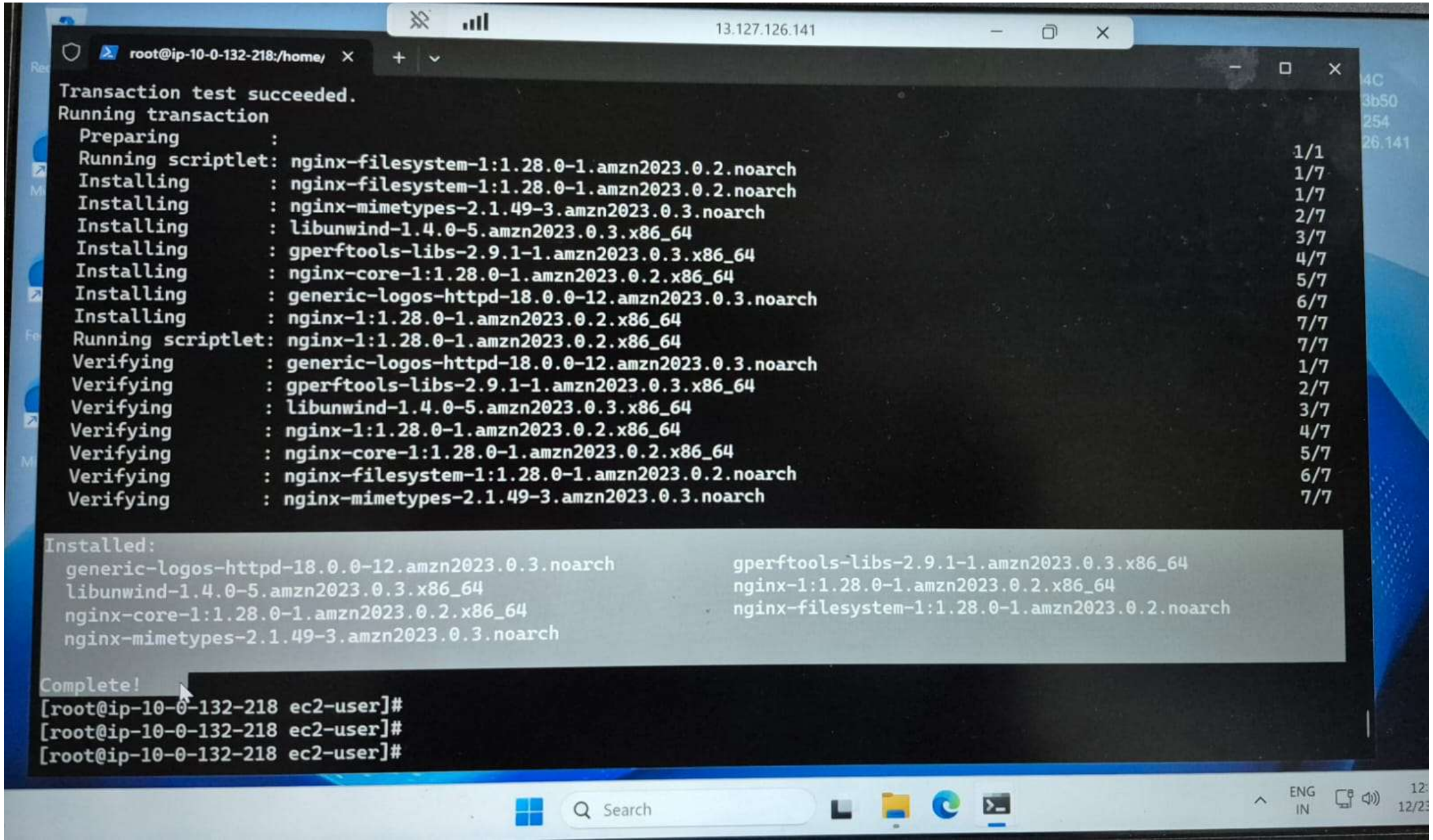
Add SG of ALB in private host



Connect to private-Linux Server



Install nginx



```
root@ip-10-0-132-218:/home/
  Installing       : nginx-filesystem-1:1.28.0-1.amzn2023.0.2.noarch                1/7
  Installing       : nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch                   2/7
  Installing       : libunwind-1.4.0-5.amzn2023.0.3.x86_64                          3/7
  Installing       : gperftools-libs-2.9.1-1.amzn2023.0.3.x86_64                    4/7
  Installing       : nginx-core-1:1.28.0-1.amzn2023.0.2.x86_64                      5/7
  Installing       : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch              6/7
  Installing       : nginx-1:1.28.0-1.amzn2023.0.2.x86_64                           7/7
  Running scriptlet: nginx-1:1.28.0-1.amzn2023.0.2.x86_64                           7/7
  Verifying        : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch              1/7
  Verifying        : gperftools-libs-2.9.1-1.amzn2023.0.3.x86_64                    2/7
  Verifying        : libunwind-1.4.0-5.amzn2023.0.3.x86_64                          3/7
  Verifying        : nginx-1:1.28.0-1.amzn2023.0.2.x86_64                           4/7
  Verifying        : nginx-core-1:1.28.0-1.amzn2023.0.2.x86_64                      5/7
  Verifying        : nginx-filesystem-1:1.28.0-1.amzn2023.0.2.noarch                6/7
  Verifying        : nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch                   7/7

Installed:
  generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch     gperftools-libs-2.9.1-1.amzn2023.0.3.x86_64
  libunwind-1.4.0-5.amzn2023.0.3.x86_64                 nginx-1:1.28.0-1.amzn2023.0.2.x86_64
  nginx-core-1:1.28.0-1.amzn2023.0.2.x86_64             nginx-filesystem-1:1.28.0-1.amzn2023.0.2.noarch
  nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch

Complete!
[root@ip-10-0-132-218 ec2-user]#
[root@ip-10-0-132-218 ec2-user]#
[root@ip-10-0-132-218 ec2-user]# systemctl start nginx
[root@ip-10-0-132-218 ec2-user]# systemctl enable nginx
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /usr/lib/systemd/system/nginx.service.
[root@ip-10-0-132-218 ec2-user]#
[root@ip-10-0-132-218 ec2-user]#
```

