Project Report

On

Incidentpulse – Devops Incident & Patch Evidence Portal

(Linux + Aws Real-Time Enterprise Project)

Submitted By

Shrusti P Chavan

IT Skill Nest

Rajajinagar, Bengaluru – 560 010

2025 – 2026

TABLE OF CONTENTS

CHAPTERS **PAGE NO.**

CHAPTER 1

INTRODUCTION

The **IncidentPulse – DevOps Incident & Patch Evidence Portal** is a real-time enterprise-oriented project designed to simulate how modern organizations manage production incidents, incident evidence, remediation documents, and operational audits using **Linux and AWS**.

The project focuses on implementing secure file handling, role-based access control, automated backups, scalable infrastructure, and cloud-based storage solutions. It provides hands-on experience in Linux system administration and cloud infrastructure design while following real-world DevOps practices.

1.1 PURPOSE AND OBJECTIVES

The main objectives of this project are:

- To design a secure incident management system using Linux and AWS
- To implement role-based access control for different user roles
- To automate backups, archival, and evidence handling
- To demonstrate scalable and highly available cloud architecture
- To gain practical exposure to DevOps tools and workflows

CHAPTER 2

SYSTEM / PROCESS OVERVIEW

2.1 HIGH-LEVEL SYSTEM OVERVIEW

The IncidentPulse system consists of a Linux-based frontend portal hosted on AWS EC2, integrated with multiple AWS services for storage, automation, and scalability. The system allows engineers to upload incident evidence securely, incident leads to manage runbooks and patch notes, and audit users to access compliance logs.

2.2 PROCESS FLOW / METHODOLOGY

The overall process flow is as follows:

1. Users access the IncidentPulse portal hosted on Linux EC2.
2. Engineers upload incident evidence securely using FTP with TLS.
3. Incident leads manage runbooks and remediation documents.
4. Evidence and documents are stored and archived using AWS S3.
5. Metadata related to incidents is stored in an RDS MySQL database.
6. Automation is handled using cron jobs, Lambda, and API Gateway.
7. CloudFront distributes runbooks securely using signed URLs.

CHAPTER 3

IMPLEMENTATION AND EXECUTION DETAILS

3.1 LINUX USER AND GROUP MANAGEMENT

Linux users and groups were created based on organizational roles:

- Incident Leads
- Engineers
- Audit / Operations

Each user was assigned to their respective group, and access was controlled using Linux permissions and ACLs to ensure separation of duties.

3.2 DIRECTORY STRUCTURE AND ACCESS CONTROL

Different directories were created for specific purposes such as runbooks, evidence uploads, and audit logs. Access permissions were strictly enforced using role-based permissions and ACLs to prevent unauthorized access.

3.3 FILE HANDLING AND VALIDATION

Incident evidence uploads were restricted to .tar.gz format to ensure standardized and compressed data storage. Older runbooks were archived weekly to maintain organized documentation and optimize storage usage.

3.4 STORAGE MANAGEMENT

Logical Volume Management (LVM) was used to create a dedicated partition for incident evidence. Disk usage was monitored, and storage was automatically expanded when usage exceeded a defined threshold. Sensitive data was secured using LUKS encryption.

3.5 AUTOMATION USING CRON JOBS

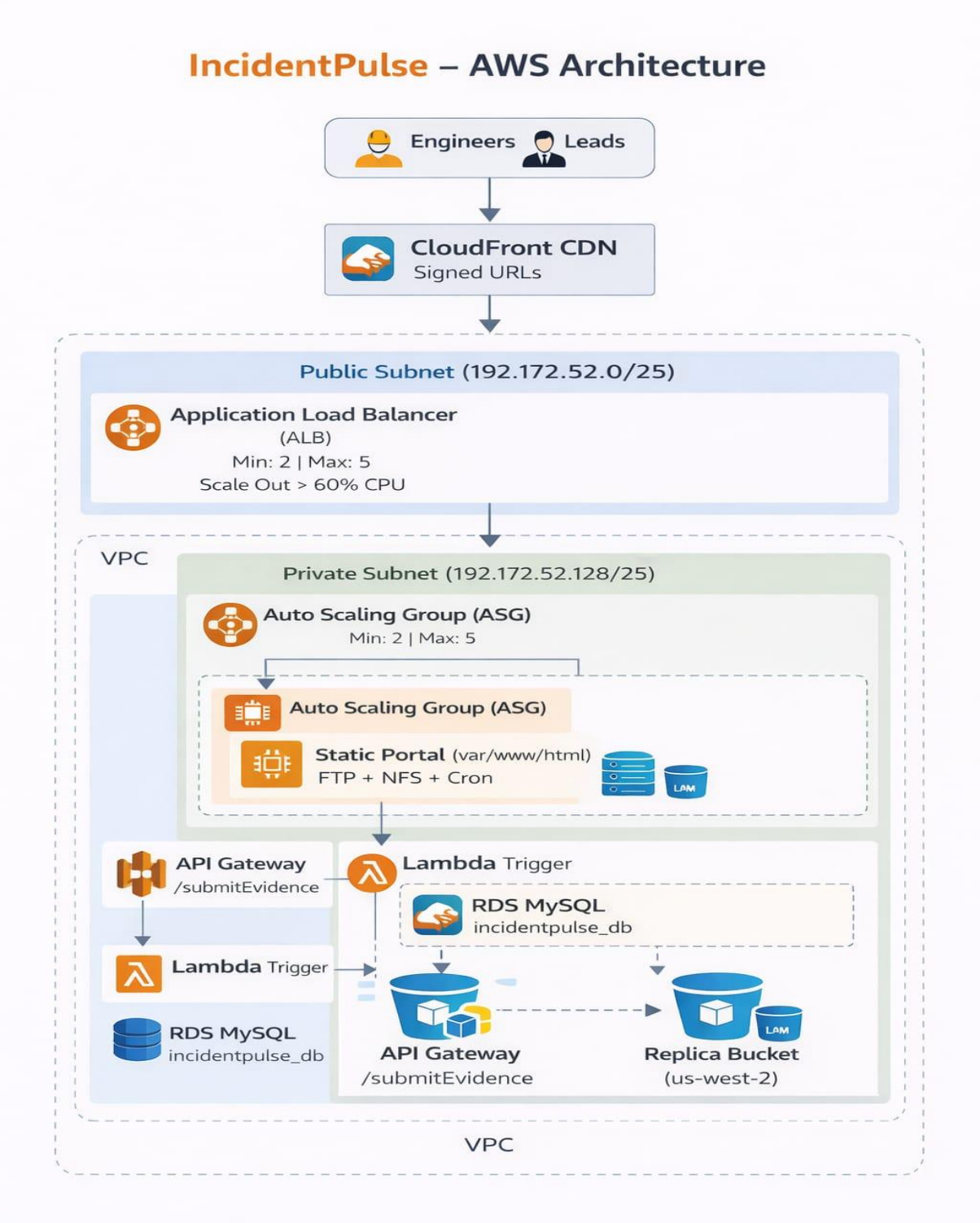
Cron jobs were configured to automate daily backups, weekly log uploads, and monthly archival tasks. This ensured consistency, reliability, and reduced manual effort.

3.6 AWS INFRASTRUCTURE IMPLEMENTATION

The AWS infrastructure included EC2, VPC, S3, IAM, RDS, ALB, Auto Scaling Group, CloudFront, Lambda, and API Gateway. The architecture was designed for high availability, scalability, and security while strictly using the permitted services.

CHAPTER 4

FLOWCHART


IncidentPulse – AWS Architecture
Engineers
Leads
CloudFront CDN
Signed URLs
Public Subnet (192.172.52.0/25)
Application Load Balancer
(ALB)
Min: 2 | Max: 5
Scale Out > 60% CPU
VPC
Private Subnet (192.172.52.128/25)
Auto Scaling Group (ASG)
Min: 2 | Max: 5
Auto Scaling Group (ASG)
Static Portal (var/www/html)
FTP + NFS + Cron
API Gateway
/submitEvidence
Lambda Trigger
RDS MySQL
incidentpulse_db
Lambda Trigger
RDS MySQL
incidentpulse_db
API Gateway
/submitEvidence
Replica Bucket
(us-west-2)
VPC

CHAPTER 5

STEP-BY-STEP EXPLANATION

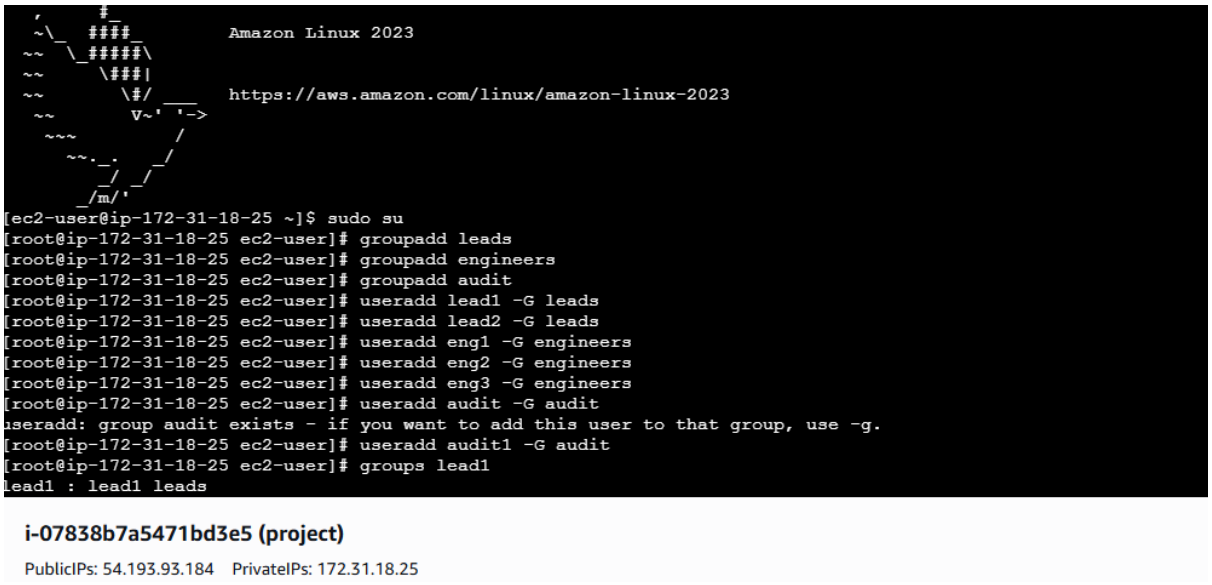
This section provides a detailed step-by-step explanation of the configuration, setup, and logical flow involved in implementing the *IncidentPulse – DevOps Incident & Patch Evidence Portal*.

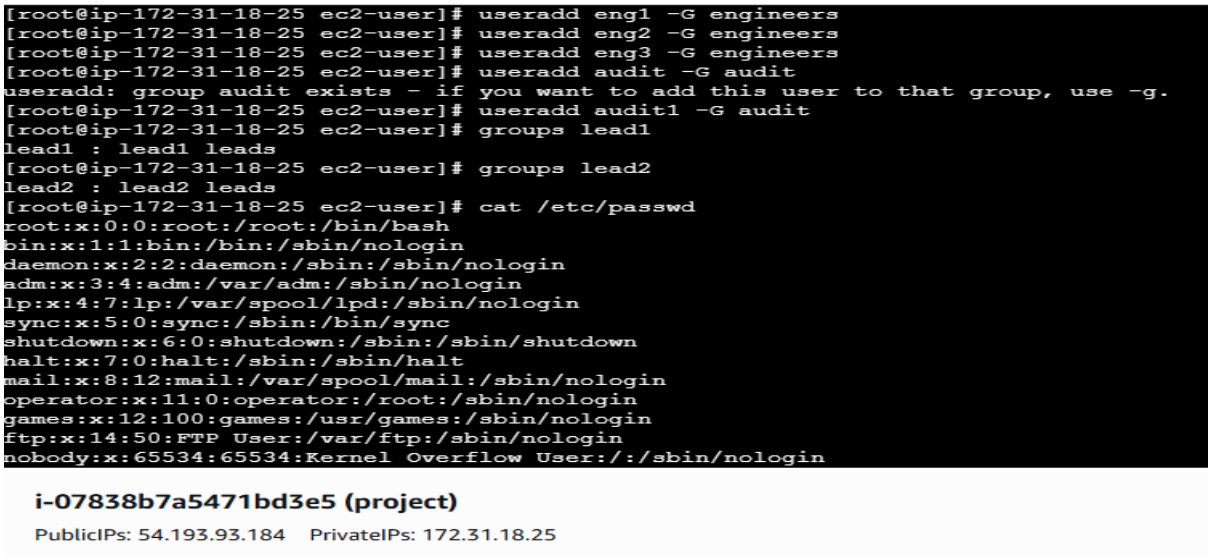
5.1 EC2 INSTANCE DEPLOYMENT

1. EC2 instances were launched using **Amazon Linux 2 AMI** with instance type **t2.micro**.
2. A **20GB EBS volume** was attached to each instance for application and data storage.
3. One EC2 instance was placed in the **public subnet** for initial access.
4. Backend EC2 instances were deployed in the **private subnet**.
5. Secure access was established from the public instance to private instances.
6. A web server (Apache/Nginx) was installed and the portal was hosted under /var/www/html.
7. Security Groups were configured to allow only required ports (HTTP/HTTPS/SSH).

5.2 LINUX USER AND GROUP CONFIGURATION

- Linux groups (leads, engineers, audit) were created.
- Users were created and mapped to their respective groups.
- File ownership and permissions were assigned based on role.
- Audit users were restricted to read-only access for compliance.





```
ec2-user:x:1000:1000:EC2 Default User:/home/ec2-user:/bin/bash
lead1:x:1001:1004::/home/lead1:/bin/bash
lead2:x:1002:1005::/home/lead2:/bin/bash
eng1:x:1003:1006::/home/eng1:/bin/bash
eng2:x:1004:1007::/home/eng2:/bin/bash
eng3:x:1005:1008::/home/eng3:/bin/bash
audit1:x:1006:1009::/home/audit1:/bin/bash
[root@ip-172-31-18-25 ec2-user]# cat /etc/groups
cat: /etc/groups: No such file or directory
[root@ip-172-31-18-25 ec2-user]# cat /etc/group
root:x:0:
bin:x:1:
daemon:x:2:
sys:x:3:
adm:x:4:ec2-user
tty:x:5:
disk:x:6:
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 54.193.93.184 PrivateIPs: 172.31.18.25

```
ec2-user:x:1000:
leads:x:1001:lead1,lead2
engineers:x:1002:eng1,eng2,eng3
audit:x:1003:audit1
lead1:x:1004:
lead2:x:1005:
eng1:x:1006:
eng2:x:1007:
eng3:x:1008:
audit1:x:1009:
[root@ip-172-31-18-25 ec2-user]#
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 54.193.93.184 PrivateIPs: 172.31.18.25

```
root@ip-172-31-18-25 ec2-user]# gpasswd -M lead1,lead2 leads
root@ip-172-31-18-25 ec2-user]# gpasswd -M eng1,eng2,eng3 engineers
root@ip-172-31-18-25 ec2-user]# usermod -a-G audit1 audit
sermod: invalid option -- '-'
sage: usermod [options] LOGIN

ptions:
 -b, --badnames                allow bad names
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 54.193.93.184 PrivateIPs: 172.31.18.25

```
root@ip-172-31-18-25 ec2-user]# usermod -aG audit1 audit
sermod: user 'audit' does not exist
root@ip-172-31-18-25 ec2-user]# usermod -aG audit audit1
root@ip-172-31-18-25 ec2-user]# grep leads /etc/group
eads:x:1001:lead1,lead2
root@ip-172-31-18-25 ec2-user]# grep engineers /etc/group
ngineers:x:1002:eng1,eng2,eng3
root@ip-172-31-18-25 ec2-user]# grep audit /etc/group
udit:x:1003:audit1
udit1:x:1009:
root@ip-172-31-18-25 ec2-user]#
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 54.193.93.184 PrivateIPs: 172.31.18.25

5.3 PASSWORD POLICY ENFORCEMENT

- System-wide password policies were configured using PAM.
- Password expiry was set to 30 days.
- Minimum password length and complexity rules were enforced.
- Policy effectiveness was verified through password reset tests.

```
root@ip-172-31-18-25 ec2-user]# chage -E 2026-02-21 lead1
root@ip-172-31-18-25 ec2-user]# chage -E 2026-02-21 lead2
root@ip-172-31-18-25 ec2-user]# chage -E 2026-02-21 eng1
root@ip-172-31-18-25 ec2-user]# chage -E 2026-02-21 eng2
root@ip-172-31-18-25 ec2-user]# chage -E 2026-02-21 eng3
root@ip-172-31-18-25 ec2-user]# chage -E 2026-02-21 audit1
root@ip-172-31-18-25 ec2-user]# chage -l lead1
ast password change                                     : Jan 21, 2026
assword expires                                         : never
assword inactive                                        : never
ccount expires                                          : Feb 21, 2026
inimum number of days between password change           : 0
```

```
[root@ip-172-31-18-25 ec2-user]# passwd lead1
Changing password for user lead1.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

```
root@ip-172-31-18-25 ec2-user]# passwd lead2
Changing password for user lead2.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
root@ip-172-31-18-25 ec2-user]# passwd eng1
Changing password for user eng1.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

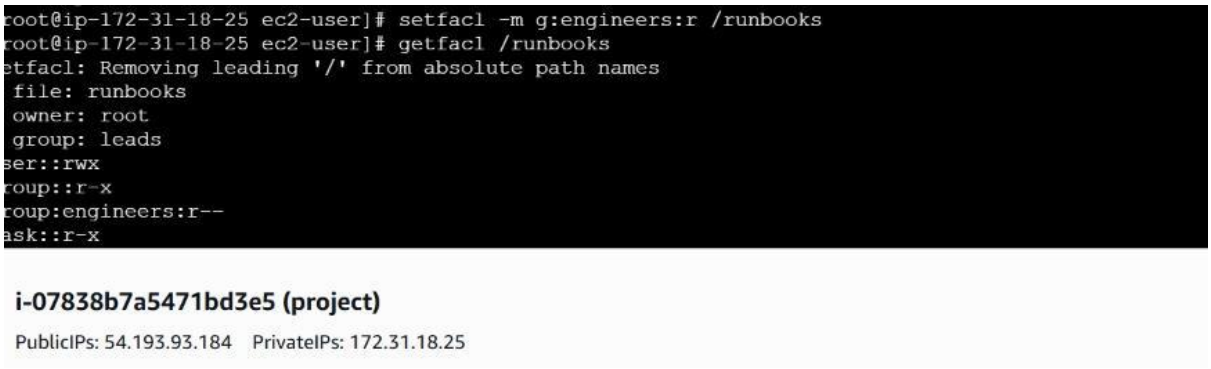
```
[root@ip-172-31-18-25 ec2-user]# nano /etc/security/pwquality.conf
[root@ip-172-31-18-25 ec2-user]# passwd eng2
Changing password for user eng2.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
Sorry, passwords do not match.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
[root@ip-172-31-18-25 ec2-user]# passwd eng3
Changing password for user eng3.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@ip-172-31-18-25 ec2-user]# passwd audit1
Changing password for user audit1.
New password:
Retype new password:
```

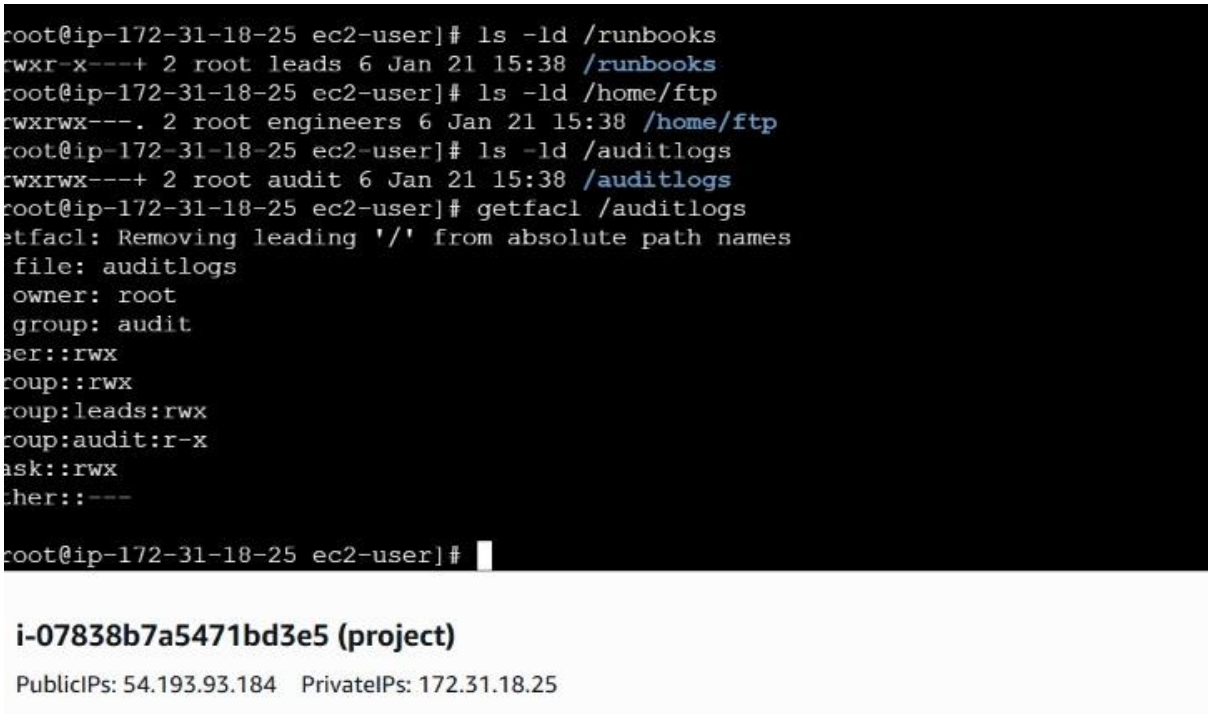
i-07838b7a5471bd3e5 (project)

PublicIPs: 54.193.93.184 PrivateIPs: 172.31.18.25

5.4 DIRECTORY SETUP AND ACCESS CONTROL

- Required directories were created:
  - /runbooks
  - /home/ftp
  - /auditlogs
- Linux permissions and ACLs were applied to enforce role-based access.
- Engineers were prevented from accessing directories outside their scope.
- Separation of duties was ensured.





5.5 FTP CONFIGURATION WITH TLS

- FTP service was installed and configured with TLS encryption.
- /home/ftp was set as the upload directory.
- Chroot was enabled to restrict engineers to the FTP directory.
- Upload permissions were limited to .tar.gz files only.
- This ensured secure and controlled evidence uploads.

```
    https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.10.20260120.html

 Version 2023.10.20260120:
   Run the following command to upgrade to 2023.10.20260120:

     dnf upgrade --releasever=2023.10.20260120

   Release notes:
    https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.10.20260120.html

================================================================================

nstalled:
  vsftpd-3.0.5-1.amzn2023.0.2.x86_64

omplete!
root@ip-172-31-18-25 ec2-user]# systemctl start vsftpd
root@ip-172-31-18-25 ec2-user]# systemctl enable vsftpd
reated symlink /etc/systemd/system/multi-user.target.wants/vsftpd.service → /usr/lib/systemd/system/vsftpd.service.
root@ip-172-31-18-25 ec2-user]#
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 204.236.147.188 PrivateIPs: 172.31.18.25

```
# with the listen_ipv6 directive.
listen=NO
#
# This directive enables listening on IPv6 sockets. By default, listening
# on the IPv6 "any" address (::) will accept connections from both IPv6
# and IPv4 clients. It is not necessary to listen on *both* IPv4 and IPv6
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
# Make sure, that one of the listen options is commented !!
listen_ipv6=YES

pam_service_name=vsftpd
userlist_enable=YES
pasv_enable=YES
pasv_min_port=1024
pasv_max_port=1048
pasv_address=204.236.147.188
allow_writeable_chroot=YES
-- INSERT --
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 204.236.147.188 PrivateIPs: 172.31.18.25

```
Complete!
[root@ip-172-31-18-25 ec2-user]# systemctl start vsftpd
[root@ip-172-31-18-25 ec2-user]# systemctl enable vsftpd
Created symlink /etc/systemd/system/multi-user.target.wants/vsftpd.service → /usr/lib/systemd/system/vsftpd.service.
[root@ip-172-31-18-25 ec2-user]# vi /etc/vsftpd/vsftpd.conf
[root@ip-172-31-18-25 ec2-user]# cd /home/ftp
[root@ip-172-31-18-25 ftp]# ls
[root@ip-172-31-18-25 ftp]# ls -ld
drwxrwx---. 2 root engineers 6 Jan 21 15:38 .
[root@ip-172-31-18-25 ftp]# ls -l
total 0
[root@ip-172-31-18-25 ftp]#
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 204.236.147.188 PrivateIPs: 172.31.18.25

5.6 FILE VALIDATION AND HANDLING LOGIC

- Uploaded files were validated for correct format (.tar.gz).
- Only compressed evidence bundles were accepted.
- Invalid file uploads were rejected automatically.
- Valid files were moved to /incident_store for storage.

```
[root@ip-172-31-18-25 ec2-user]# passwd audit1
Changing password for user audit1.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@ip-172-31-18-25 ec2-user]# mkdir /runbooks
[root@ip-172-31-18-25 ec2-user]# mkdir /home/ftp
[root@ip-172-31-18-25 ec2-user]# mkdir /auditlogs
[root@ip-172-31-18-25 ec2-user]# chown :leads /runbooks
[root@ip-172-31-18-25 ec2-user]# chown :engineers /home/ftp
[root@ip-172-31-18-25 ec2-user]# chown :audit /auditlogs
[root@ip-172-31-18-25 ec2-user]# chmod 750 /runbooks
[root@ip-172-31-18-25 ec2-user]# chmod 770 /home/ftp
[root@ip-172-31-18-25 ec2-user]# chmod 770 /auditlogs
[root@ip-172-31-18-25 ec2-user]#
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 54.193.93.184 PrivateIPs: 172.31.18.25

```
root@ip-172-31-18-25 ec2-user]# setfacl -m g:audit:rx /auditlogs
root@ip-172-31-18-25 ec2-user]# setfacl -m g:leads:rwx /auditlogs
root@ip-172-31-18-25 ec2-user]# getfacl /auditlogs
etfacl: Removing leading '/' from absolute path names
 file: auditlogs
 owner: root
 group: audit
ser::rwx
roup::rwx
roup:leads:rwx
roup:audit:r-x
ask::rwx
ther::---
```

```
root@ip-172-31-18-25 ec2-user]# setfacl -m g:engineers:r /runbooks
root@ip-172-31-18-25 ec2-user]# getfacl /runbooks
etfacl: Removing leading '/' from absolute path names
 file: runbooks
 owner: root
 group: leads
ser::rwx
roup::r-x
roup:engineers:r--
ask::r-x
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 54.193.93.184 PrivateIPs: 172.31.18.25

```
root@ip-172-31-18-25 ec2-user]# ls -ld /runbooks
rwxr-x---+ 2 root leads 6 Jan 21 15:38 /runbooks
root@ip-172-31-18-25 ec2-user]# ls -ld /home/ftp
rwxrwx---. 2 root engineers 6 Jan 21 15:38 /home/ftp
root@ip-172-31-18-25 ec2-user]# ls -ld /auditlogs
rwxrwx---+ 2 root audit 6 Jan 21 15:38 /auditlogs
root@ip-172-31-18-25 ec2-user]# getfacl /auditlogs
etfacl: Removing leading '/' from absolute path names
 file: auditlogs
 owner: root
 group: audit
ser::rwx
roup::rwx
roup:leads:rwx
roup:audit:r-x
ask::rwx
ther::---

root@ip-172-31-18-25 ec2-user]#
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 54.193.93.184 PrivateIPs: 172.31.18.25

5.7 STORAGE MANAGEMENT USING LVM

- Logical Volume Management (LVM) was used to create /incident_store.
- Initial size was set to 5GB.
- Disk usage was continuously monitored.
- When usage exceeded 80%, the volume was expanded to 10GB automatically.
- This ensured uninterrupted evidence storage.

```
    ~~~          /
      ~~._.    _/
         _/ _/
       _/m/'
Last login: Fri Jan 23 16:07:51 2026 from 13.52.6.1
[ec2-user@ip-172-31-18-25 ~]$ sudo su
[root@ip-172-31-18-25 ec2-user]# lsblk
NAME          MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
nvme0n1       259:0    0   8G  0 disk
├─nvme0n1p1   259:1    0   8G  0 part /
├─nvme0n1p127 259:2    0   1M  0 part
└─nvme0n1p128 259:3    0  10M  0 part /boot/efi
nvme1n1       259:4    0  20G  0 disk
[root@ip-172-31-18-25 ec2-user]#
```

i-07838b7a5471bd3e5 (project)

```
0 2 * * 0 tar -czvf /archieve/runbooks/runbooks.tar.gz /runbooks
*/5 * * * df -h /home/ec2-user/incident_store |grep -q "8[0-9]%" && lvextend -L +10G /dev/incident_vg/incident_lv -r
~
~
~
```

```
Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x18366a6e.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-41943039, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-41943039, default 41943039): +18G

Created a new partition 1 of type 'Linux' and of size 18 GiB.

Command (m for help): t
Selected partition 1
Hex code or alias (type L to list all): 8e
Changed type of partition 'Linux' to 'Linux LVM'.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 50.18.142.71 PrivateIPs: 172.31.18.25

```
alling ioctl() to re-read partition table.
yncing disks.

root@ip-172-31-18-25 ec2-user]# partprobe /dev/sdb
root@ip-172-31-18-25 ec2-user]# partx /dev/sdb
R START      END  SECTORS SIZE NAME UUID
1  2048 37750783 37748736  18G      18366a6e-01
root@ip-172-31-18-25 ec2-user]# pvcreate /dev/sdb
 Cannot use /dev/sdb: device is partitioned
root@ip-172-31-18-25 ec2-user]# yum install lvm2 -y
ast metadata expiration check: 0:21:39 ago on Tue Jan 27 15:03:38 2026.
ackage lvm2-2.03.16-1.amzn2023.0.5.x86_64 is already installed.
ependencies resolved.
othing to do.
omplete!
root@ip-172-31-18-25 ec2-user]# pvcreate /dev/sdb1
 Physical volume "/dev/sdb1" successfully created.
root@ip-172-31-18-25 ec2-user]# pvs
 PV         VG Fmt  Attr PSize  PFree
 /dev/sdb1     lvm2 ---  18.00g 18.00g
root@ip-172-31-18-25 ec2-user]# vgcreate incident_vg /dev/sdb1
 Volume group "incident_vg" successfully created
root@ip-172-31-18-25 ec2-user]# vgs
 VG          #PV #LV #SN Attr   VSize   VFree
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 50.18.142.71 PrivateIPs: 172.31.18.25

```
 Volume group "incident_vg" successfully created
root@ip-172-31-18-25 ec2-user]# vgs
 VG          #PV #LV #SN Attr   VSize   VFree
 incident_vg   1   0   0 wz--n- <18.00g <18.00g
root@ip-172-31-18-25 ec2-user]# lvcreate -L +5G -n incident_lv incident_vg
 Logical volume "incident_lv" created.
root@ip-172-31-18-25 ec2-user]# lvs
 LV          VG          Attr       LSize Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
 incident_lv incident_vg -wi-a----- 5.00g
root@ip-172-31-18-25 ec2-user]# mkfs.xfs /dev/incident_vg/incident_lv
eta-data=/dev/incident_vg/incident_lv isize=512    agcount=8, agsize=163840 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1 nrext64=0
         =                       exchange=0
ata      =                       bsize=4096   blocks=1310720, imaxpct=25
         =                       sunit=1      swidth=1 blks
aming    =version 2              bsize=4096   ascii-ci=0, ftype=1, parent=0
og       =internal log           bsize=4096   blocks=16384, version=2
         =                       sectsz=512   sunit=1 blks, lazy-count=1
ealtime =none                    extsz=4096   blocks=0, rtextents=0
root@ip-172-31-18-25 ec2-user]# ls
ncident_001.tar.gz  incident_buddle  incident_evidence  incident_store
root@ip-172-31-18-25 ec2-user]# mount /dev/incident_vg/incident_lv incident_store
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 50.18.142.71 PrivateIPs: 172.31.18.25

```
oot@ip-172-31-18-25 ec2-user]# vi /etc/fstab

oot@ip-172-31-18-25 ec2-user]# lvextend -L +10G /dev/incident_vg/incident_lv
 Size of logical volume incident_vg/incident_lv changed from 5.00 GiB (1280 extents) to 15.00 GiB (3840 extents).
 Logical volume incident_vg/incident_lv successfully resized.
oot@ip-172-31-18-25 ec2-user]# xfs_growfs incident_store
eta-data=/dev/mapper/incident_vg-incident_lv isize=512    agcount=8, agsize=163840 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1 nrext64=0
         =                       exchange=0
ata      =                       bsize=4096   blocks=1310720, imaxpct=25
         =                       sunit=1      swidth=1 blks
aming    =version 2              bsize=4096   ascii-ci=0, ftype=1, parent=0
og       =internal log           bsize=4096   blocks=16384, version=2
         =                       sectsz=512   sunit=1 blks, lazy-count=1
ealtime =none                    extsz=4096   blocks=0, rtextents=0
ata blocks changed from 1310720 to 3932160
oot@ip-172-31-18-25 ec2-user]# df -h
ilesystem                        Size  Used Avail Use% Mounted on
evtmpfs                          4.0M     0  4.0M   0% /dev
mpfs                             459M     0  459M   0% /dev/shm
mpfs                             184M  456K  183M   1% /run
dev/nvme0n1p1                    8.0G  1.7G  6.3G  21% /
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 50.18.142.71 PrivateIPs: 172.31.18.25

```
mpfs                             459M     0  459M   0% /dev/shm
mpfs                             184M  456K  183M   1% /run
/dev/nvme0n1p1                   8.0G  1.7G  6.3G  21% /
mpfs                             459M     0  459M   0% /tmp
/dev/nvme0n1p128                  10M  1.3M  8.7M  13% /boot/efi
mpfs                              92M     0   92M   0% /run/user/1000
/dev/mapper/incident_vg-incident_lv   15G  141M   15G   1% /home/ec2-user/incident_store
[root@ip-172-31-18-25 ec2-user]# crontab -e
crontab: installing new crontab
```

```
Invalid crontab file, can't install.
Do you want to retry the same edit? (Y/N) n
crontab: edits left in /tmp/crontab.OAfzyI
[root@ip-172-31-18-25 ec2-user]# systemctl status crond
● crond.service - Command Scheduler
     Loaded: loaded (/usr/lib/systemd/system/crond.service; enabled; preset: enabled)
     Active: active (running) since Tue 2026-01-27 15:03:35 UTC; 32min ago
   Main PID: 1571 (crond)
      Tasks: 1 (limit: 1067)
     Memory: 1.0M
        CPU: 11ms
     CGroup: /system.slice/crond.service
             └─1571 /usr/sbin/crond -n

Jan 27 15:03:35 ip-172-31-18-25.us-west-1.compute.internal systemd[1]: Started crond.service - Command Scheduler.
Jan 27 15:03:35 ip-172-31-18-25.us-west-1.compute.internal crond[1571]: (CRON) STARTUP (1.5.7)
Jan 27 15:03:35 ip-172-31-18-25.us-west-1.compute.internal crond[1571]: (CRON) INFO (Syslog will be used instead of sendmail.)
Jan 27 15:03:35 ip-172-31-18-25.us-west-1.compute.internal crond[1571]: (CRON) INFO (RANDOM_DELAY will be scaled with factor 13% if used.)
Jan 27 15:03:35 ip-172-31-18-25.us-west-1.compute.internal crond[1571]: (CRON) INFO (running with inotify support)
[root@ip-172-31-18-25 ec2-user]# df -h /home/ec2-user/incident_store
Filesystem                      Size  Used Avail Use% Mounted on
/dev/mapper/incident_vg-incident_lv   15G  141M   15G   1% /home/ec2-user/incident_store
[root@ip-172-31-18-25 ec2-user]#
[root@ip-172-31-18-25 ec2-user]#
```

i-07838b7a5471bd3e5 (project)

```
UID=497419d5-2417-43e6-927d-6777766bb648     /           xfs    defaults,noatime  1   1
UID=A1F2-F73A        /boot/efi       vfat    defaults,noatime,uid=0,gid=0,umask=0077,shortname=winnt,x-systemd.automount 0 2
/dev/incident_vg/incident_lv incident_store xfs defaults 0 0
```

```
Command (m for help): n
Partition type
   p   primary (1 primary, 0 extended, 3 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (2-4, default 2): 2
First sector (37750784-41943039, default 37750784):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (37750784-41943039, default 41943039): +2G
Value out of range.
Last sector, +/-sectors or +/-size{K,M,G,T,P} (37750784-41943039, default 41943039):

Created a new partition 2 of type 'Linux' and of size 2 GiB.

Command (m for help):
```

```
Command (m for help): w
The partition table has been altered.
Syncing disks.

[root@ip-172-31-18-25 ec2-user]# lsblk
NAME                        MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
nvme0n1                     259:0    0   8G  0 disk
├─nvme0n1p1                 259:1    0   8G  0 part /
├─nvme0n1p127               259:2    0   1M  0 part
└─nvme0n1p128               259:3    0  10M  0 part /boot/efi
nvme1n1                     259:4    0  20G  0 disk
├─nvme1n1p1                 259:5    0  18G  0 part
│ └─incident_vg-incident_lv 253:0    0  15G  0 lvm  /home/ec2-user/incident_store
└─nvme1n1p2                 259:6    0   2G  0 part
[root@ip-172-31-18-25 ec2-user]#
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 50.18.142.71 PrivateIPs: 172.31.18.25

5.8 SECURE VAULT CONFIGURATION (LUKS)

- A 2GB partition /vault_secure was created.
- LUKS encryption was applied to protect sensitive data.
- Only Incident Leads were authorized to mount the encrypted volume.
- Engineers and audit users were denied access.