Search

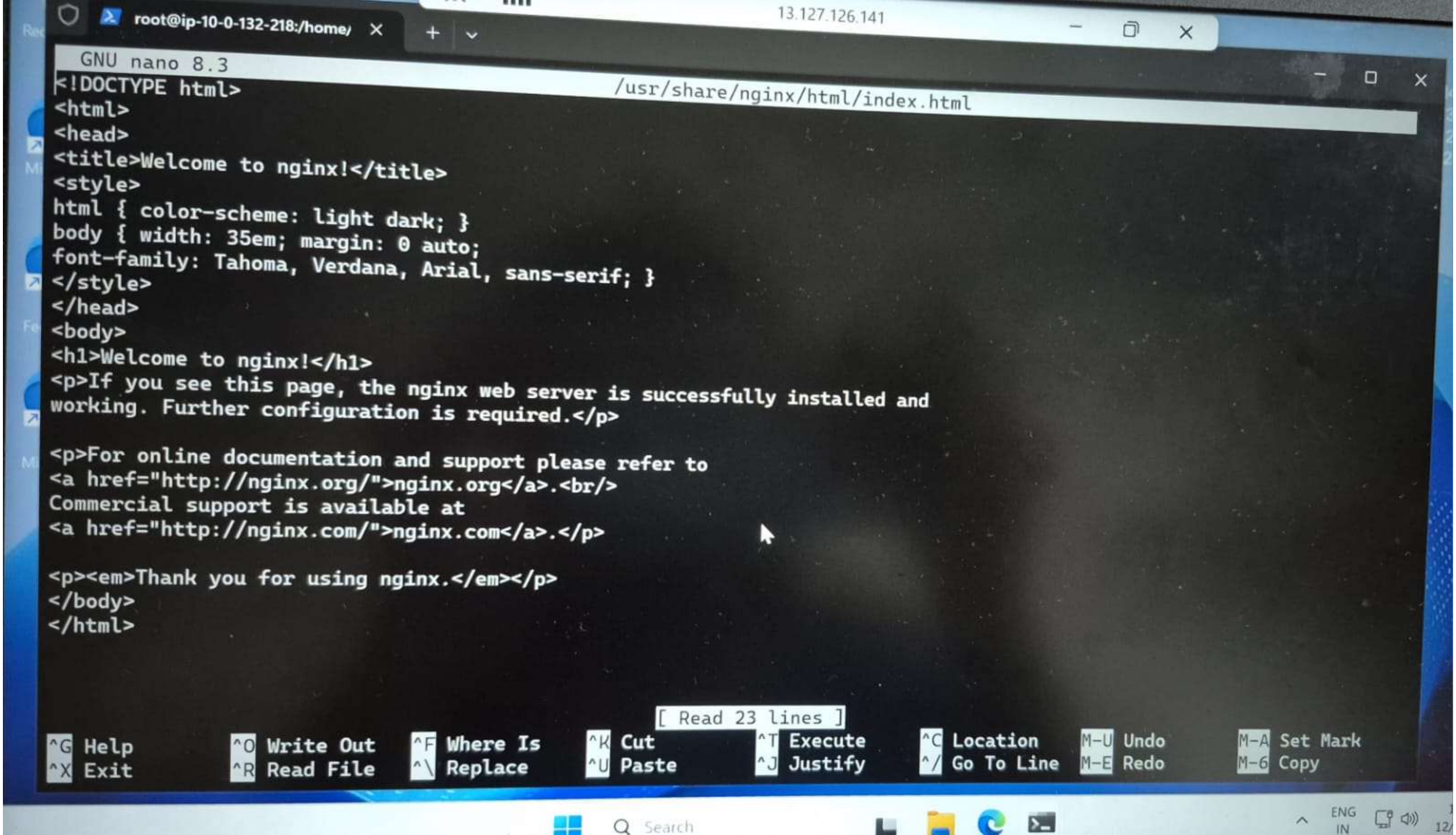
ENG
IN

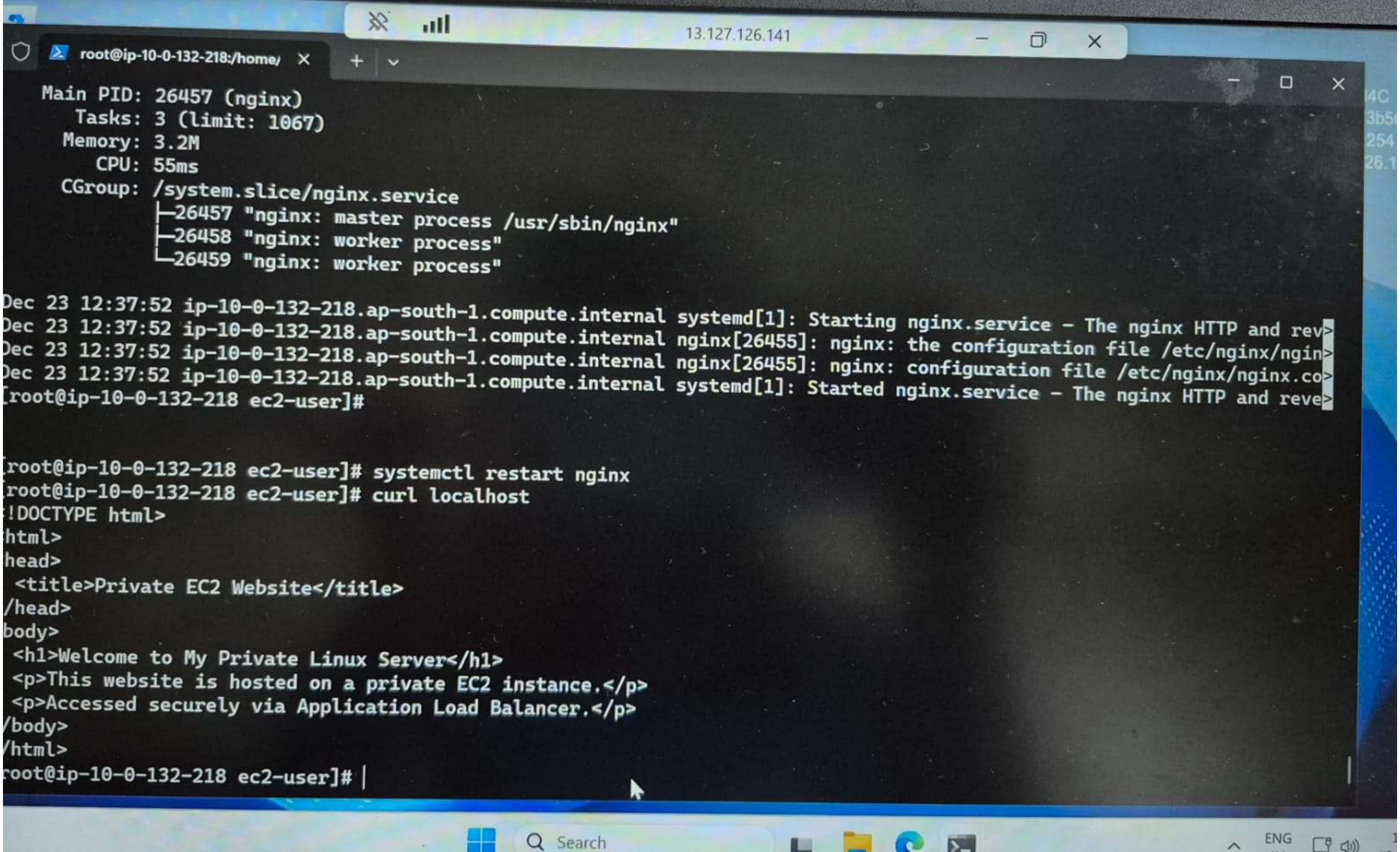
12:38 PM
12/23/2025

```
root@ip-10-0-132-218:/home/
  generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch      gperftools-libs-2.9.1-1.amzn2023.0.3.x86_64
  libunwind-1.4.0-5.amzn2023.0.3.x86_64                  nginx-1:1.28.0-1.amzn2023.0.2.x86_64
  nginx-core-1:1.28.0-1.amzn2023.0.2.x86_64              nginx-filesystem-1:1.28.0-1.amzn2023.0.2.noarch
  nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch

Complete!
[root@ip-10-0-132-218 ec2-user]#
[root@ip-10-0-132-218 ec2-user]#
[root@ip-10-0-132-218 ec2-user]# systemctl start nginx
[root@ip-10-0-132-218 ec2-user]# systemctl enable nginx
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /usr/lib/systemd/system/nginx.service.
[root@ip-10-0-132-218 ec2-user]#
[root@ip-10-0-132-218 ec2-user]# systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-23 12:37:52 UTC; 57s ago
   Main PID: 26457 (nginx)
      Tasks: 3 (limit: 1067)
     Memory: 3.2M
        CPU: 55ms
     CGroup: /system.slice/nginx.service
             ├─26457 "nginx: master process /usr/sbin/nginx"
             ├─26458 "nginx: worker process"
             └─26459 "nginx: worker process"

Dec 23 12:37:52 ip-10-0-132-218.ap-south-1.compute.internal systemd[1]: Starting nginx.service - The nginx HTTP and rev>
Dec 23 12:37:52 ip-10-0-132-218.ap-south-1.compute.internal nginx[26455]: nginx: the configuration file /etc/nginx/ngin>
Dec 23 12:37:52 ip-10-0-132-218.ap-south-1.compute.internal nginx[26455]: nginx: configuration file /etc/nginx/nginx.co>
Dec 23 12:37:52 ip-10-0-132-218.ap-south-1.compute.internal systemd[1]: Started nginx.service - The nginx HTTP and reve>
lines 1-16/16 (END)
```


root@ip-10-0-132-218:/home/
13.127.126.141
GNU nano 8.3
/usr/share/nginx/html/index.html
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.
For online documentation and support please refer to
nginx.org.
Commercial support is available at
nginx.com.
Thank you for using nginx.
</body>
</html>
[Read 23 lines]
^G Help
^X Exit
^O Write Out
^R Read File
^F Where Is
^\ Replace
^K Cut
^U Paste
^T Execute
^J Justify
^C Location
^/ Go To Line
M-U Undo
M-E Redo
M-A Set Mark
M-6 Copy
Search
ENG
IN


13.127.126.141
root@ip-10-0-132-218:/home/
Main PID: 26457 (nginx)
Tasks: 3 (limit: 1067)
Memory: 3.2M
CPU: 55ms
CGroup: /system.slice/nginx.service
├─26457 "nginx: master process /usr/sbin/nginx"
├─26458 "nginx: worker process"
└─26459 "nginx: worker process"
Dec 23 12:37:52 ip-10-0-132-218.ap-south-1.compute.internal systemd[1]: Starting nginx.service - The nginx HTTP and rev>
Dec 23 12:37:52 ip-10-0-132-218.ap-south-1.compute.internal nginx[26455]: nginx: the configuration file /etc/nginx/ngin>
Dec 23 12:37:52 ip-10-0-132-218.ap-south-1.compute.internal nginx[26455]: nginx: configuration file /etc/nginx/nginx.co>
Dec 23 12:37:52 ip-10-0-132-218.ap-south-1.compute.internal systemd[1]: Started nginx.service - The nginx HTTP and reve>
[root@ip-10-0-132-218 ec2-user]#
[root@ip-10-0-132-218 ec2-user]# systemctl restart nginx
[root@ip-10-0-132-218 ec2-user]# curl localhost
<!DOCTYPE html>
<html>
<head>
<title>Private EC2 Website</title>
</head>
<body>
<h1>Welcome to My Private Linux Server</h1>
This website is hosted on a private EC2 instance.
Accessed securely via Application Load Balancer.
</body>
</html>
[root@ip-10-0-132-218 ec2-user]#
Search
ENG

Edit inbound rules Info

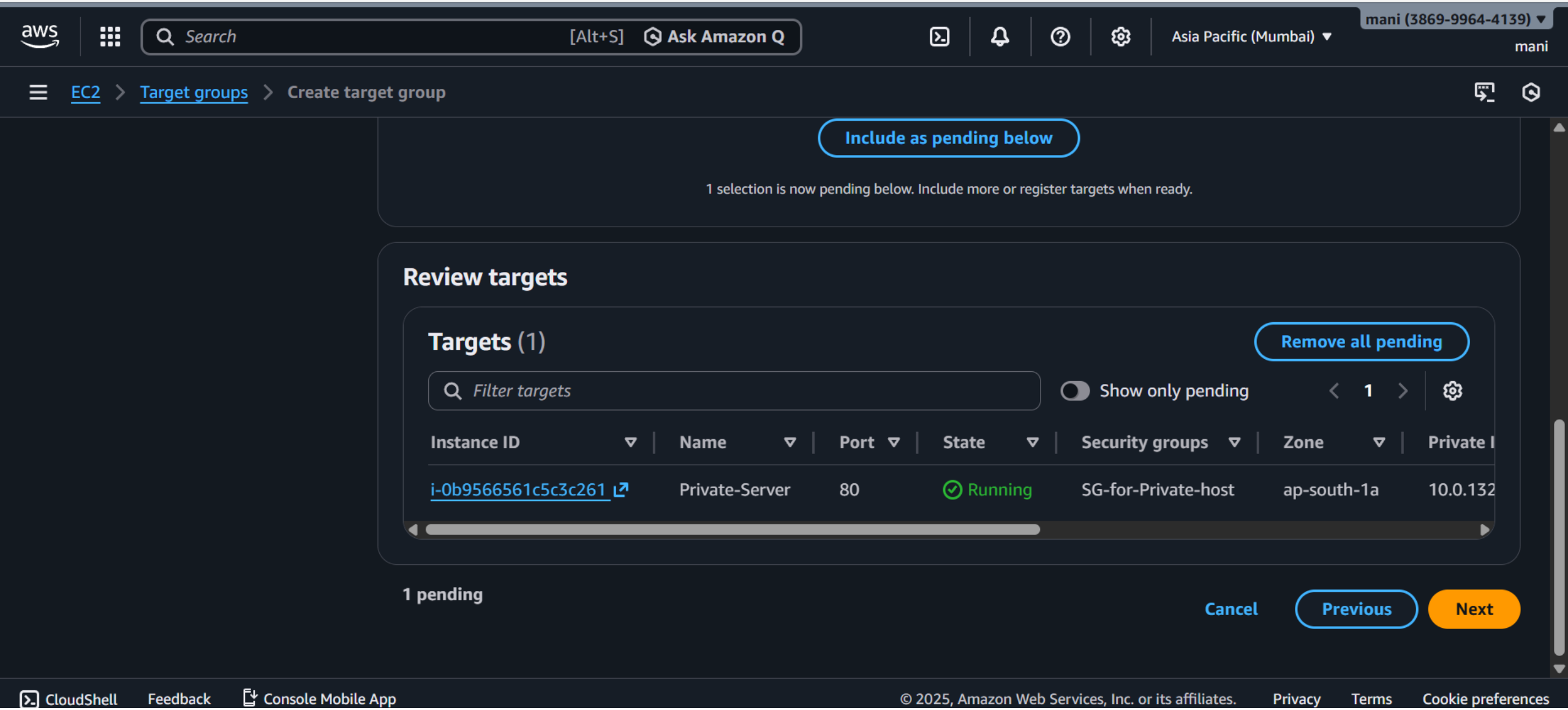
Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-0364a33a68e5452e4	SSH	TCP	22	Cus...	sg-044f14d920349d4a3	SG-bastion	Delete
sgr-0b8172b95128eb7cd	HTTP	TCP	80	Cus...	sg-0dae3d92edf1d48f9	SG-ALB	Delete

Add rule

Create Target Group:



Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores
Auto Scaling
Auto Scaling Groups
Settings

Successfully created the target group: **Private-ALB-TG**. Anomaly detection is automatically applied to all registered targets. Results can be viewed in the **Targets** tab.