```
[root@ip-172-31-18-25 ec2-user]# yum install cryptsetup -y
Last metadata expiration check: 0:42:45 ago on Tue Jan 27 15:03:38 2026.
Package cryptsetup-2.6.1-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-18-25 ec2-user]# cryptsetup luksFormat /dev/sdb2

WARNING!
========
This will overwrite data on /dev/sdb2 irrevocably.

Are you sure? (Type 'yes' in capital letters): YES
Enter passphrase for /dev/sdb2:
Verify passphrase:
[root@ip-172-31-18-25 ec2-user]# cryptsetup luksOpen /dev/sdb2 /vault_secure
Enter passphrase for /dev/sdb2:
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 50.18.142.71 PrivateIPs: 172.31.18.25

```
[root@ip-172-31-18-25 ec2-user]# cryptsetup luksFormat /dev/sdb2

WARNING!
========
This will overwrite data on /dev/sdb2 irrevocably.

Are you sure? (Type 'yes' in capital letters): YES
Enter passphrase for /dev/sdb2:
Verify passphrase:
[root@ip-172-31-18-25 ec2-user]# cryptsetup luksOpen /dev/sdb2 /vault_secure
Enter passphrase for /dev/sdb2:
Name "/vault_secure" invalid. It contains "/".
[root@ip-172-31-18-25 ec2-user]# cryptsetup luksOpen /dev/sdb2 vault_secure
Enter passphrase for /dev/sdb2:
No key available with this passphrase.
Enter passphrase for /dev/sdb2:
[root@ip-172-31-18-25 ec2-user]# ls /dev/mapper
control  incident_vg-incident_lv  vault_secure
[root@ip-172-31-18-25 ec2-user]# mkfs.xfs /dev/mapper/vault_secure
meta-data=/dev/mapper/vault_secure isize=512    agcount=8, agsize=64992 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1 nrext64=0
         =                       exchange=0
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 50.18.142.71 PrivateIPs: 172.31.18.25

```
realtime =none                   extsz=4096   blocks=0, rtextents=0
[root@ip-172-31-18-25 ec2-user]# mkdir secure
[root@ip-172-31-18-25 ec2-user]# mount /dev/mapper/vault_secure secure
[root@ip-172-31-18-25 ec2-user]# df -h |grep secure
/dev/mapper/vault_secure              2.0G   47M  1.9G   3% /home/ec2-user/secure
[root@ip-172-31-18-25 ec2-user]#
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 50.18.142.71 PrivateIPs: 172.31.18.25

```
root@ip-172-31-18-25 ec2-user]# grep leads /etc/group
eads:x:1001:lead1,lead2
root@ip-172-31-18-25 ec2-user]# ls -ld
rwx------. 7 ec2-user ec2-user 16384 Jan 27 15:56 .
root@ip-172-31-18-25 ec2-user]# groupmod -n incident_leads leads
root@ip-172-31-18-25 ec2-user]# grep incident_leads /etc/group
ncident_leads:x:1001:lead1,lead2
```

```
incident_leads:x:1001:lead1,lead2
[root@ip-172-31-18-25 ec2-user]# chown root:incident_leads secure
[root@ip-172-31-18-25 ec2-user]# chmod 770 secure
[root@ip-172-31-18-25 ec2-user]# ls -ld secure
drwxrwx---. 2 root incident_leads 6 Jan 27 15:55 secure
[root@ip-172-31-18-25 ec2-user]# setfacl -m g:incident_leads:rwx /home/ec2-user
[root@ip-172-31-18-25 ec2-user]# su lead1
[lead1@ip-172-31-18-25 ec2-user]$ cd /home/ec2-user/secure
[lead1@ip-172-31-18-25 secure]$ touch file1
[lead1@ip-172-31-18-25 secure]$ ls
file1
[lead1@ip-172-31-18-25 secure]$ exit
exit
[root@ip-172-31-18-25 ec2-user]# su eng1
[eng1@ip-172-31-18-25 ec2-user]$ cd /home/ec2-user/secure
bash: cd: /home/ec2-user/secure: Permission denied
[eng1@ip-172-31-18-25 ec2-user]$ exit
exit
[root@ip-172-31-18-25 ec2-user]# su lead2
[lead2@ip-172-31-18-25 ec2-user]$ cd /home/ec2-user/secure
[lead2@ip-172-31-18-25 secure]$ touch file2
[lead2@ip-172-31-18-25 secure]$ ls
file1  file2
[lead2@ip-172-31-18-25 secure]$
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 50.18.142.71 PrivateIPs: 172.31.18.25

5.9 NFS CONFIGURATION FOR RUNBOOKS

- /runbooks directory was shared using NFS.
- Incident Leads were given read/write access.
- Engineers were given read-only access.
- Access was restricted to authorized client systems only.

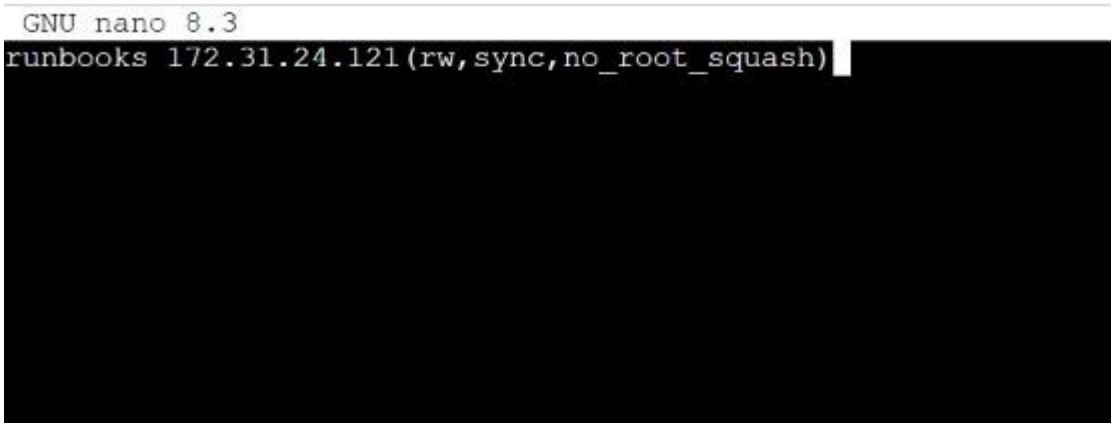
```
A newer release of "Amazon Linux" is available.
  Version 2023.10.20260120:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Tue Jan 27 15:07:05 2026 from 13.52.6.115
[ec2-user@ip-172-31-18-25 ~]$ sudo sau
sudo: sau: command not found
[ec2-user@ip-172-31-18-25 ~]$ sudo su
[root@ip-172-31-18-25 ec2-user]# yum install nfs-utils -y
Last metadata expiration check: 1 day, 0:09:38 ago on Tue Jan 27 15:03:38 2026.
Package nfs-utils-1:2.5.4-2.rc3.amzn2023.0.3.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-18-25 ec2-user]#
```

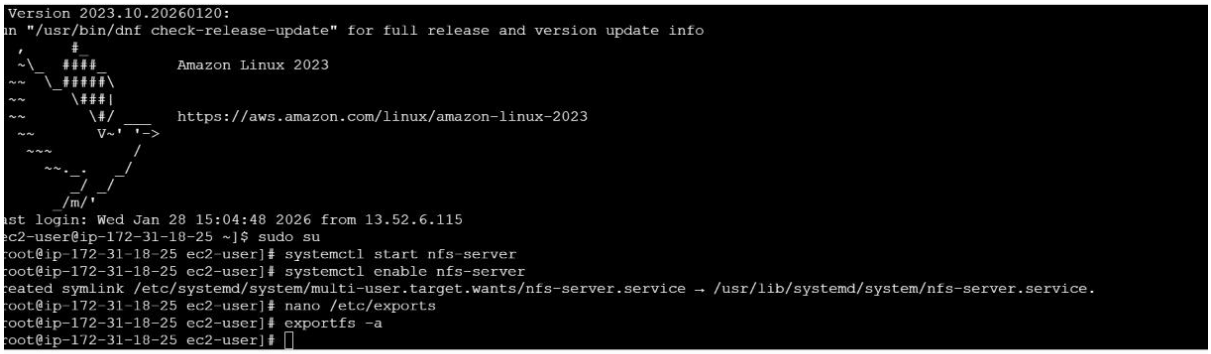
i-07838b7a5471bd3e5 (project)

PublicIPs: 54.176.201.143 PrivateIPs: 172.31.18.25

i-02bf5ec92dba94028 (project2)

PublicIPs: 50.18.29.114 PrivateIPs: 172.31.24.121





i-07838b7a5471bd3e5 (project)

```
  ~~       V~' '->
   ~~~         /
     ~~._.   _/
        _/ _/
      _/m/'
ast login: Wed Jan 28 15:13:59 2026 from 13.52.6.116
ec2-user@ip-172-31-24-121 ~]$ sudo su
root@ip-172-31-24-121 ec2-user]# mkdir nfs
root@ip-172-31-24-121 ec2-user]# mount 54.183.114.185:/runbooks nfs
ount.nfs: Connection timed out
root@ip-172-31-24-121 ec2-user]# mount 54.183.114.185:/runbooks /home/ec2-user/nfs
ount.nfs: Connection timed out
root@ip-172-31-24-121 ec2-user]# mount 172.31.18.25:/runbooks /home/ec2-user/nfs
root@ip-172-31-24-121 ec2-user]# mount
roc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
/sfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
evtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=4096k,nr_inodes=113868,mode=755)
ecurityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
mpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel)
evpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000)
```

i-02bf5ec92dba94028 (project2)

PublicIPs: 54.176.174.221 PrivateIPs: 172.31.24.121

```
GNU nano 8.3                                    /etc/fstab
UID=497419d5-2417-43e6-927d-6777766bb648     /           xfs    defaults,noatime  1   1
UID=A1F2-F73A          /boot/efi       vfat    defaults,noatime,uid=0,gid=0,umask=0077,shortname=winnt,x-systemd.automount
72.31.18.25:/runbooks    /home/ec2-user/nfs defaults 0 0
```

```
tmpfs on /tmp type tmpfs (rw,nosuid,nodev,seclabel,nr_inodes=1048576)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime,seclabel)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,seclabel,pagesize=2M)
ramfs on /run/credentials/systemd-sysctl.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
ramfs on /run/credentials/systemd-tmpfiles-setup-dev.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
systemd-1 on /boot/efi type autofs (rw,relatime,fd=52,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=13684)
ramfs on /run/credentials/systemd-tmpfiles-setup.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw,relatime)
/dev/nvme0n1p128 on /boot/efi type vfat (rw,noatime,fmask=0077,dmask=0077,codepage=437,iocharset=ascii,shortname=winnt,errors=remount-ro,x-systemd.a
omount)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=93884k,nr_inodes=23471,mode=700,uid=1000,gid=1000)
172.31.18.25:/runbooks on /home/ec2-user/nfs type nfs4 (rw,relatime,vers=4.2,rsize=131072,wsize=131072,namlen=255,hard,proto=tcp,timeo=600,retrans=2
ec=sys,clientaddr=172.31.24.121,local_lock=none,addr=172.31.18.25)
[root@ip-172-31-24-121 ec2-user]# nano /etc/fstab
[root@ip-172-31-24-121 ec2-user]# cd nfs
[root@ip-172-31-24-121 nfs]# ls
filenfs
[root@ip-172-31-24-121 nfs]#
```

i-02bf5ec92dba94028 (project2)

```
newer release of "Amazon Linux" is available.
Version 2023.10.20260120:
n "/usr/bin/dnf check-release-update" for full release and version update info
  ,     #_
  ~\_   ####_        Amazon Linux 2023
 ~~  \_#####\
 ~~     \###|
 ~~       \#/ ___    https://aws.amazon.com/linux/amazon-linux-2023
  ~~       V~' '->
   ~~~         /
     ~~._.   _/
        _/ _/
      _/m/'
st login: Wed Jan 28 15:51:09 2026 from 13.52.6.115
c2-user@ip-172-31-18-25 ~]$ sudo su
oot@ip-172-31-18-25 ec2-user]#  cd /runbooks
oot@ip-172-31-18-25 runbooks]# touch filenfs
oot@ip-172-31-18-25 runbooks]#
```

i-07838b7a5471bd3e5 (project)

```
# with the listen_ipv6 directive.
listen=NO
#
# This directive enables listening on IPv6 sockets. By default, listening
# on the IPv6 "any" address (::) will accept connections from both IPv6
# and IPv4 clients. It is not necessary to listen on *both* IPv4 and IPv6
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
# Make sure, that one of the listen options is commented !!
listen_ipv6=YES