Give feedback

Private-ALB-TG

Actions

Details

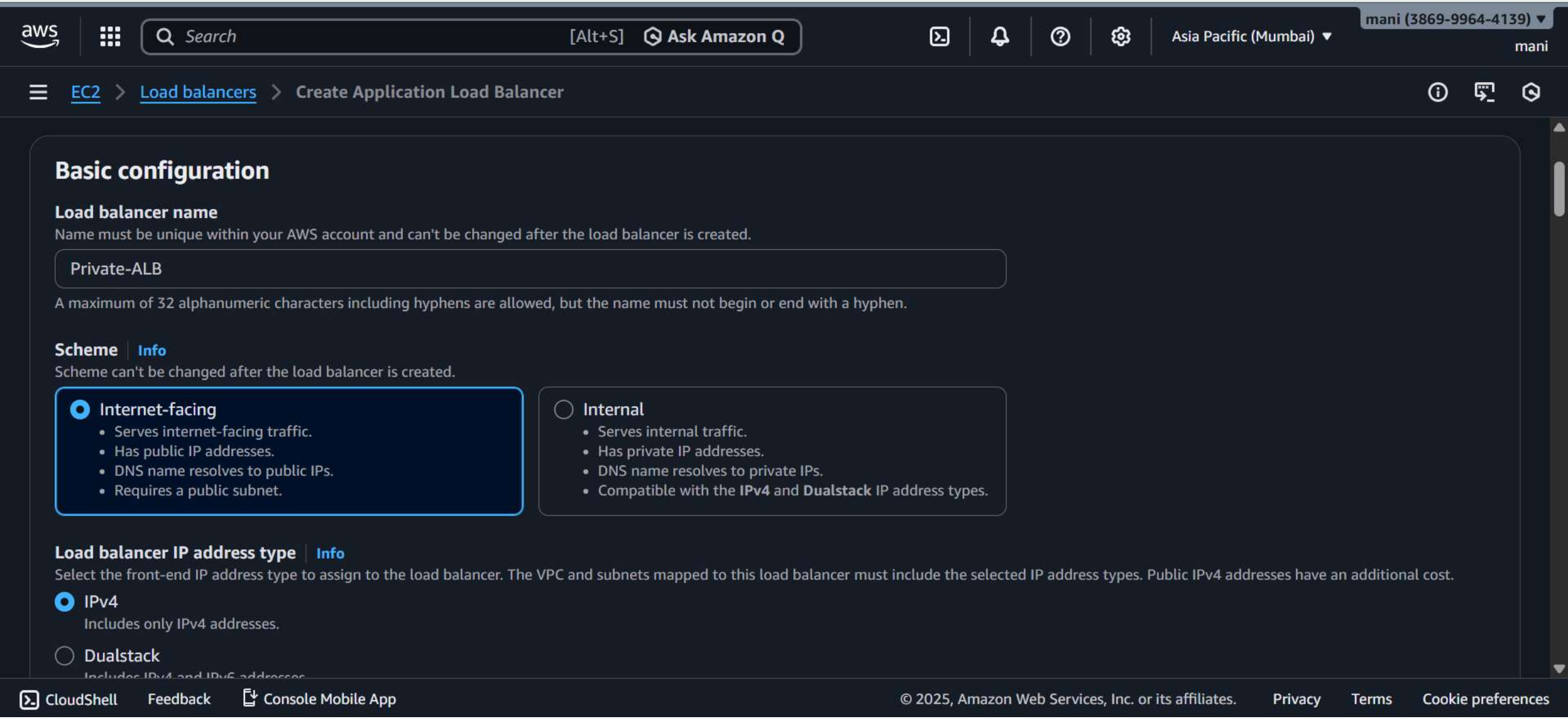
arn:aws:elasticloadbalancing:ap-south-1:386999644139:targetgroup/Private-ALB-TG/2c6d1d0ea2da3e54

Target type	Protocol : Port	Protocol version	VPC
Instance	HTTP: 80	HTTP1	vpc-037a353fbaae8dec3

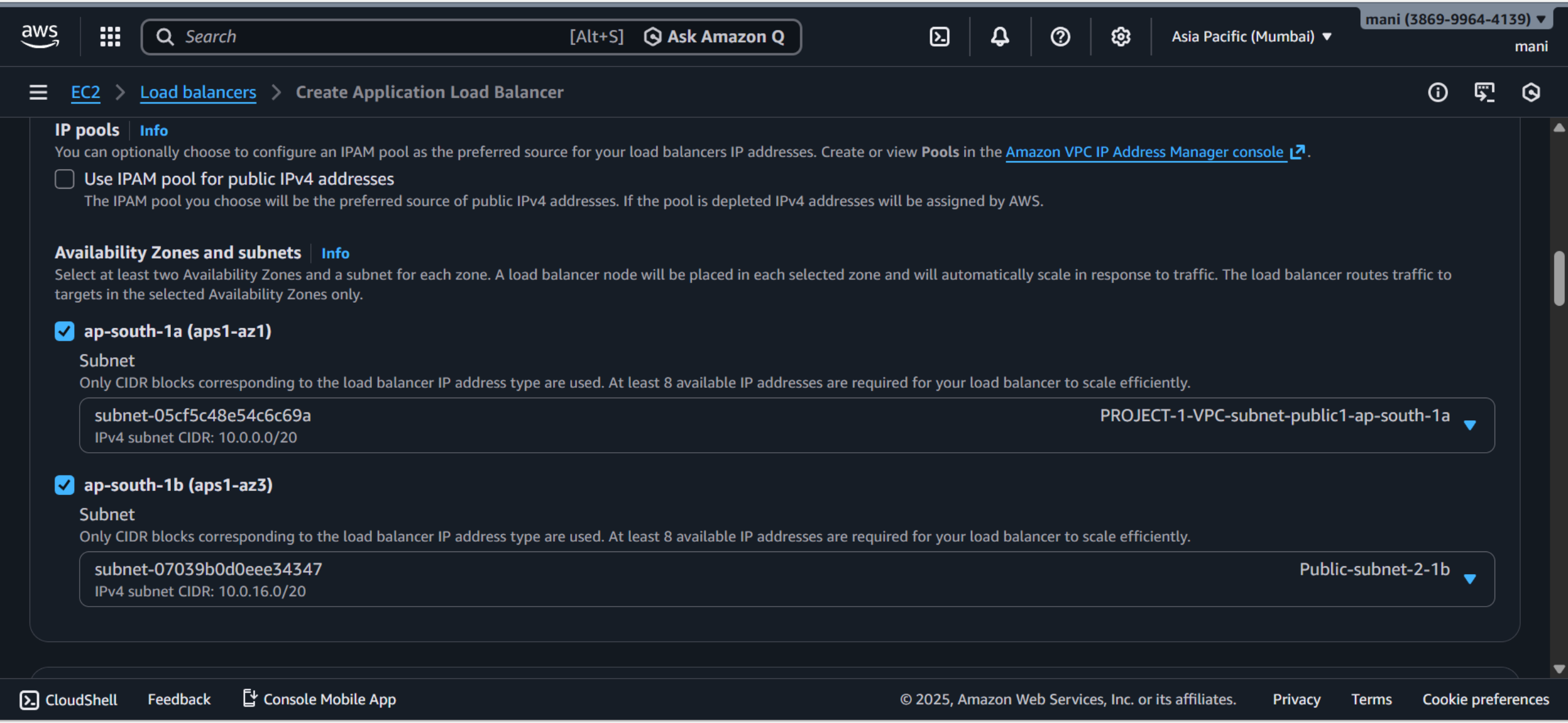
IP address type	Load balancer
IPv4	None associated