pam_service_name=vsftpd
userlist_enable=YES
pasv_enable=YES
pasv_min_port=1024
pasv_max_port=1048
pasv_address=54.183.114.185
allow_writeable_chroot=YES
```

i-07838b7a5471bd3e5 (project)

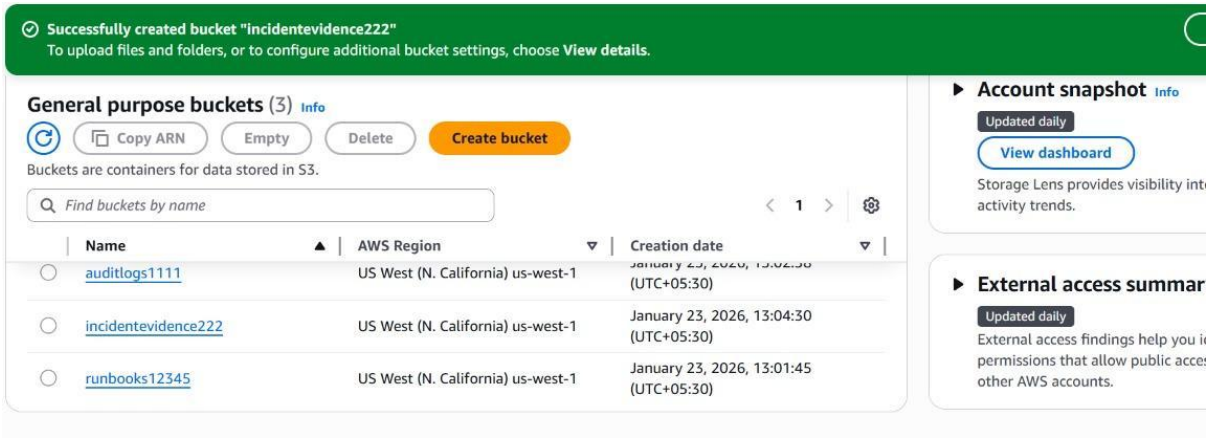
PublicIPs: 54.183.114.185 PrivateIPs: 172.31.18.25

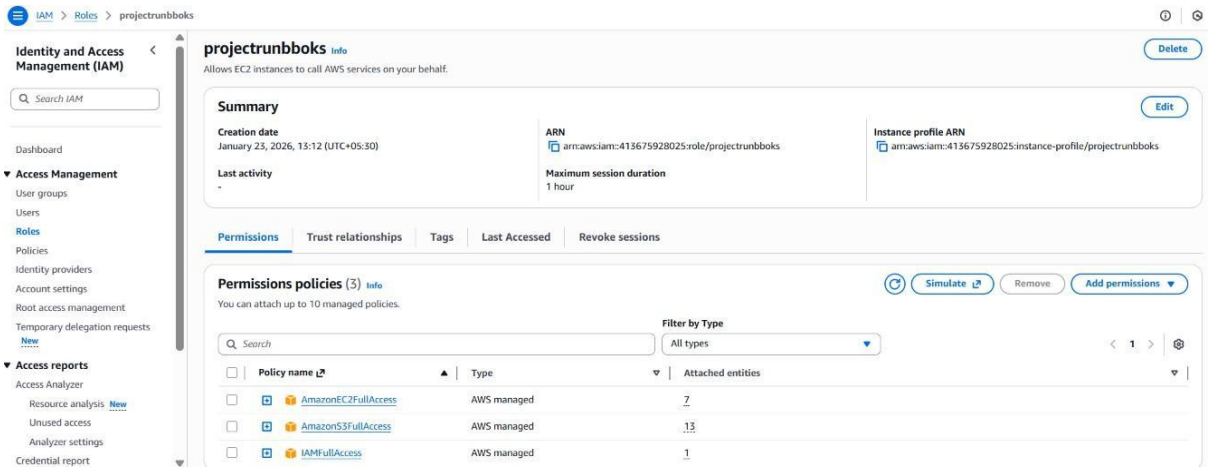
```
[root@ip-172-31-18-25 ec2-user]# sudo mkdir -p /home/ftp/upload
[root@ip-172-31-18-25 ec2-user]# ls
file3  incident_001.tar.gz  incident_buddle  incident_evidence  incident_store  secure
[root@ip-172-31-18-25 ec2-user]# chown root:root /home/ftp
[root@ip-172-31-18-25 ec2-user]# chmod 755 /home/ftp
[root@ip-172-31-18-25 ec2-user]# useradd -G engineers ftpuser
[root@ip-172-31-18-25 ec2-user]# passwd ftpuser
Changing password for user ftpuser.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@ip-172-31-18-25 ec2-user]# chown ftpuser:engineers /home/ftp/upload
[root@ip-172-31-18-25 ec2-user]# chmod 755 /home/ftp/upload
[root@ip-172-31-18-25 ec2-user]# nano /etc/vsftpd/vsftpd.conf
[root@ip-172-31-18-25 ec2-user]# systemctl start vsftpd
[root@ip-172-31-18-25 ec2-user]# systemctl enable vsftpd
```

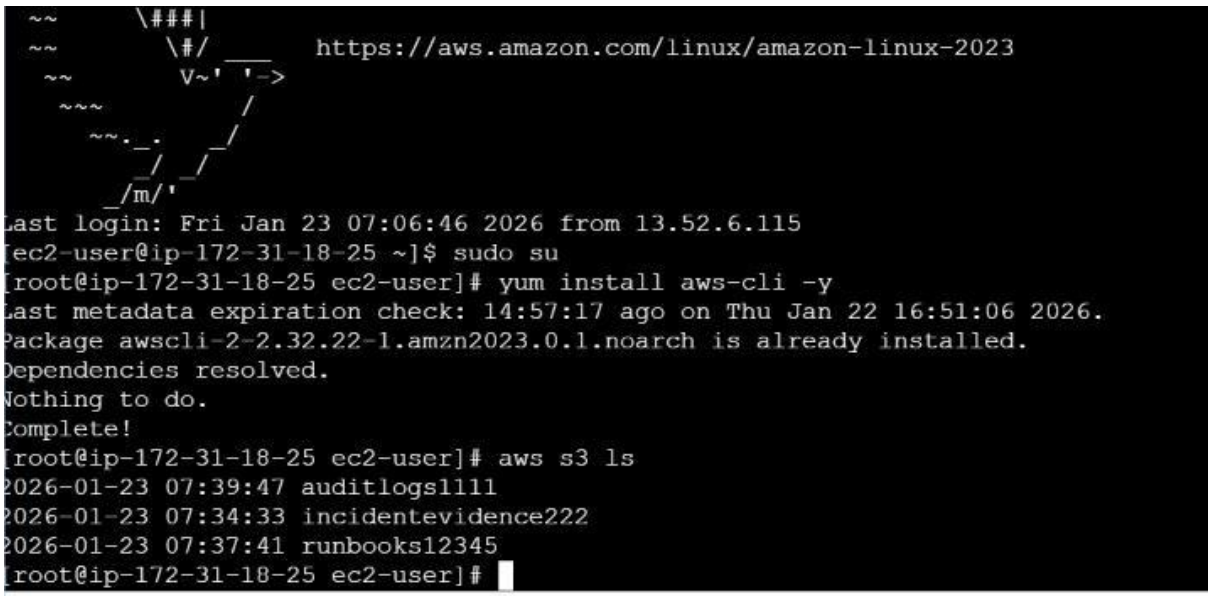
```
[root@ip-172-31-12-242 ec2-user]# touch evidence.txt
[root@ip-172-31-12-242 ec2-user]# ls
evidence.txt
[root@ip-172-31-12-242 ec2-user]# cat > evidence.txt
incidence evidence file
[root@ip-172-31-12-242 ec2-user]# lftp -u ftpuser ftp://204.236.154.216
Password:
lftp ftpuser@204.236.154.216:~> cd upload
cd ok, cwd=/upload
lftp ftpuser@204.236.154.216:/upload> put evidence.txt
24 bytes transferred
lftp ftpuser@204.236.154.216:/upload> ls
-rw-r--r--    1 1007     1010           24 Feb 02 07:54 evidence.txt
lftp ftpuser@204.236.154.216:/upload> get evidence.txt
get: /home/ec2-user/evidence.txt: file already exists and xfer:clobber is unset
lftp ftpuser@204.236.154.216:/upload>
```

5.10 CRON JOB AUTOMATION LOGIC

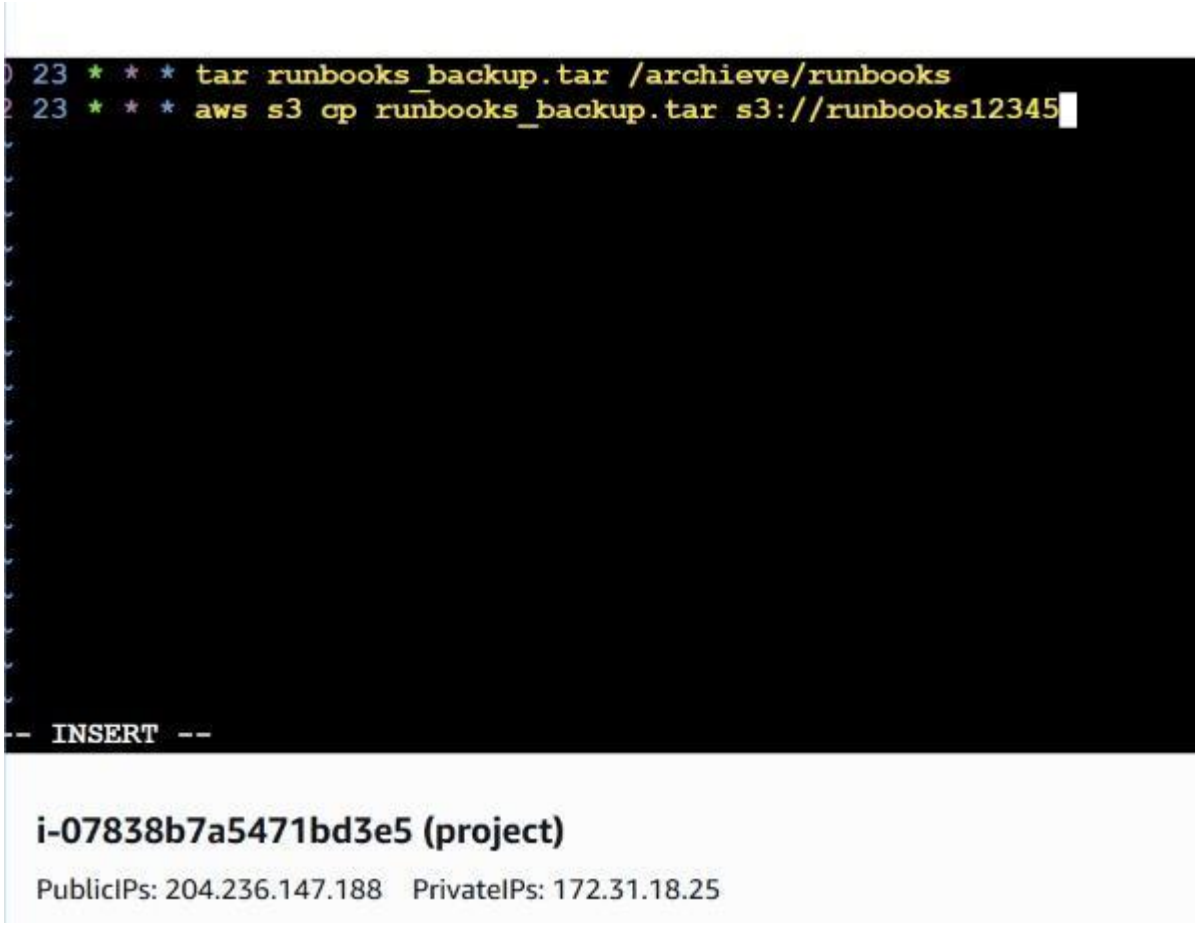
- Cron jobs were configured for automation:
    - Daily runbook backup to S3
    - Weekly audit log upload
    - Monthly evidence archival
- Each cron job was configured under the appropriate user account.
- Automation reduced manual effort and ensured consistency.




IAM > Roles > projectrunbboks
Identity and Access Management (IAM)
Search IAM
Dashboard
Access Management
User groups
Users
Roles
Policies
Identity providers
Account settings
Root access management
Temporary delegation requests
New
Access reports
Access Analyzer
Resource analysis New
Unused access
Analyzer settings
Credential report
projectrunbboks Info
Allows EC2 instances to call AWS services on your behalf.
Delete
Summary
Edit
Creation date
January 23, 2026, 13:12 (UTC+05:30)
ARN
arn:aws:iam::413675928025:role/projectrunbboks
Instance profile ARN
arn:aws:iam::413675928025:instance-profile/projectrunbboks
Last activity
-
Maximum session duration
1 hour
Permissions
Trust relationships
Tags
Last Accessed
Revoke sessions
Permissions policies (3) Info
You can attach up to 10 managed policies.
Simulate
Remove
Add permissions
Filter by Type
Search
All types
Policy name
Type
Attached entities
AmazonEC2FullAccess
AWS managed
7
AmazonS3FullAccess
AWS managed
13
IAMFullAccess
AWS managed
1


https://aws.amazon.com/linux/amazon-linux-2023
Last login: Fri Jan 23 07:06:46 2026 from 13.52.6.115
[ec2-user@ip-172-31-18-25 ~]$ sudo su
[root@ip-172-31-18-25 ec2-user]# yum install aws-cli -y
Last metadata expiration check: 14:57:17 ago on Thu Jan 22 16:51:06 2026.
Package awscli-2-2.32.22-1.amzn2023.0.1.noarch is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-18-25 ec2-user]# aws s3 ls
2026-01-23 07:39:47 auditlogs1111
2026-01-23 07:34:33 incidentevidence222
2026-01-23 07:37:41 runbooks12345
[root@ip-172-31-18-25 ec2-user]#


i-07838b7a5471bd3e5 (project)
PublicIPs: 204.236.147.188 PrivateIPs: 172.31.18.25


23 * * * tar runbooks_backup.tar /archieve/runbooks
23 * * * aws s3 cp runbooks_backup.tar s3://runbooks12345
-- INSERT --
i-07838b7a5471bd3e5 (project)
PublicIPs: 204.236.147.188 PrivateIPs: 172.31.18.25

```
) 18 * * 0 aws s3 cp /auditlogs s3://auditlogs1111
```

:wq

i-07838b7a5471bd3e5 (project)

PublicIPs: 204.236.147.188 PrivateIPs: 172.31.18.25

CloudShell Feedback Console Mobile App

```
[root@ip-172-31-18-25 ec2-user]# cd /auditlogs
[root@ip-172-31-18-25 auditlogs]# pwd
/auditlogs
[root@ip-172-31-18-25 auditlogs]# su audit1
[audit1@ip-172-31-18-25 auditlogs]$ crontab -e
crontab: installing new crontab
[audit1@ip-172-31-18-25 auditlogs]$
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 204.236.147.188 PrivateIPs: 172.31.18.25

```
30 0 1 * * tar -cvf incident_monthly_backup.tar /home/ec2-user/incident_evidence
32 0 1 * * aws s3 cp incident_monthly_backup.tar s3://incidentevidence222
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq
```

i-07838b7a5471bd3e5 (project)

PublicIPs: 204.236.147.188 PrivateIPs: 172.31.18.25

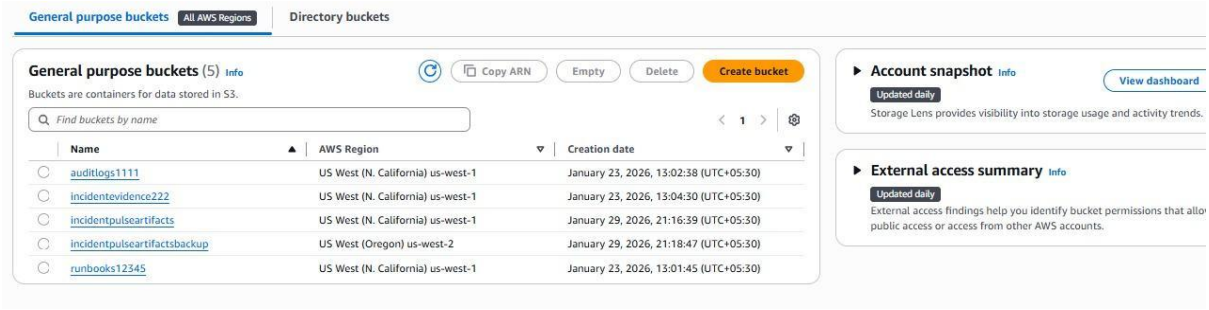
```
[root@ip-172-31-18-25 incident_evidence]# pwd
/home/ec2-user/incident_evidence
[root@ip-172-31-18-25 incident_evidence]# cd ..
[root@ip-172-31-18-25 ec2-user]# su lead2
[lead2@ip-172-31-18-25 ec2-user]$ crontab -e
no crontab for lead2 - using an empty one
crontab: installing new crontab
[lead2@ip-172-31-18-25 ec2-user]$ exit
exit
[root@ip-172-31-18-25 ec2-user]#
```

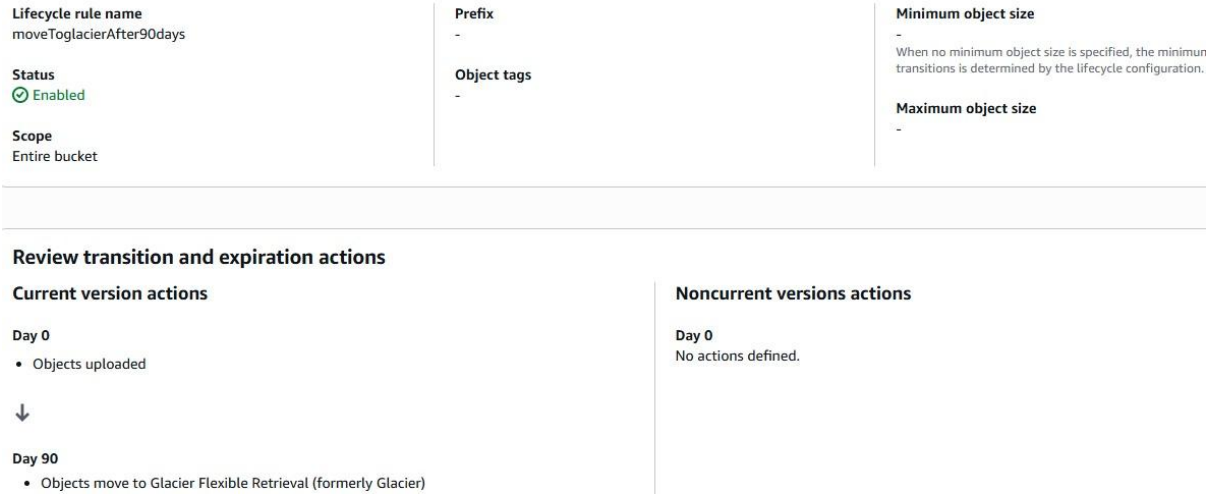
i-07838b7a5471bd3e5 (project)

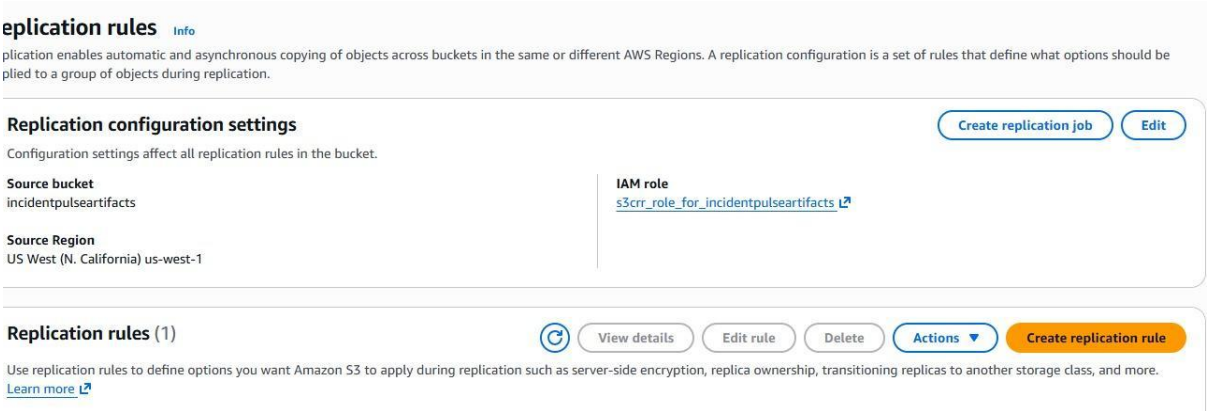
PublicIPs: 204.236.147.188 PrivateIPs: 172.31.18.25

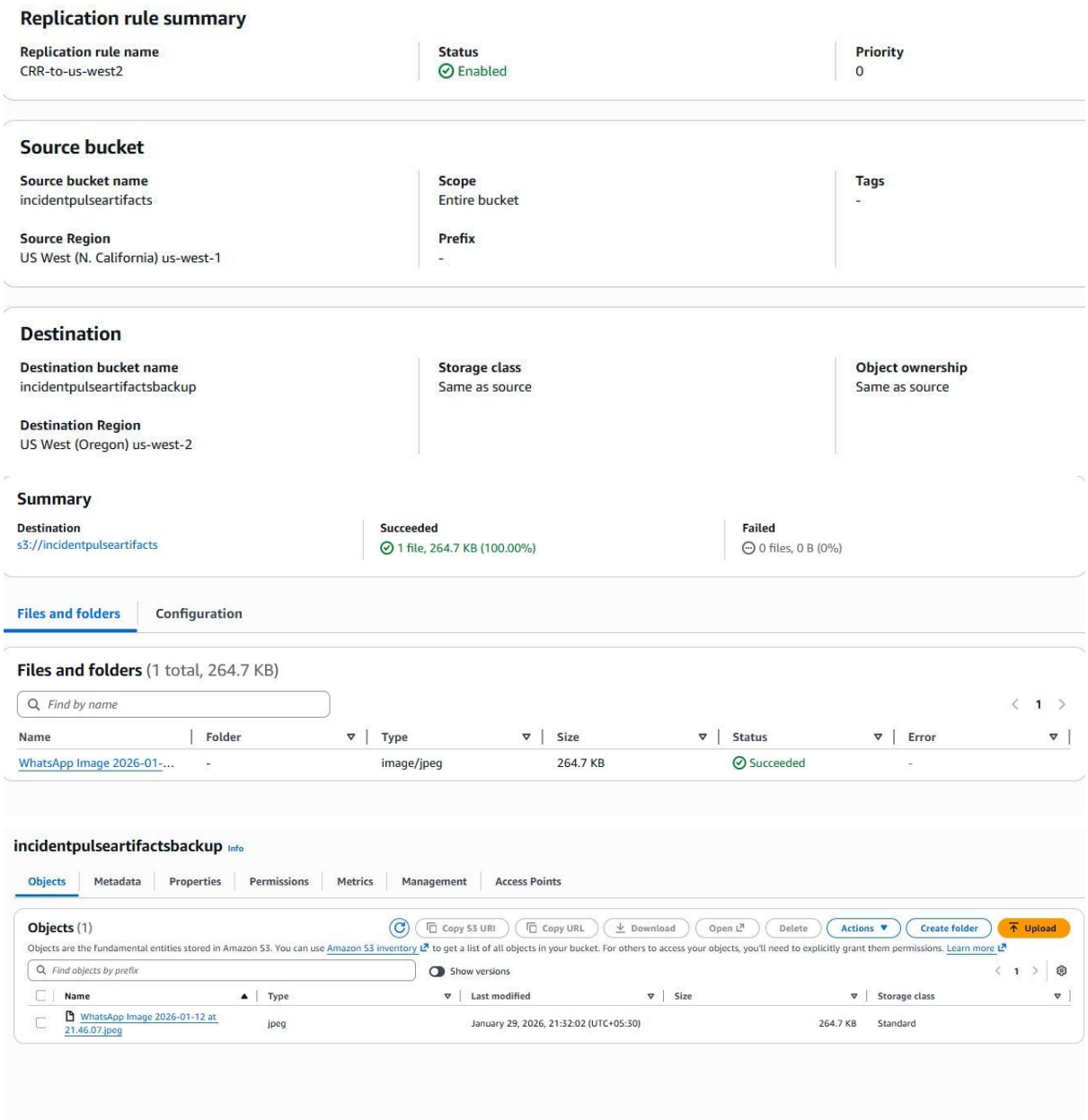
5.11 S3 STORAGE AND LIFECYCLE MANAGEMENT

- S3 bucket incidentpulse-artifacts was created.
- Versioning was enabled to preserve historical data.
- Lifecycle rules were applied to move old data to Glacier.
- Cross-region replication ensured disaster recovery.



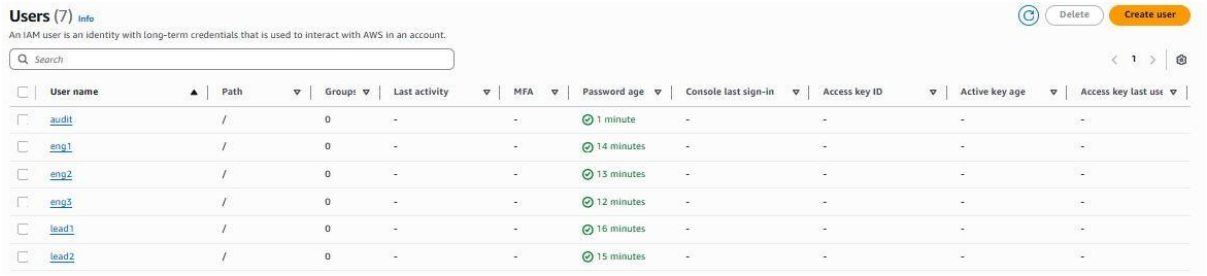


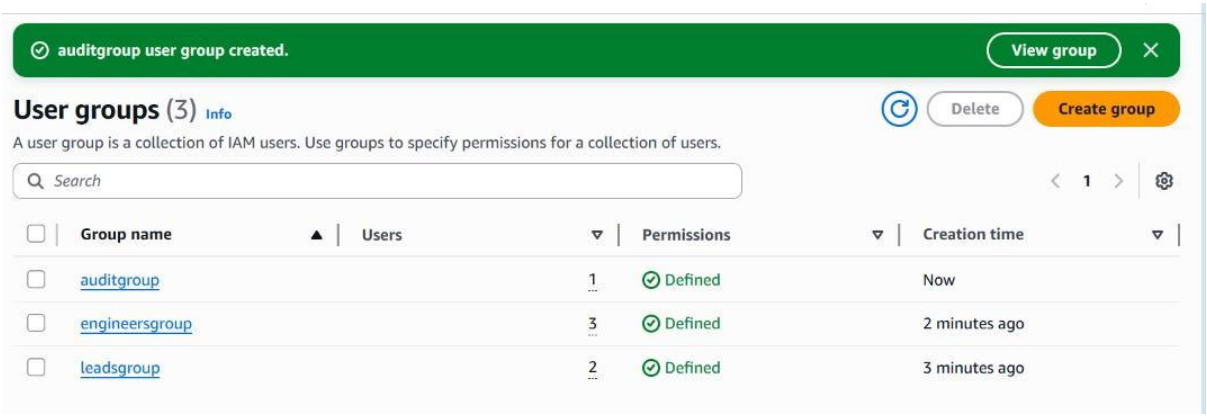


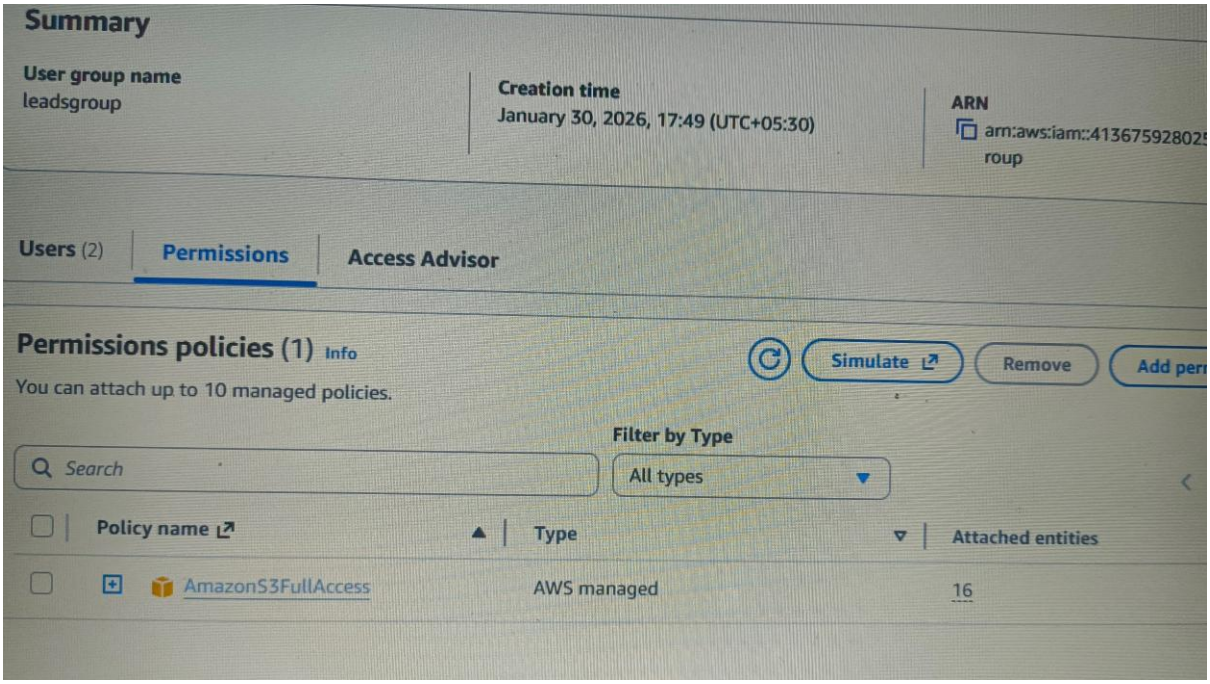


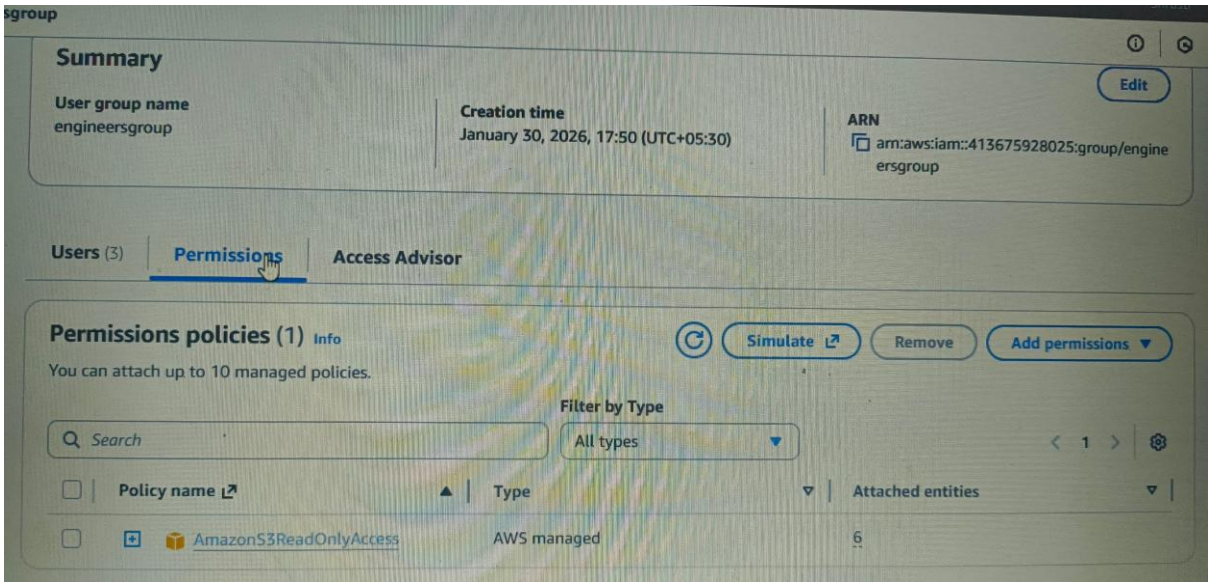
5.12 IAM ROLE AND POLICY CONFIGURATION

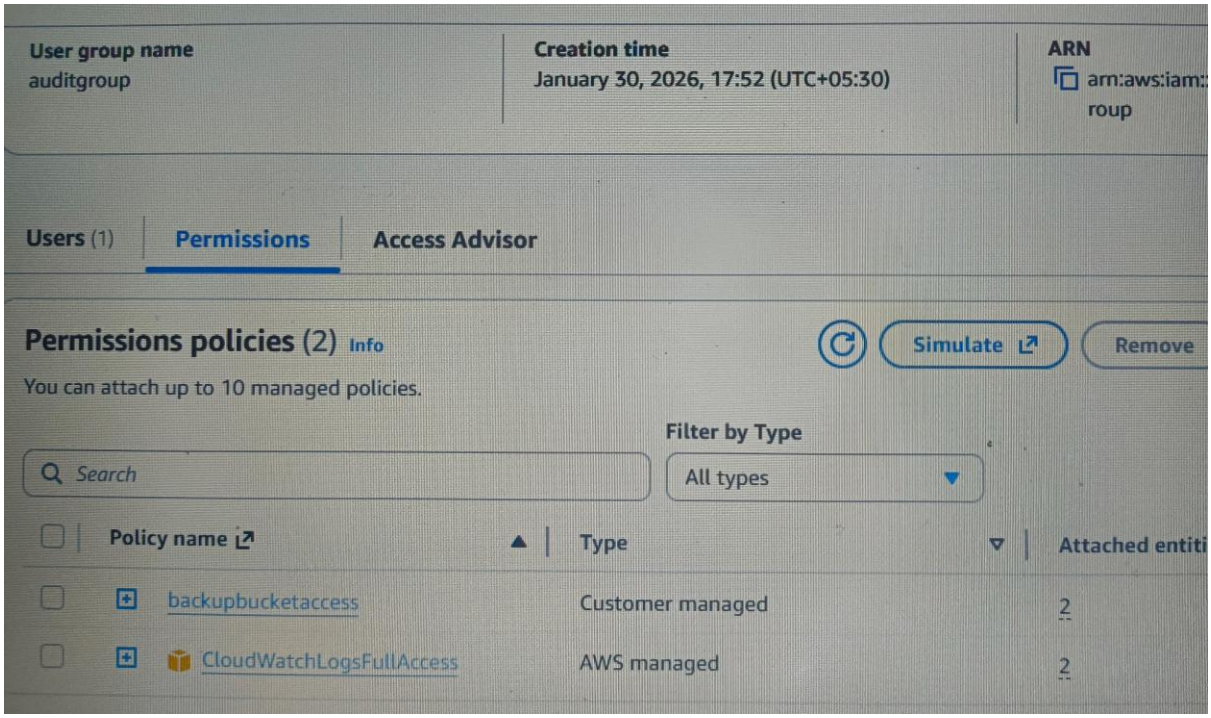
- IAM groups were created based on roles.
- Least privilege access was enforced.
- Policies were tested by logging in with each role.
- Unauthorized access attempts were blocked.


Users (7) Info
An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.
Delete
Create user
Search
User name
Path
Groups
Last activity
MFA
Password age
Console last sign-in
Access key ID
Active key age
Access key last use
audit / 0 - - 1 minute - - - -
eng1 / 0 - - 14 minutes - - - -
eng2 / 0 - - 13 minutes - - - -
eng3 / 0 - - 12 minutes - - - -
lead1 / 0 - - 16 minutes - - - -
lead2 / 0 - - 15 minutes - - - -


auditgroup user group created.
View group
User groups (3) Info
A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.
Delete
Create group
Search
Group name
Users
Permissions
Creation time
auditgroup 1 Defined Now
engineersgroup 3 Defined 2 minutes ago
leadsgroup 2 Defined 3 minutes ago

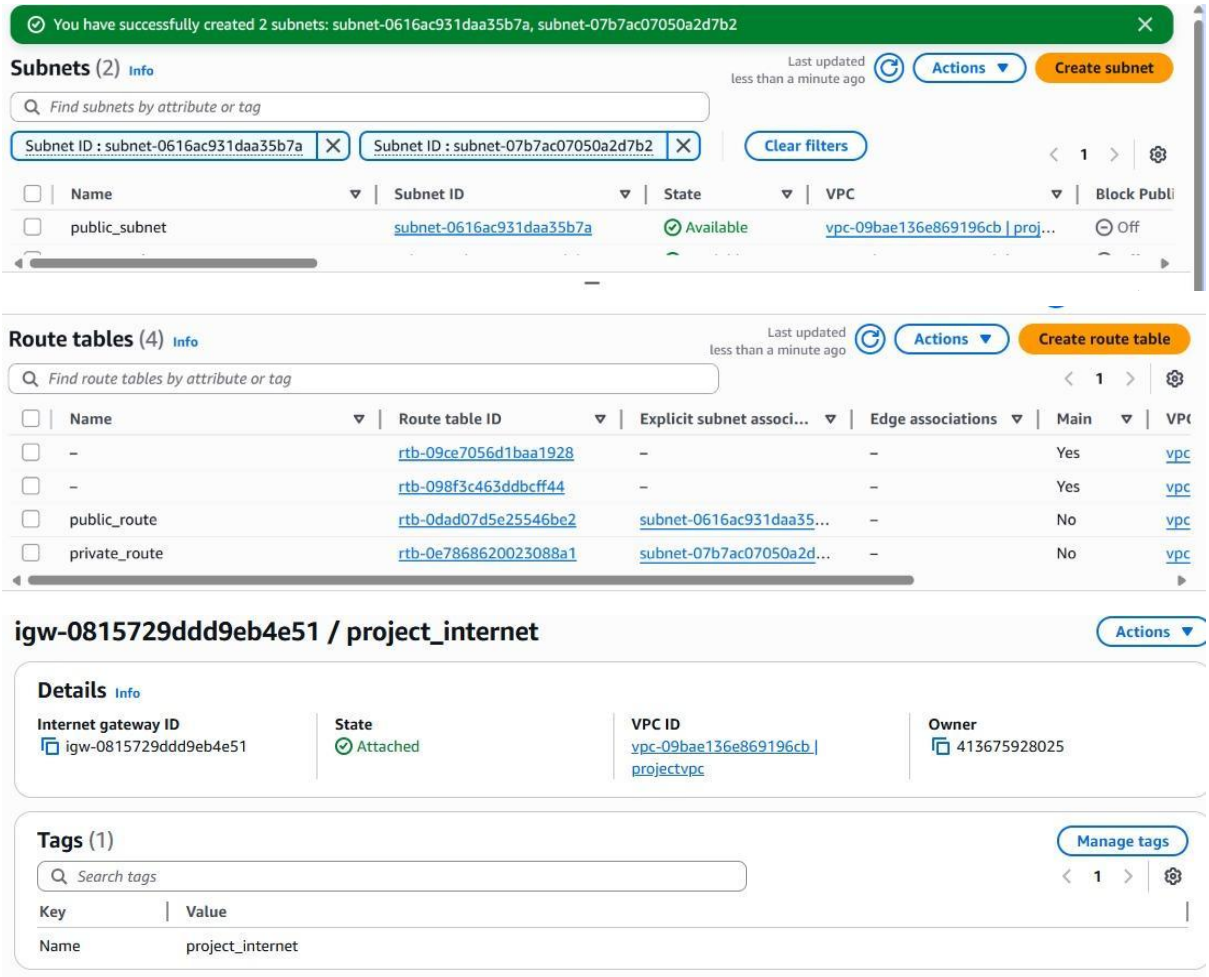

Summary
User group name
leadsgroup
Creation time
January 30, 2026, 17:49 (UTC+05:30)
ARN
Users (2)
Permissions
Access Advisor
Permissions policies (1) Info
You can attach up to 10 managed policies.
Simulate
Remove
Filter by Type
Search
All types
Policy name
Type
Attached entities
AmazonS3FullAccess
AWS managed
16