1	0	0	1	0	0
Total targets	Healthy	Unhealthy	Unused	Initial	Draining
	0 Anomalous				

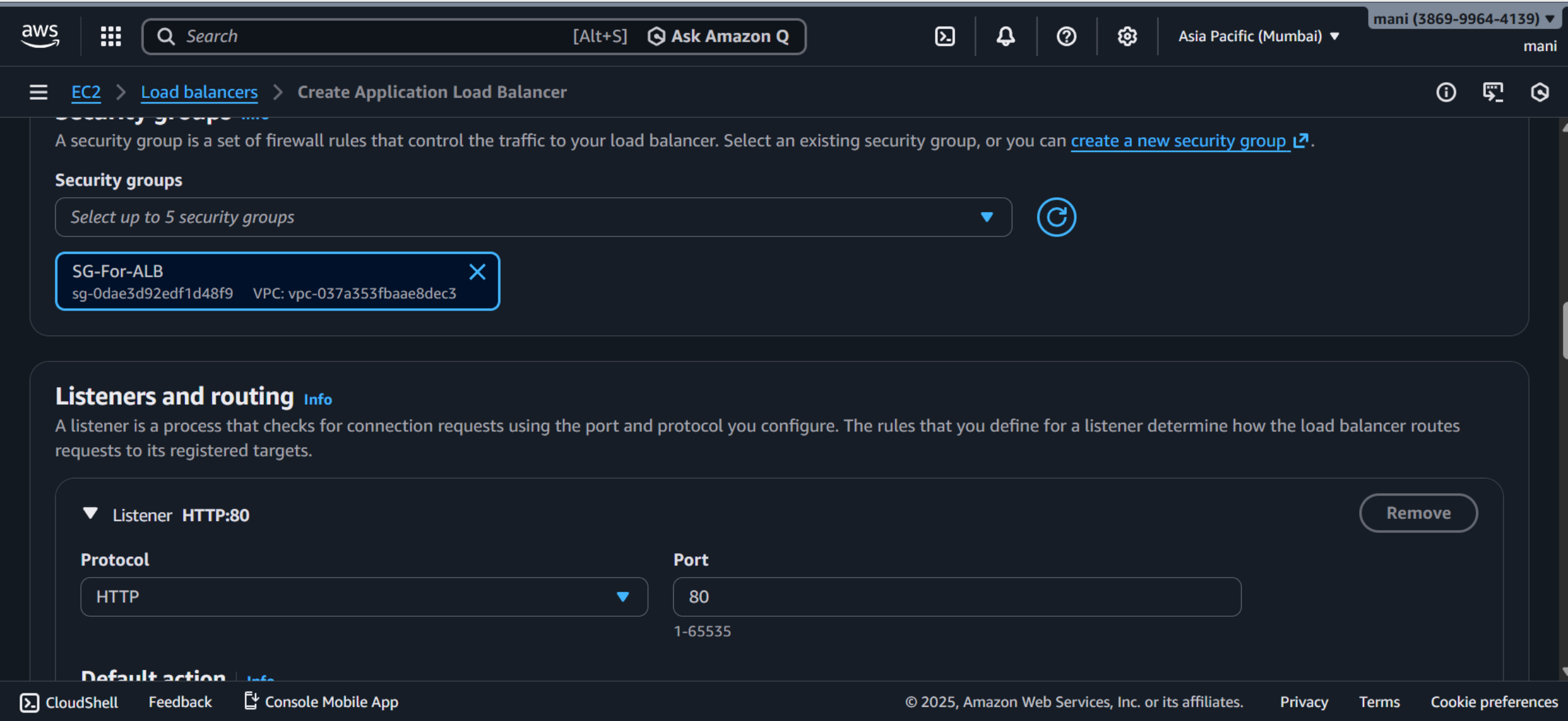
Create ALB:


aws
Search
[Alt+S]
Ask Amazon Q
Asia Pacific (Mumbai)
mani (3869-9964-4139)
mani
EC2
Load balancers
Create Application Load Balancer
Basic configuration
Load balancer name
Name must be unique within your AWS account and can't be changed after the load balancer is created.
Private-ALB
A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.
Scheme
Info
Scheme can't be changed after the load balancer is created.
Internet-facing
Serves internet-facing traffic.
Has public IP addresses.
DNS name resolves to public IPs.
Requires a public subnet.
Internal
Serves internal traffic.
Has private IP addresses.
DNS name resolves to private IPs.
Compatible with the IPv4 and Dualstack IP address types.
Load balancer IP address type
Info
Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses have an additional cost.
IPv4
Includes only IPv4 addresses.
Dualstack
CloudShell
Feedback
Console Mobile App
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Choose two public subnets in different Azs:



Choose the SG of ALB:



Listener **HTTP:80** Remove

Protocol

HTTP

Port

80

1-65535

Default action Info

The default action is used if no other rules apply. Choose the default action for traffic on this listener.

Routing action

- (•) Forward to target groups
- ( ) Redirect to URL
- ( ) Return fixed response

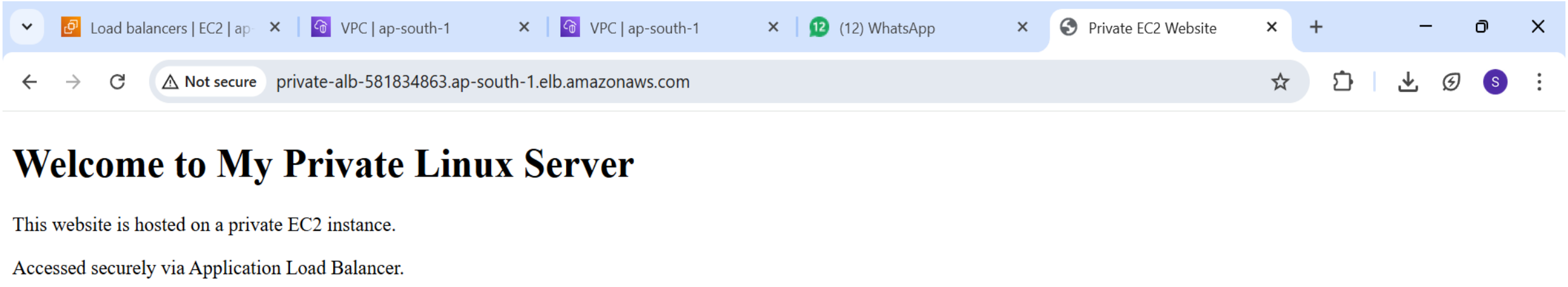
Forward to target group Info

Choose a target group and specify routing weight or create target group.

Target group		Weight	Percent
Private-ALB-TG Target type: Instance, IPv4 | Target stickiness: Off	HTTP	1 0-999	100%

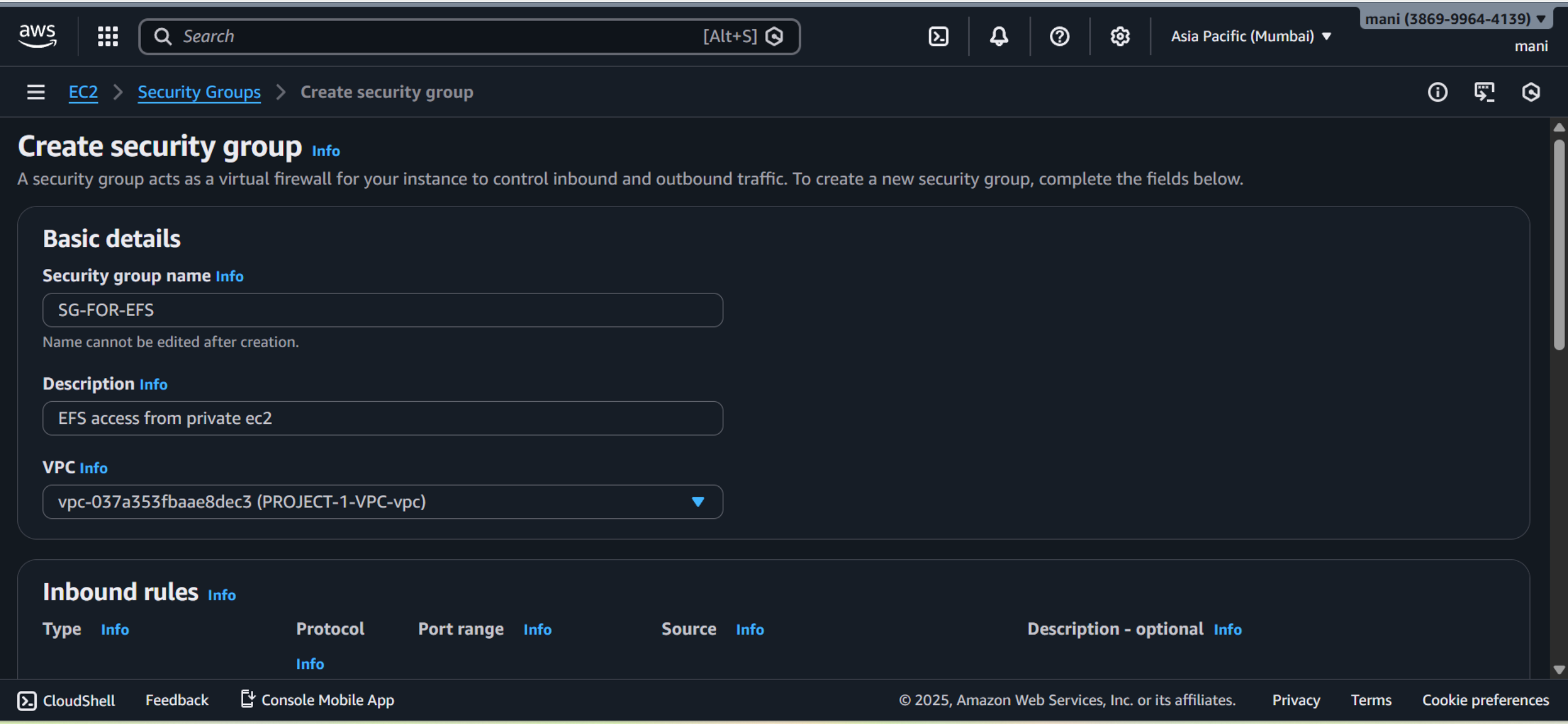
+ Add target group

Copy the DNS of ALB into the Browser:

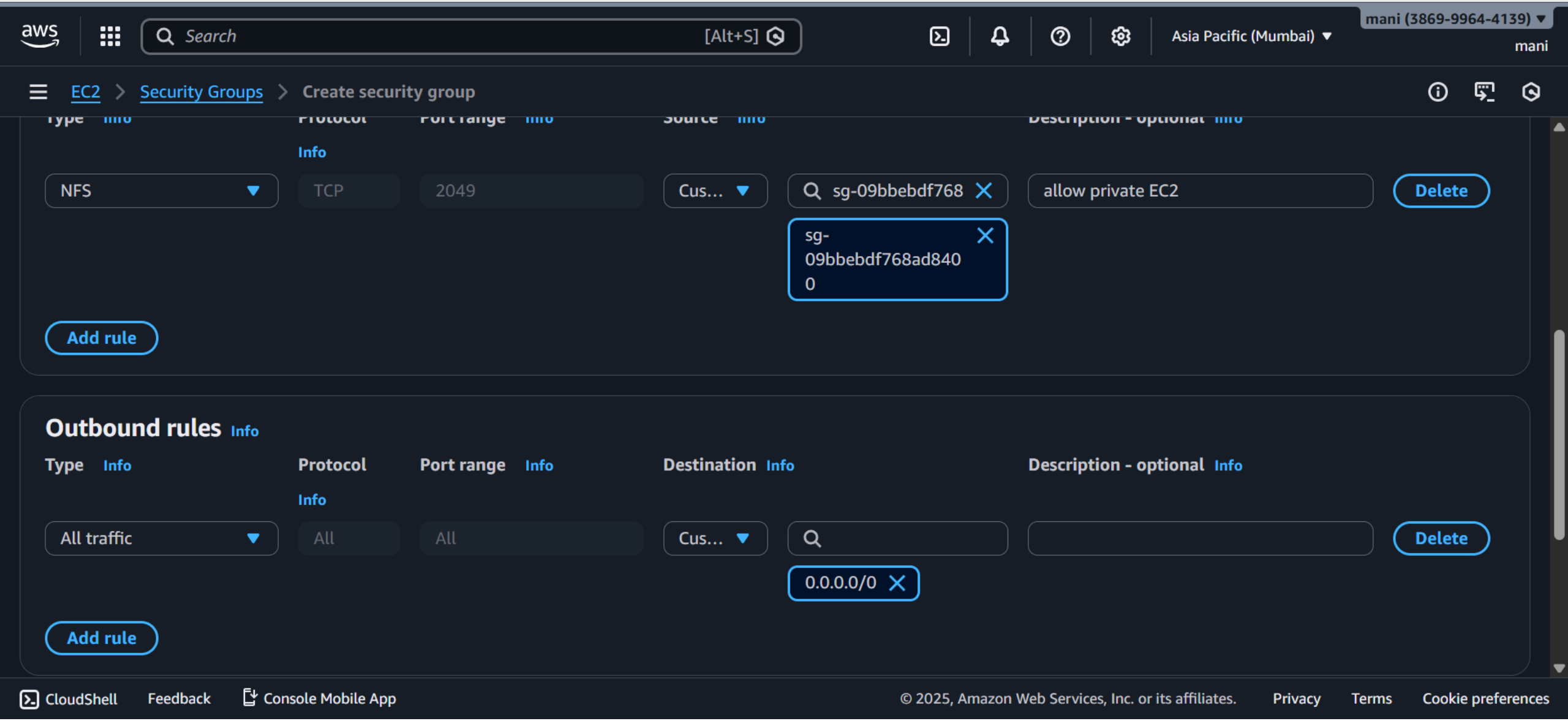


Mount EFS on private subnet:

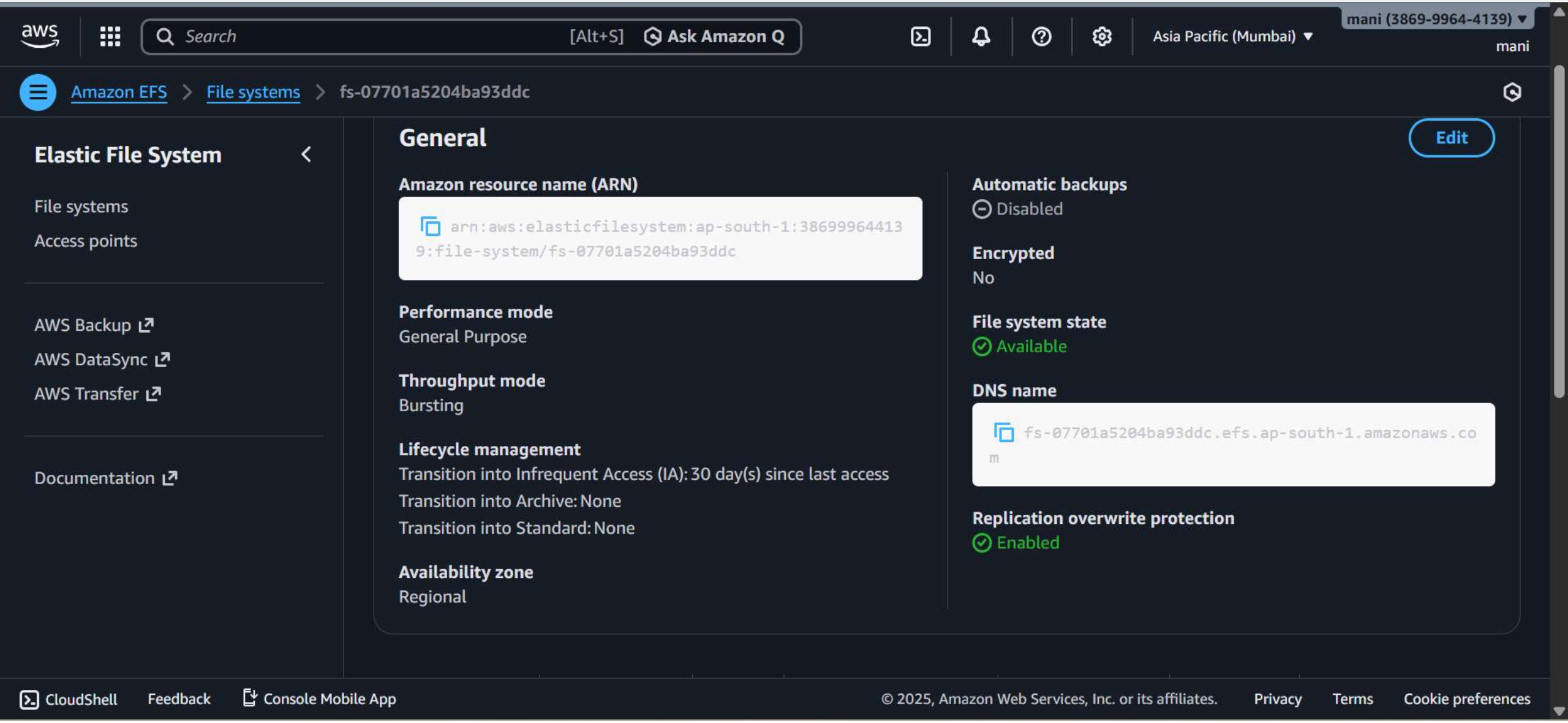
Create SG for EFS

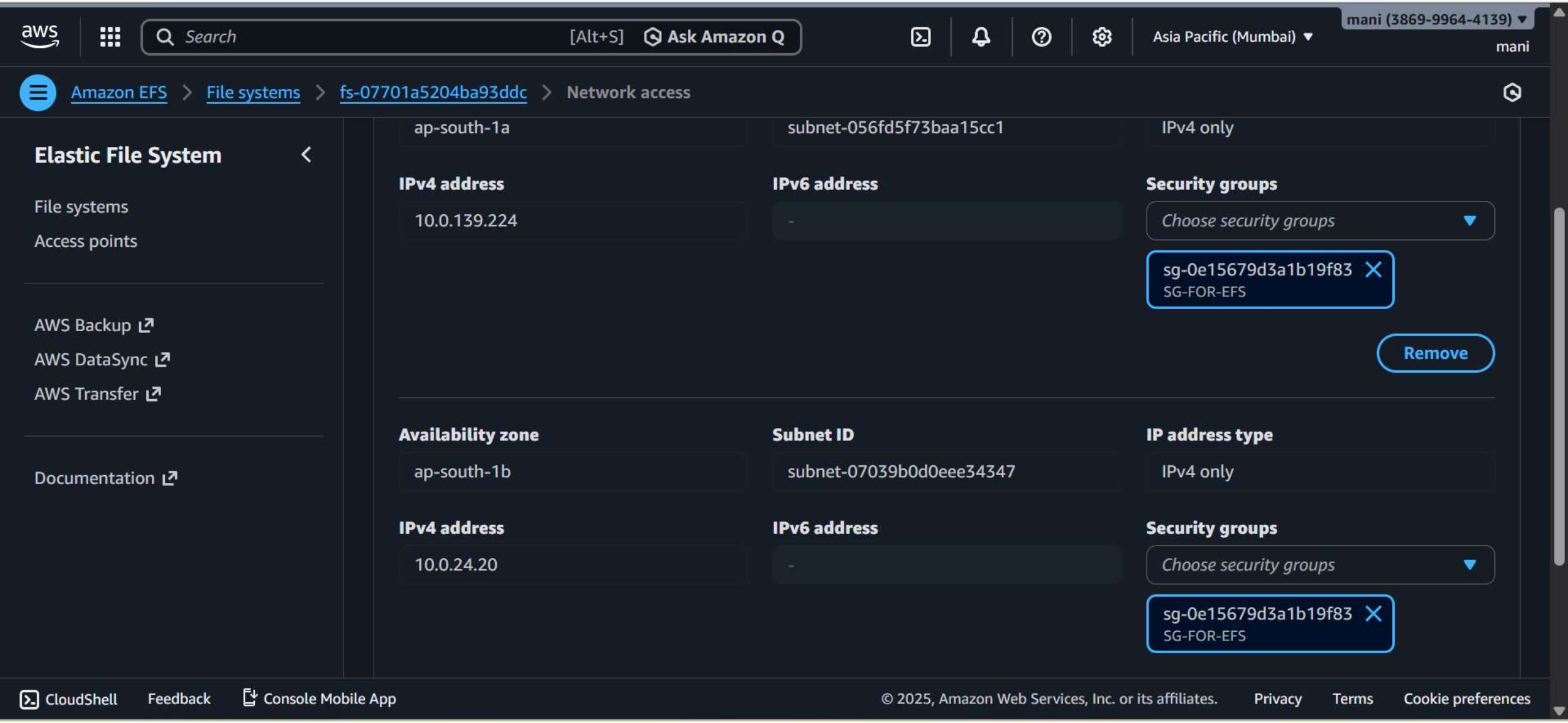


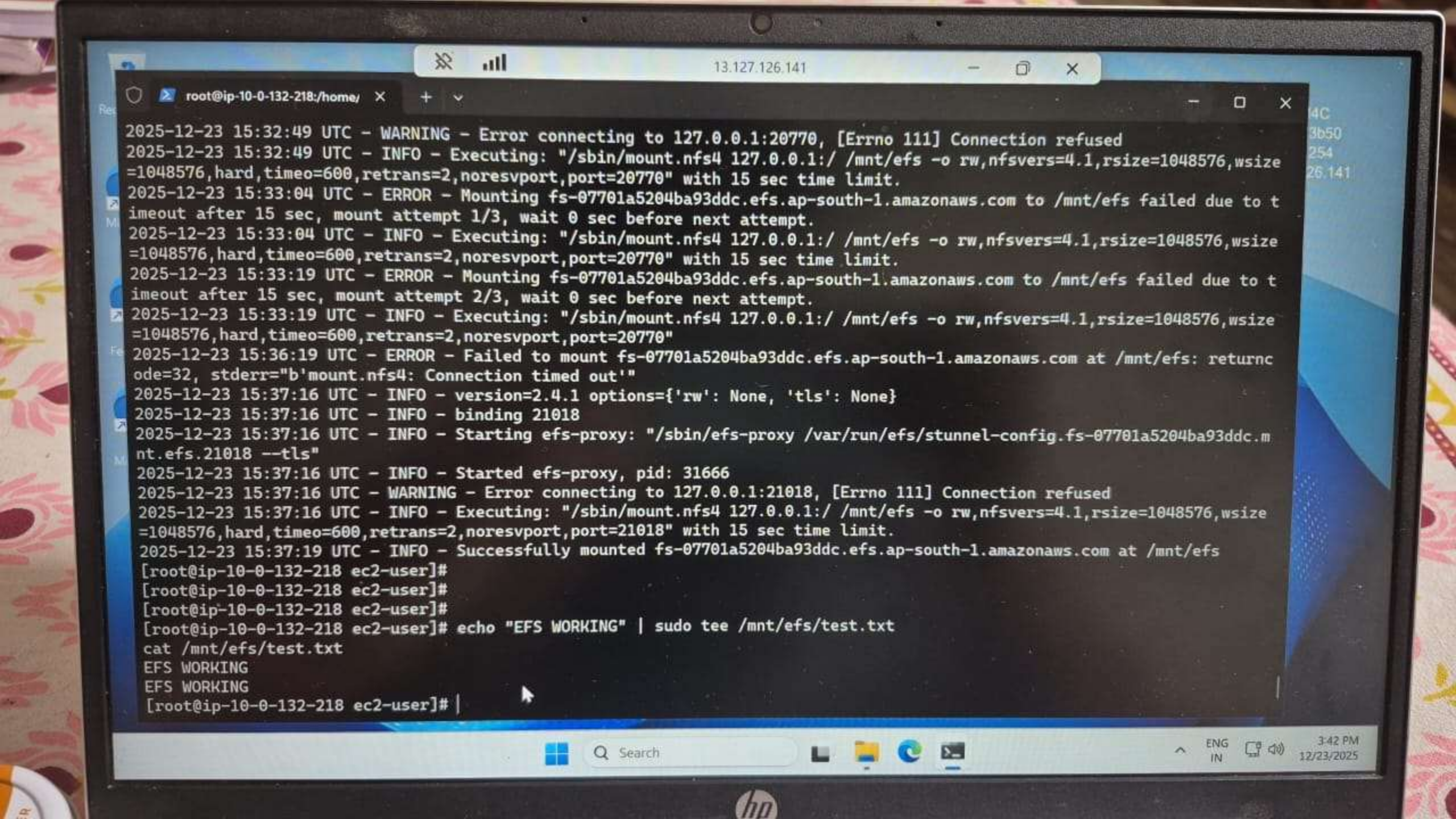
Allow NFS port, source: sg for private instance:



Create file system (EFS):




aws
Search
[Alt+S]
Ask Amazon Q
Asia Pacific (Mumbai)
mani (3869-9964-4139)
mani
Amazon EFS
File systems
fs-07701a5204ba93ddc
Network access
Elastic File System
File systems
Access points
AWS Backup
AWS DataSync
AWS Transfer
Documentation
ap-south-1a
subnet-056fd5f73baa15cc1
IPv4 only
IPv4 address
10.0.139.224
IPv6 address
-
Security groups
Choose security groups
sg-0e15679d3a1b19f83
SG-FOR-EFS
Remove
Availability zone
ap-south-1b
Subnet ID
subnet-07039b0d0eee34347
IP address type
IPv4 only
IPv4 address
10.0.24.20
IPv6 address
-
Security groups
Choose security groups
sg-0e15679d3a1b19f83
SG-FOR-EFS
CloudShell
Feedback
Console Mobile App
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


13.127.126.141
root@ip-10-0-132-218:/home/
2025-12-23 15:32:49 UTC - WARNING - Error connecting to 127.0.0.1:20770, [Errno 111] Connection refused
2025-12-23 15:32:49 UTC - INFO - Executing: "/sbin/mount.nfs4 127.0.0.1:/ /mnt/efs -o rw,nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport,port=20770" with 15 sec time limit.
2025-12-23 15:33:04 UTC - ERROR - Mounting fs-07701a5204ba93ddc.efs.ap-south-1.amazonaws.com to /mnt/efs failed due to timeout after 15 sec, mount attempt 1/3, wait 0 sec before next attempt.
2025-12-23 15:33:04 UTC - INFO - Executing: "/sbin/mount.nfs4 127.0.0.1:/ /mnt/efs -o rw,nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport,port=20770" with 15 sec time limit.