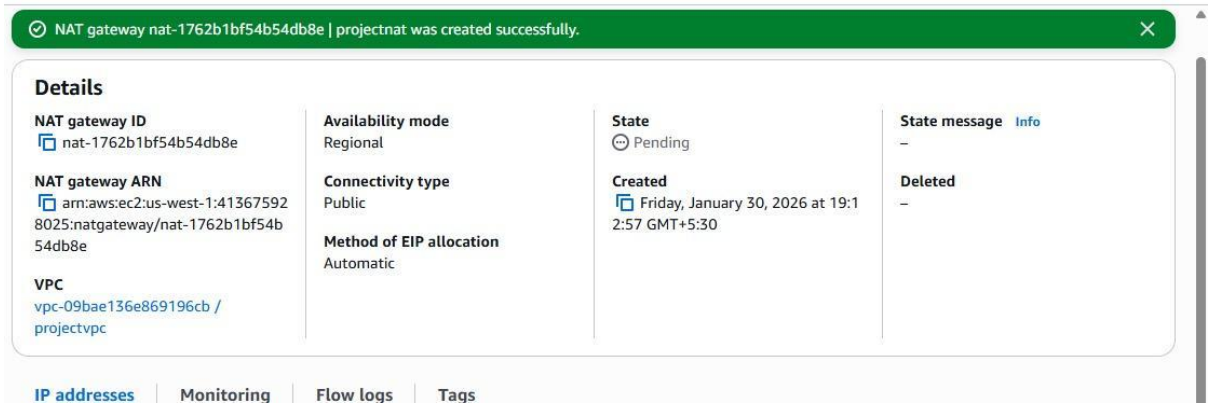

sgroup
Summary
Edit
User group name
engineersgroup
Creation time
January 30, 2026, 17:50 (UTC+05:30)
ARN
arn:aws:iam::413675928025:group/engine
ersgroup
Users (3)
Permissions
Access Advisor
Permissions policies (1) Info
You can attach up to 10 managed policies.
Simulate
Remove
Add permissions
Filter by Type
Search
All types
1
Policy name
Type
Attached entities
AmazonS3ReadOnlyAccess
AWS managed
6

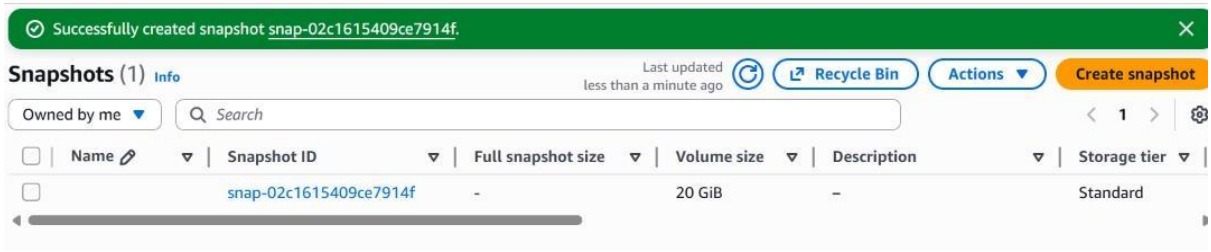

User group name
auditgroup
Creation time
January 30, 2026, 17:52 (UTC+05:30)
ARN
arn:aws:iam::
roup
Users (1)
Permissions
Access Advisor
Permissions policies (2) Info
You can attach up to 10 managed policies.
Simulate
Remove
Filter by Type
Search
All types
Policy name
Type
Attached entiti
backupbucketaccess
Customer managed
2
CloudWatchLogsFullAccess
AWS managed
2

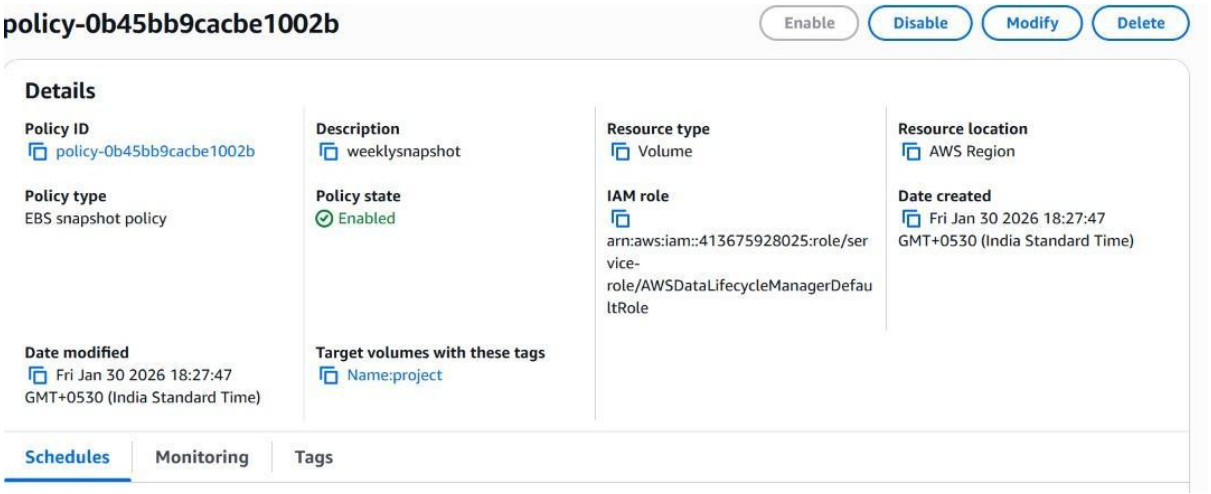
5.13 AWS VPC AND NETWORK SETUP

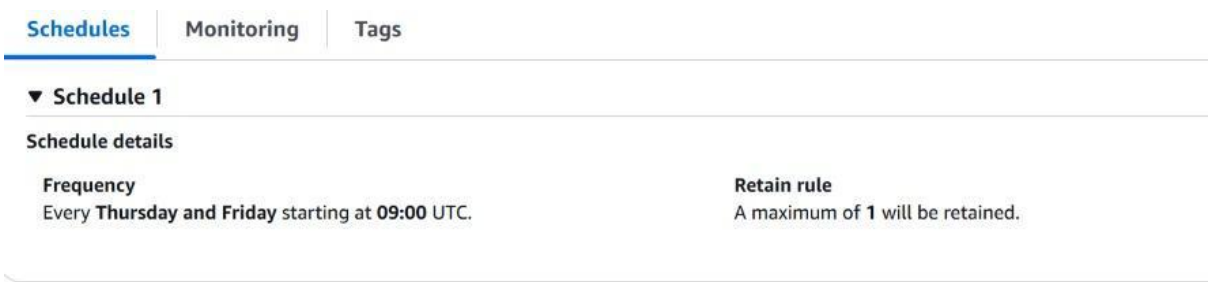
1. A dedicated **VPC** was created to host the IncidentPulse infrastructure.
2. Two subnets were configured:
    - **Public Subnet** for internet-facing components such as Load Balancer and Bastion/Public EC2.
    - **Private Subnet** for backend resources like application EC2 instances and RDS.
3. An **Internet Gateway** was attached to the VPC and associated with the public route table.
4. A **NAT Gateway** was configured in the public subnet to allow private subnet instances to access the internet securely.
5. Route tables were updated to ensure proper traffic flow between public and private subnets.

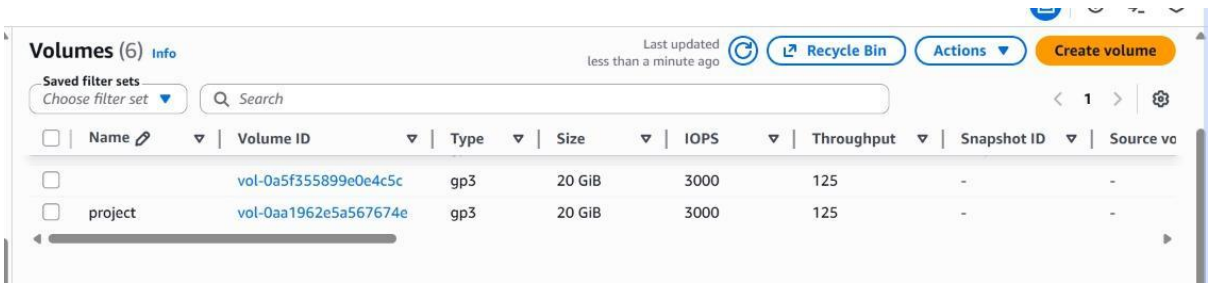


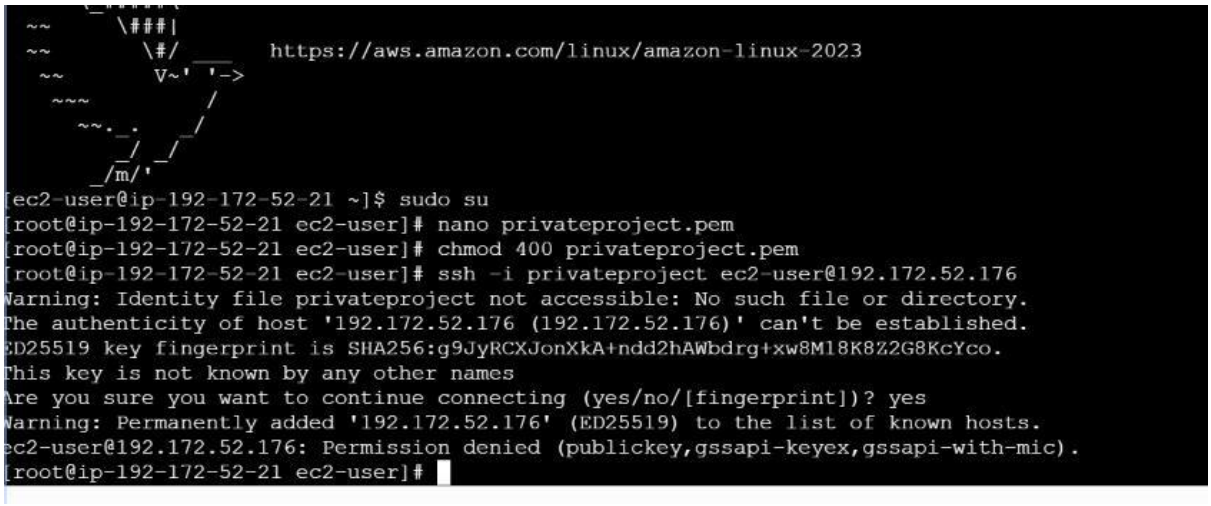

NAT gateway nat-1762b1bf54b54db8e | projectnat was created successfully.
Details
NAT gateway ID
nat-1762b1bf54b54db8e
NAT gateway ARN
arn:aws:ec2:us-west-1:413675928025:natgateway/nat-1762b1bf54b54db8e
VPC
vpc-09bae136e869196cb / projectvpc
Availability mode
Regional
Connectivity type
Public
Method of EIP allocation
Automatic
State
Pending
Created
Friday, January 30, 2026 at 19:12:57 GMT+5:30
State message
Info
Deleted
IP addresses
Monitoring
Flow logs
Tags


Successfully created snapshot snap-02c1615409ce7914f.
Snapshots (1) Info
Last updated less than a minute ago
Recycle Bin
Actions
Create snapshot
Owned by me
Search
Name
Snapshot ID
Full snapshot size
Volume size
Description
Storage tier
snap-02c1615409ce7914f
-
20 GiB
-
Standard

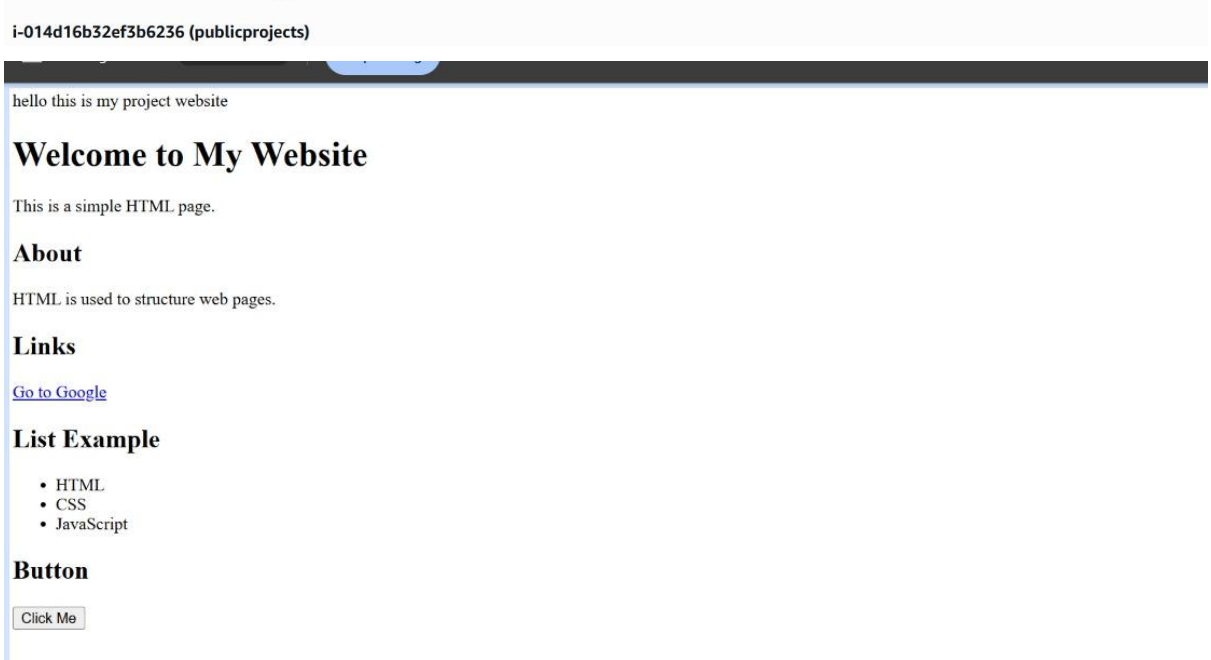

policy-0b45bb9cacbe1002b
Enable
Disable
Modify
Delete
Details
Policy ID
policy-0b45bb9cacbe1002b
Policy type
EBS snapshot policy
Description
weeklysnapshot
Policy state
Enabled
Resource type
Volume
IAM role
arn:aws:iam::413675928025:role/service-role/AWSDataLifecycleManagerDefaultRole
Resource location
AWS Region
Date created
Fri Jan 30 2026 18:27:47 GMT+0530 (India Standard Time)
Date modified
Fri Jan 30 2026 18:27:47 GMT+0530 (India Standard Time)
Target volumes with these tags
Name:project
Schedules
Monitoring
Tags


Schedules
Monitoring
Tags
Schedule 1
Schedule details
Frequency
Every Thursday and Friday starting at 09:00 UTC.
Retain rule
A maximum of 1 will be retained.


Volumes (6) Info
Last updated less than a minute ago
Recycle Bin
Actions
Create volume
Saved filter sets
Choose filter set
Search
Name
Volume ID
Type
Size
IOPS
Throughput
Snapshot ID
Source vo
vol-0a5f355899e0e4c5c
gp3
20 GiB
3000
125
project
vol-0aa1962e5a567674e
gp3
20 GiB
3000
125

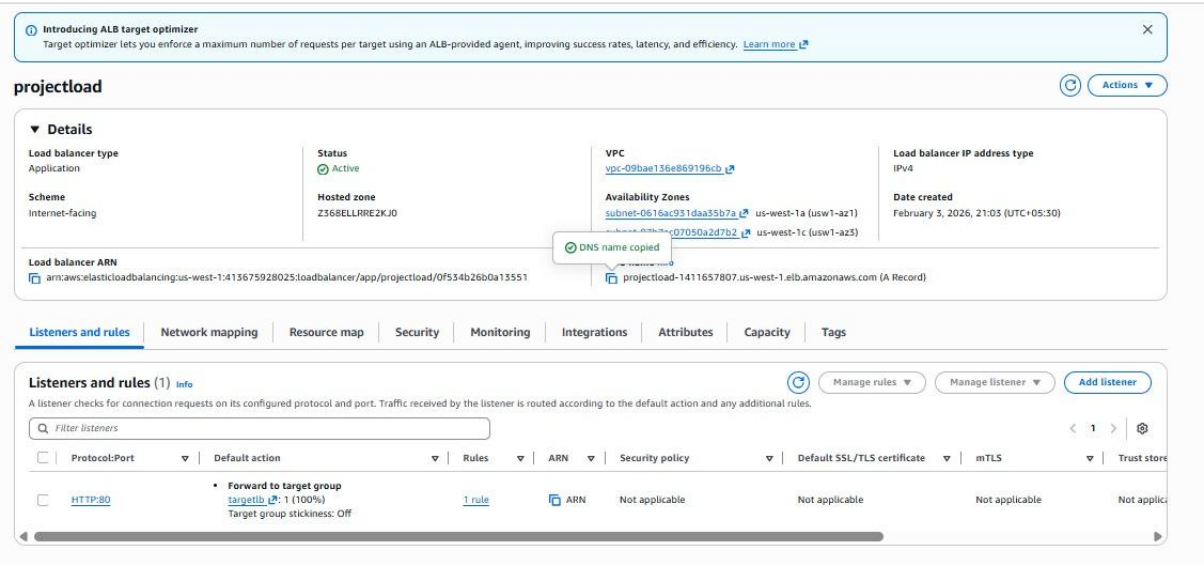

https://aws.amazon.com/linux/amazon-linux-2023
[ec2-user@ip-192-172-52-21 ~]$ sudo su
[root@ip-192-172-52-21 ec2-user]# nano privateproject.pem
[root@ip-192-172-52-21 ec2-user]# chmod 400 privateproject.pem
[root@ip-192-172-52-21 ec2-user]# ssh -i privateproject ec2-user@192.172.52.176
Warning: Identity file privateproject not accessible: No such file or directory.
The authenticity of host '192.172.52.176 (192.172.52.176)' can't be established.
ED25519 key fingerprint is SHA256:g9JyRCXJonXkA+ndd2hAWbdrg+xw8M18K8Z2G8KcYco.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.172.52.176' (ED25519) to the list of known hosts.
ec2-user@192.172.52.176: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[root@ip-192-172-52-21 ec2-user]#

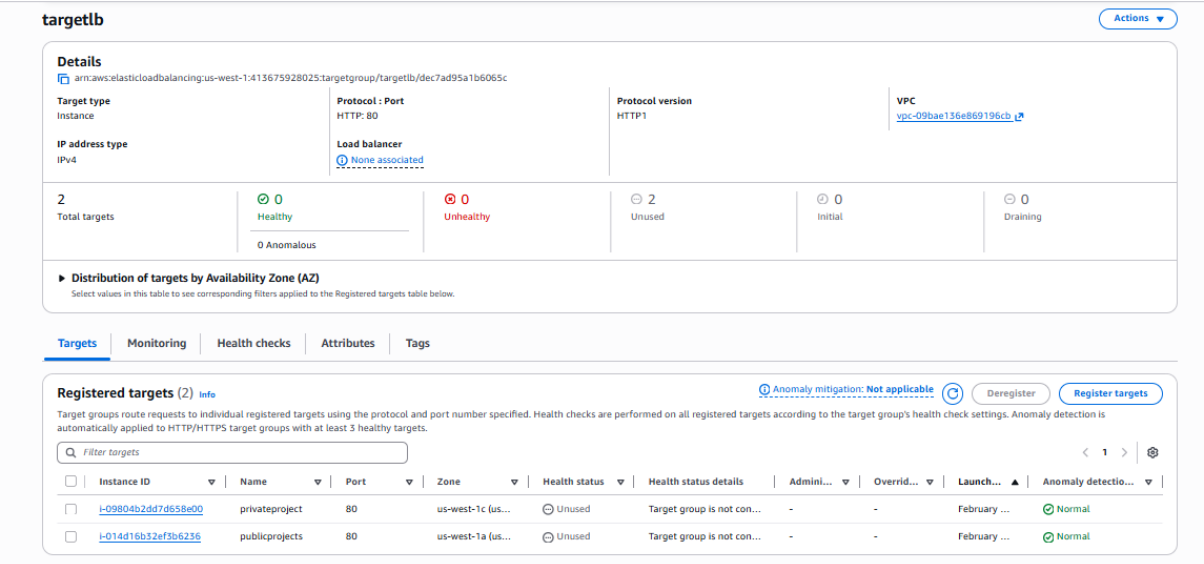

apr-util-openssl-1.6.3-1.amzn2023.0.2.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
httpd-2.4.66-1.amzn2023.0.1.x86_64
httpd-core-2.4.66-1.amzn2023.0.1.x86_64
httpd-filesystem-2.4.66-1.amzn2023.0.1.noarch
httpd-tools-2.4.66-1.amzn2023.0.1.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch
mod_http2-2.0.27-1.amzn2023.0.3.x86_64
mod_lua-2.4.66-1.amzn2023.0.1.x86_64
omplete!
root@ip-192-172-52-21 ec2-user]# systemctl start httpd
root@ip-192-172-52-21 ec2-user]# systemctl enable httpd
reated symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
root@ip-192-172-52-21 ec2-user]# pwd
home/ec2-user
root@ip-192-172-52-21 ec2-user]# cd /var/www/html
root@ip-192-172-52-21 html]# nano index.html
root@ip-192-172-52-21 html]# cd /home/ec2-user
root@ip-192-172-52-21 ec2-user]# systemctl restart httpd
root@ip-192-172-52-21 ec2-user]#

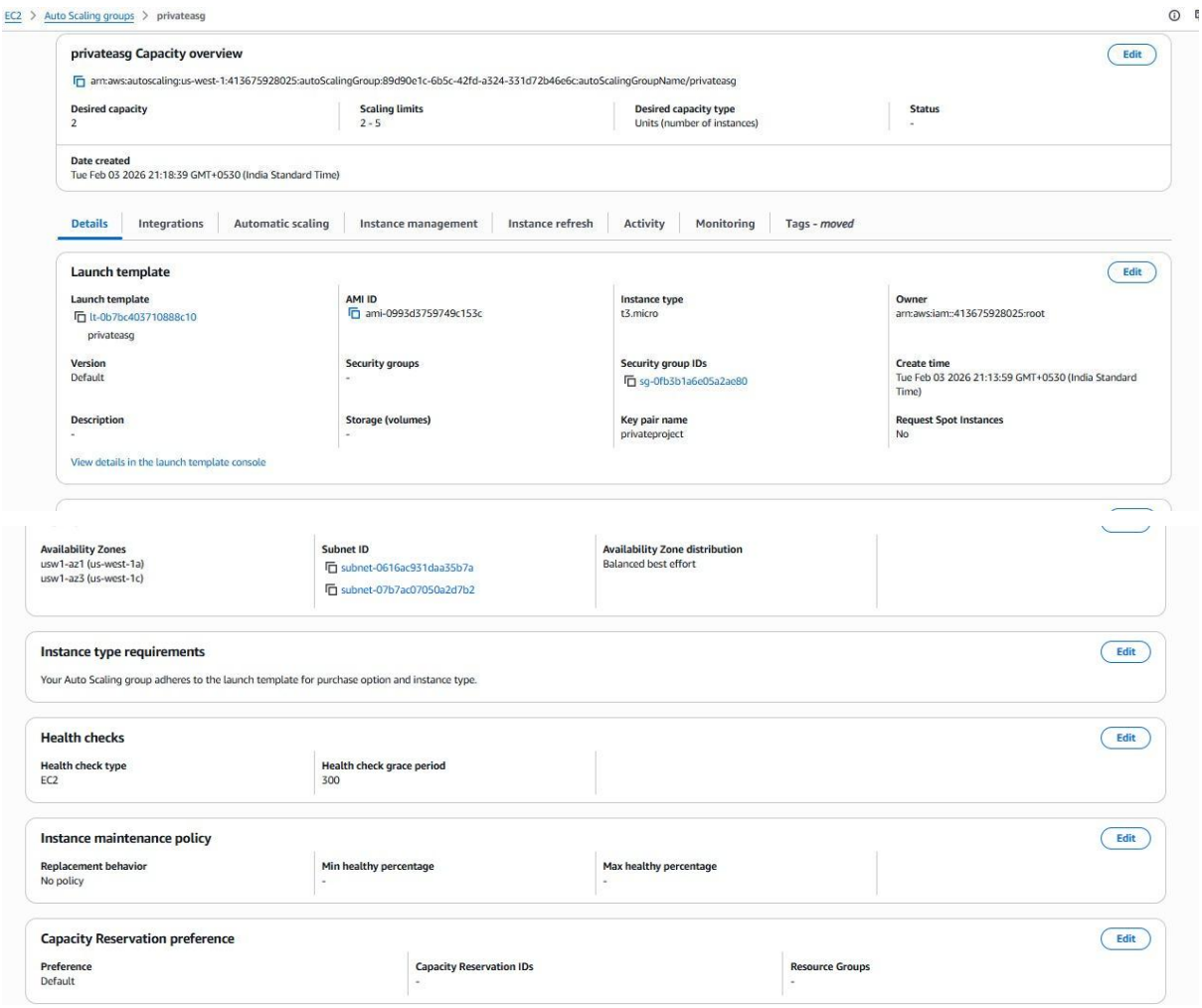

i-014d16b32ef3b6236 (publicprojects)
hello this is my project website
Welcome to My Website
This is a simple HTML page.
About
HTML is used to structure web pages.
Links
Go to Google
List Example
HTML
CSS
JavaScript
Button
Click Me

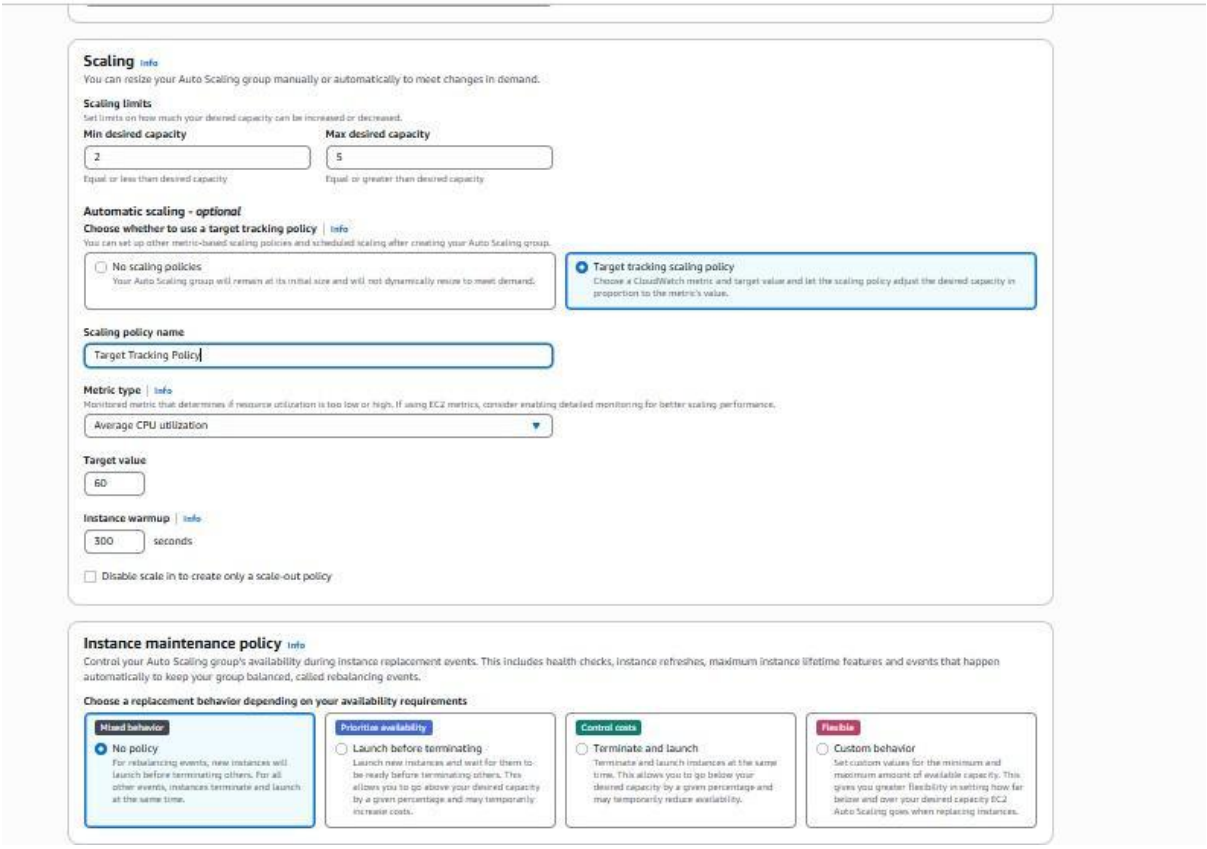
5.14 LOAD BALANCING AND AUTO SCALING LOGIC

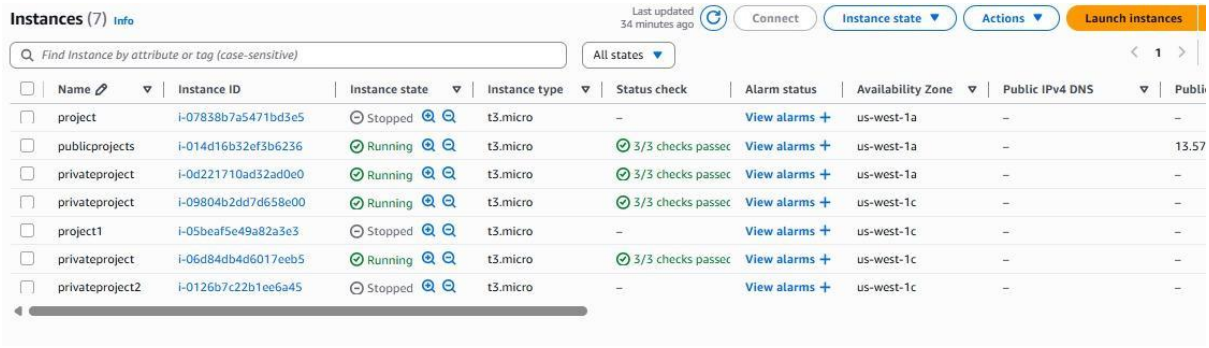
- Application Load Balancer distributed traffic across EC2 instances.
- Auto Scaling Group monitored CPU utilization.
- Scaling logic:
    - CPU > 60% for 5 minutes → scale out
- This ensured high availability and performance.





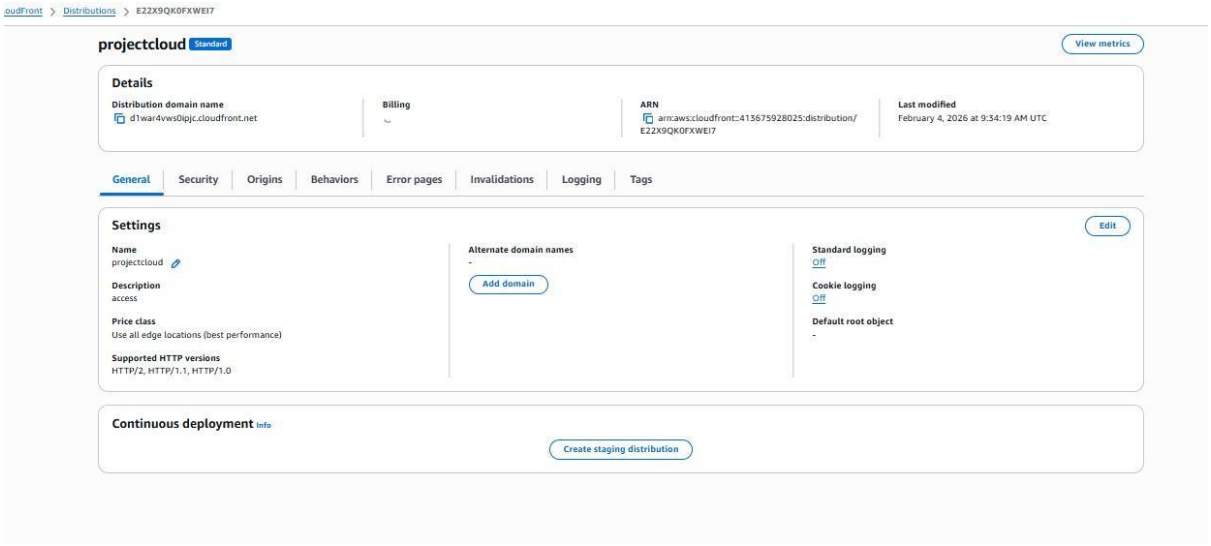

EC2 > Auto Scaling groups > privateasg
privateasg Capacity overview
Edit
arn:aws:autoscaling:us-west-1:413675928025:autoScalingGroup:89d90e1c-6b5c-42fd-a324-331d72b46e6c:autoScalingGroupName/privateasg
Desired capacity
2
Scaling limits
2 - 5
Desired capacity type
Units (number of instances)
Status
-
Date created
Tue Feb 03 2026 21:18:39 GMT+0530 (India Standard Time)
Details
Integrations
Automatic scaling
Instance management
Instance refresh
Activity
Monitoring
Tags - moved
Launch template
Edit
Launch template
lt-0b7bc403710888c10
privateasg
AMI ID
ami-0993d3759749c153c
Instance type
t3.micro
Owner
arn:aws:iam::413675928025:root
Version
Default
Security groups
-
Security group IDs
sg-0fb3b1a6e05a2ae80
Create time
Tue Feb 03 2026 21:13:59 GMT+0530 (India Standard Time)
Description
-
Storage (volumes)
-
Key pair name
privateproject
Request Spot Instances
No
View details in the launch template console
Availability Zones
usw1-az1 (us-west-1a)
usw1-az3 (us-west-1c)
Subnet ID
subnet-0616ac931daa35b7a
subnet-07b7ac07050a2d7b2
Availability Zone distribution
Balanced best effort
Instance type requirements
Edit
Your Auto Scaling group adheres to the launch template for purchase option and instance type.
Health checks
Edit
Health check type
EC2
Health check grace period
300
Instance maintenance policy
Edit
Replacement behavior
No policy
Min healthy percentage
-
Max healthy percentage
-
Capacity Reservation preference
Edit
Preference
Default
Capacity Reservation IDs
-
Resource Groups
-

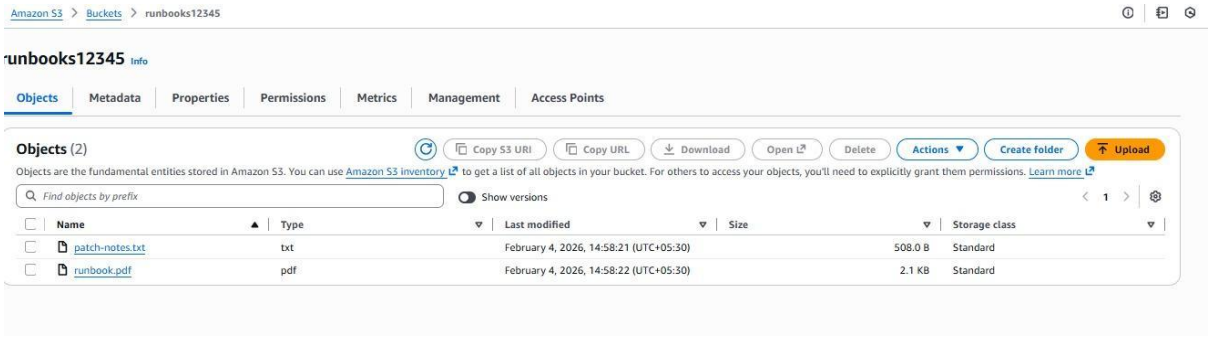

Scaling Info
You can resize your Auto Scaling group manually or automatically to meet changes in demand.
Scaling limits
Min desired capacity
2
Max desired capacity
5
Automatic scaling - optional
Choose whether to use a target tracking policy | Info
No scaling policies
Target tracking scaling policy
Scaling policy name
Target Tracking Policy
Metric type | Info
Average CPU utilization
Target value
60
Instance warmup | Info
300
seconds
Disable scale in to create only a scale-out policy
Instance maintenance policy Info
Control your Auto Scaling group's availability during instance replacement events. This includes health checks, instance refreshes, maximum instance lifetime features and events that happen automatically to keep your group balanced, called rebalancing events.
Choose a replacement behavior depending on your availability requirements
Mixed behavior
No policy
Prioritize availability
Launch before terminating
Control costs
Terminate and launch
Flexible
Custom behavior