2025-12-23 15:33:19 UTC - ERROR - Mounting fs-07701a5204ba93ddc.efs.ap-south-1.amazonaws.com to /mnt/efs failed due to timeout after 15 sec, mount attempt 2/3, wait 0 sec before next attempt.
2025-12-23 15:33:19 UTC - INFO - Executing: "/sbin/mount.nfs4 127.0.0.1:/ /mnt/efs -o rw,nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport,port=20770"
2025-12-23 15:36:19 UTC - ERROR - Failed to mount fs-07701a5204ba93ddc.efs.ap-south-1.amazonaws.com at /mnt/efs: returncode=32, stderr="b'mount.nfs4: Connection timed out'"
2025-12-23 15:37:16 UTC - INFO - version=2.4.1 options={'rw': None, 'tls': None}
2025-12-23 15:37:16 UTC - INFO - binding 21018
2025-12-23 15:37:16 UTC - INFO - Starting efs-proxy: "/sbin/efs-proxy /var/run/efs/stunnel-config.fs-07701a5204ba93ddc.mnt.efs.21018 --tls"
2025-12-23 15:37:16 UTC - INFO - Started efs-proxy, pid: 31666
2025-12-23 15:37:16 UTC - WARNING - Error connecting to 127.0.0.1:21018, [Errno 111] Connection refused
2025-12-23 15:37:16 UTC - INFO - Executing: "/sbin/mount.nfs4 127.0.0.1:/ /mnt/efs -o rw,nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport,port=21018" with 15 sec time limit.
2025-12-23 15:37:19 UTC - INFO - Successfully mounted fs-07701a5204ba93ddc.efs.ap-south-1.amazonaws.com at /mnt/efs
[root@ip-10-0-132-218 ec2-user]#
[root@ip-10-0-132-218 ec2-user]#
[root@ip-10-0-132-218 ec2-user]#
[root@ip-10-0-132-218 ec2-user]# echo "EFS WORKING" | sudo tee /mnt/efs/test.txt
cat /mnt/efs/test.txt
EFS WORKING
EFS WORKING
[root@ip-10-0-132-218 ec2-user]#
Search
ENG
IN
3:42 PM
12/23/2025