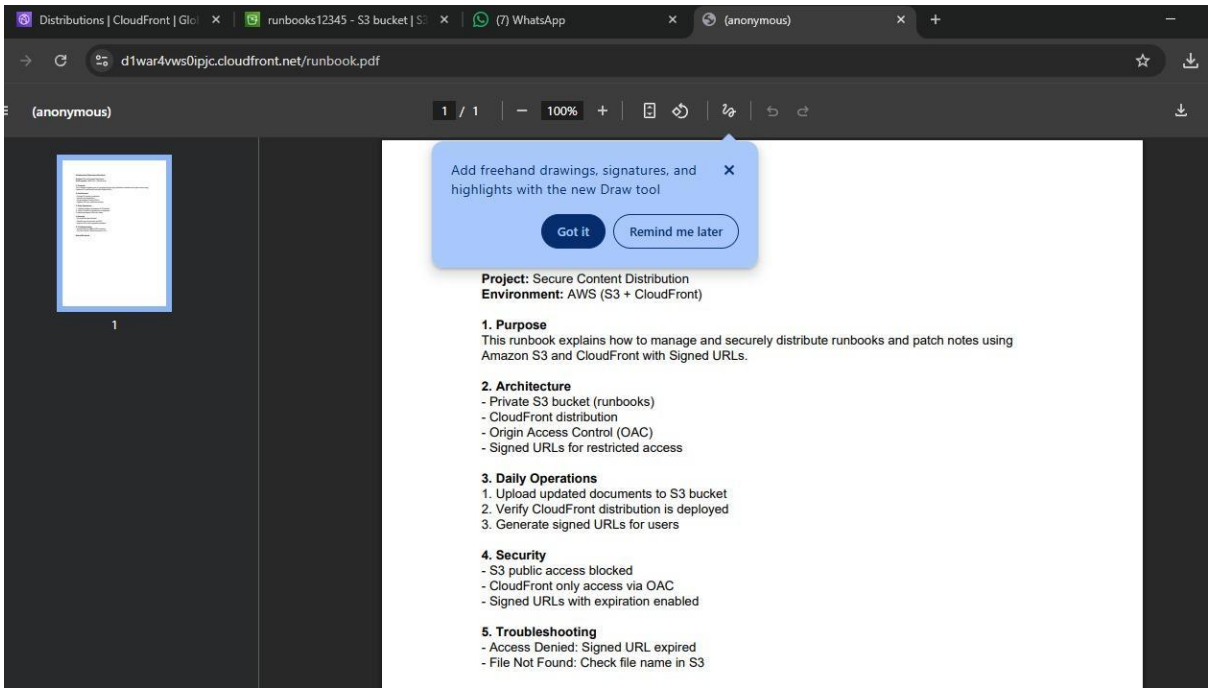

Instances (7) Info
Last updated 34 minutes ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 DNS
project | i-07838b7a5471bd3e5 | Stopped | t3.micro | – | View alarms | us-west-1a
publicprojects | i-014d16b32ef3b6236 | Running | t3.micro | 3/3 checks passed | View alarms | us-west-1a
privateproject | i-0d221710ad32ad0e0 | Running | t3.micro | 3/3 checks passed | View alarms | us-west-1a
privateproject | i-09804b2dd7d658e00 | Running | t3.micro | 3/3 checks passed | View alarms | us-west-1c
project1 | i-05beaf5e49a82a3e3 | Stopped | t3.micro | – | View alarms | us-west-1c
privateproject | i-06d84db4d6017eeb5 | Running | t3.micro | 3/3 checks passed | View alarms | us-west-1c
privateproject2 | i-0126b7c22b1ee6a45 | Stopped | t3.micro | – | View alarms | us-west-1c

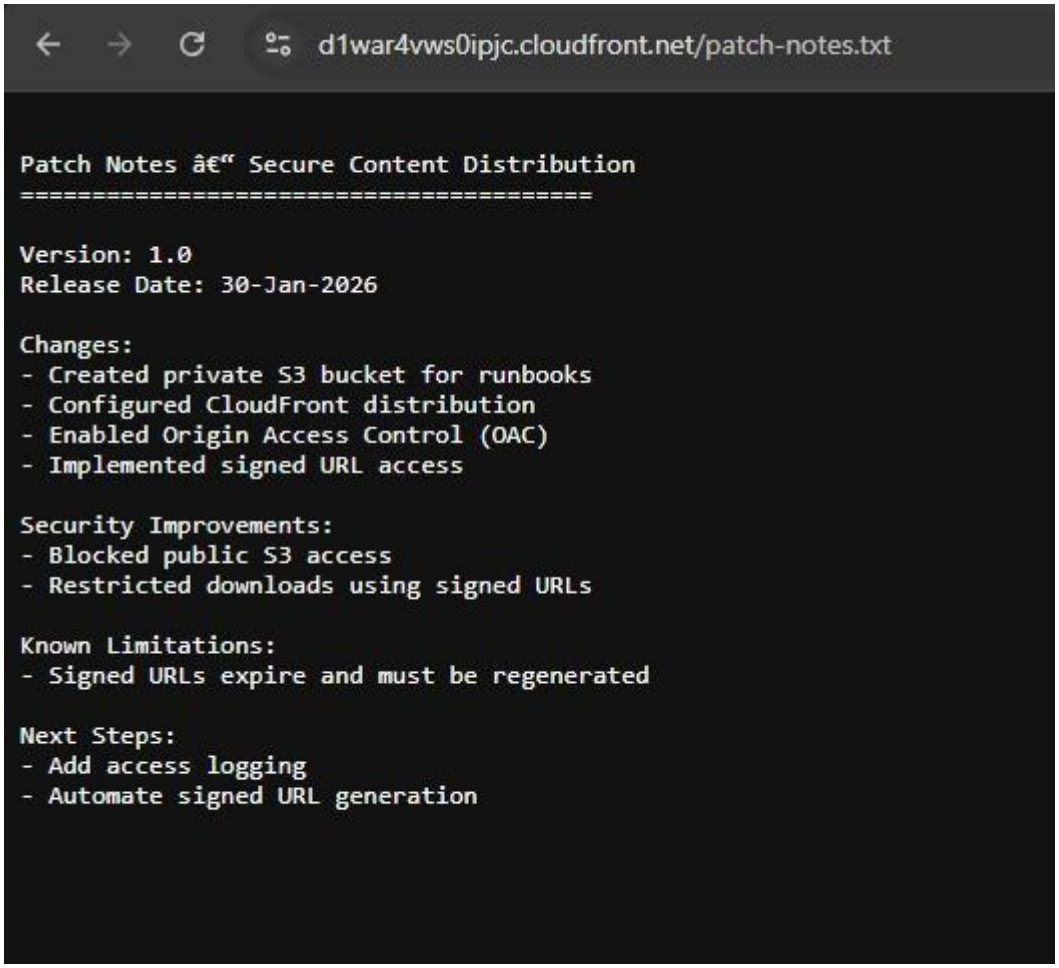
5.15 CLOUDFRONT CONFIGURATION

- CloudFront was configured to distribute runbooks securely.
- Signed URLs restricted access to authorized users.
- Caching improved performance and reduced load.



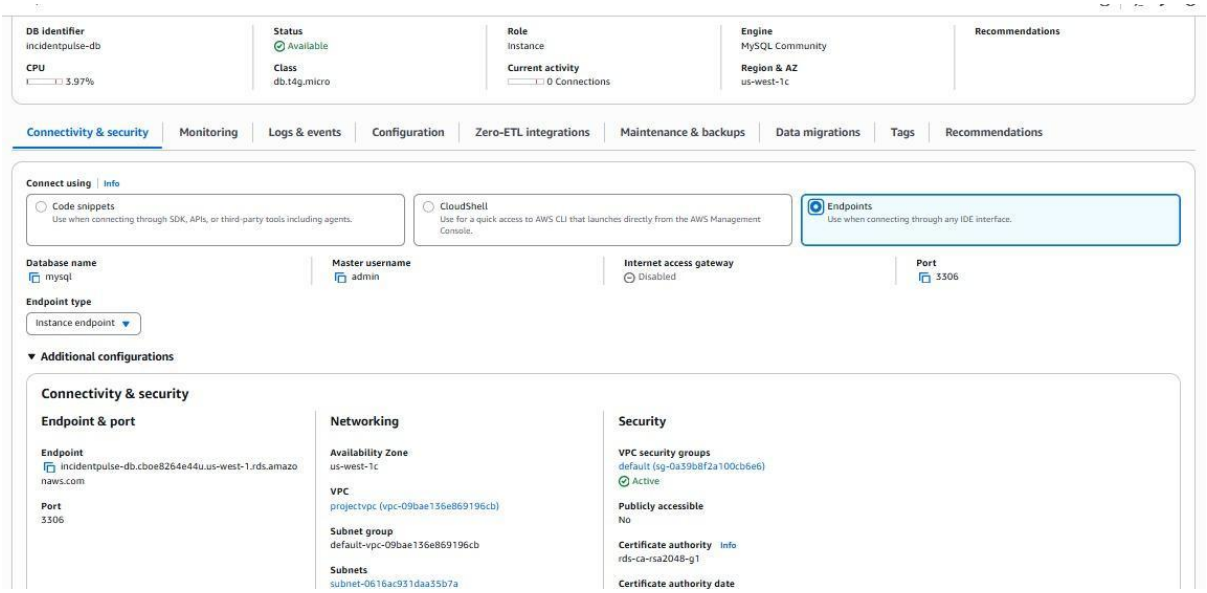



Distributions | CloudFront | Glo
runbooks12345 - S3 bucket | S
(7) WhatsApp
(anonymous)
d1war4vws0ipjc.cloudfront.net/runbook.pdf
(anonymous)
1 / 1
100%
Add freehand drawings, signatures, and highlights with the new Draw tool
Got it
Remind me later
Project: Secure Content Distribution
Environment: AWS (S3 + CloudFront)
1. Purpose
This runbook explains how to manage and securely distribute runbooks and patch notes using Amazon S3 and CloudFront with Signed URLs.
2. Architecture
- Private S3 bucket (runbooks)
- CloudFront distribution
- Origin Access Control (OAC)
- Signed URLs for restricted access
3. Daily Operations
1. Upload updated documents to S3 bucket
2. Verify CloudFront distribution is deployed
3. Generate signed URLs for users
4. Security
- S3 public access blocked
- CloudFront only access via OAC
- Signed URLs with expiration enabled
5. Troubleshooting
- Access Denied: Signed URL expired
- File Not Found: Check file name in S3


d1war4vws0ipjc.cloudfront.net/patch-notes.txt
Patch Notes â€“ Secure Content Distribution
Version: 1.0
Release Date: 30-Jan-2026
Changes:
- Created private S3 bucket for runbooks
- Configured CloudFront distribution
- Enabled Origin Access Control (OAC)
- Implemented signed URL access
Security Improvements:
- Blocked public S3 access
- Restricted downloads using signed URLs
Known Limitations:
- Signed URLs expire and must be regenerated
Next Steps:
- Add access logging
- Automate signed URL generation

5.16 RDS (MYSQL) CONFIGURATION

- RDS MySQL was deployed in a private subnet.
- Database incidentpulse_db was created.
- Tables were created for incidents, evidence, and runbooks.
- Only EC2 and Lambda services were allowed database access.




```
    -> );
Query OK, 0 rows affected (0.101 sec)

MySQL [incidentpulse_db]> CREATE TABLE runbooks (
    ->   id INT AUTO_INCREMENT PRIMARY KEY,
    ->   title VARCHAR(255),
    ->   s3_link VARCHAR(500),
    ->   updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
    -> );

Query OK, 0 rows affected (0.064 sec)

MySQL [incidentpulse_db]>
MySQL [incidentpulse_db]> INSERT INTO incidents (severity, title, owner)
    -> VALUES ('P1', 'Database down', 'shrusti');

Query OK, 1 row affected (0.042 sec)

MySQL [incidentpulse_db]>
MySQL [incidentpulse_db]> INSERT INTO evidence (incident_id, filename, uploader)
    -> VALUES (1, 'error-log.pdf', 'shrusti');

Query OK, 1 row affected (0.048 sec)

MySQL [incidentpulse_db]>
MySQL [incidentpulse_db]> INSERT INTO runbooks (title, s3_link)
    -> VALUES ('DB Restart Guide', 's3://runbooks9901/db-restart.pdf');
```

```
ySQL [incidentpulse_db]> CREATE TABLE incidents (
    ->   id INT AUTO_INCREMENT PRIMARY KEY,
    ->   severity VARCHAR(10),
    ->   title VARCHAR(255),
    ->   owner VARCHAR(100),
    ->   created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
    -> );
uery OK, 0 rows affected (0.098 sec)

ySQL [incidentpulse_db]> SELECT * FROM incidents;
mpty set (0.013 sec)

ySQL [incidentpulse_db]>
ySQL [incidentpulse_db]> CREATE TABLE evidence (
    ->   id INT AUTO_INCREMENT PRIMARY KEY,
    ->   incident_id INT,
    ->   filename VARCHAR(255),
    ->   uploader VARCHAR(100),
    ->   timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    ->   FOREIGN KEY (incident_id) REFERENCES incidents(id)
    -> );
uery OK, 0 rows affected (0.101 sec)

ySQL [incidentpulse_db]> CREATE TABLE runbooks (
    ->   id INT AUTO_INCREMENT PRIMARY KEY,
    ->   title VARCHAR(255),
```

i-014d16b32ef3b6236 (publicprojects)

PublicIPs: 54.219.145.15 PrivateIPs: 192.172.52.21

```
    ->   updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
    -> );

Query OK, 0 rows affected (0.064 sec)

MySQL [incidentpulse_db]>
MySQL [incidentpulse_db]> INSERT INTO incidents (severity, title, owner)
    -> VALUES ('P1', 'Database down', 'shrusti');

Query OK, 1 row affected (0.042 sec)

MySQL [incidentpulse_db]>
MySQL [incidentpulse_db]> INSERT INTO evidence (incident_id, filename, uploader)
    -> VALUES (1, 'error-log.pdf', 'shrusti');

Query OK, 1 row affected (0.048 sec)

MySQL [incidentpulse_db]>
MySQL [incidentpulse_db]> INSERT INTO runbooks (title, s3_link)
    -> VALUES ('DB Restart Guide', 's3://runbooks9901/db-restart.pdf');
Query OK, 1 row affected (0.012 sec)

MySQL [incidentpulse_db]> SELECT * FROM incidents;
+----+----------+---------------+---------+---------------------+
| id | severity | title         | owner   | created_at          |
+----+----------+---------------+---------+---------------------+
|  1 | P1       | Database down | shrusti | 2026-02-04 15:53:52 |
```

i-014d16b32ef3b6236 (publicprojects)

PublicIPs: 54.219.145.15 PrivateIPs: 192.172.52.21

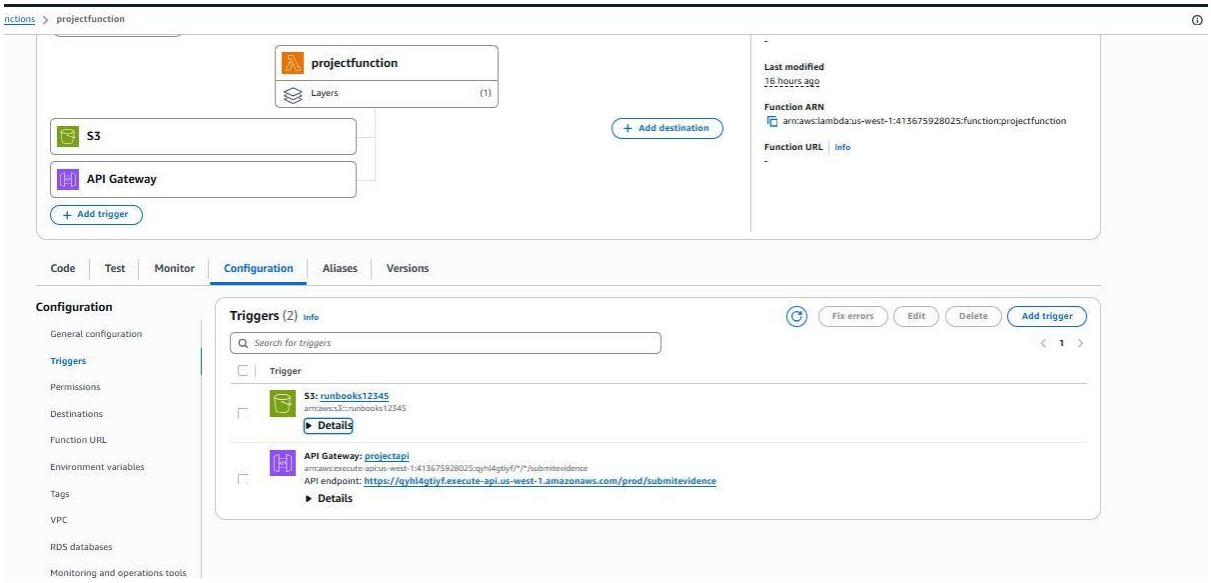
```
Query OK, 1 row affected (0.012 sec)

MySQL [incidentpulse_db]> SELECT * FROM incidents;
+----+----------+---------------+---------+---------------------+
| id | severity | title         | owner   | created_at          |
+----+----------+---------------+---------+---------------------+
|  1 | P1       | Database down | shrusti | 2026-02-04 15:53:52 |
+----+----------+---------------+---------+---------------------+
1 row in set (0.007 sec)

MySQL [incidentpulse_db]> SELECT * FROM evidence;
+----+-------------+--------------+----------+---------------------+
| id | incident_id | filename     | uploader | timestamp           |
+----+-------------+--------------+----------+---------------------+
|  1 |           1 | error-log.pdf | shrusti  | 2026-02-04 15:54:21 |
+----+-------------+--------------+----------+---------------------+
1 row in set (0.003 sec)

MySQL [incidentpulse_db]> SELECT * FROM runbooks;
+----+-----------------+-------------------------------+---------------------+
| id | title           | s3_link                       | updated_at          |
+----+-----------------+-------------------------------+---------------------+
|  1 | DB Restart Guide | s3://runbooks9901/db-restart.pdf | 2026-02-04 15:54:48 |
+----+-----------------+-------------------------------+---------------------+
1 row in set (0.004 sec)

MySQL [incidentpulse_db]>
```

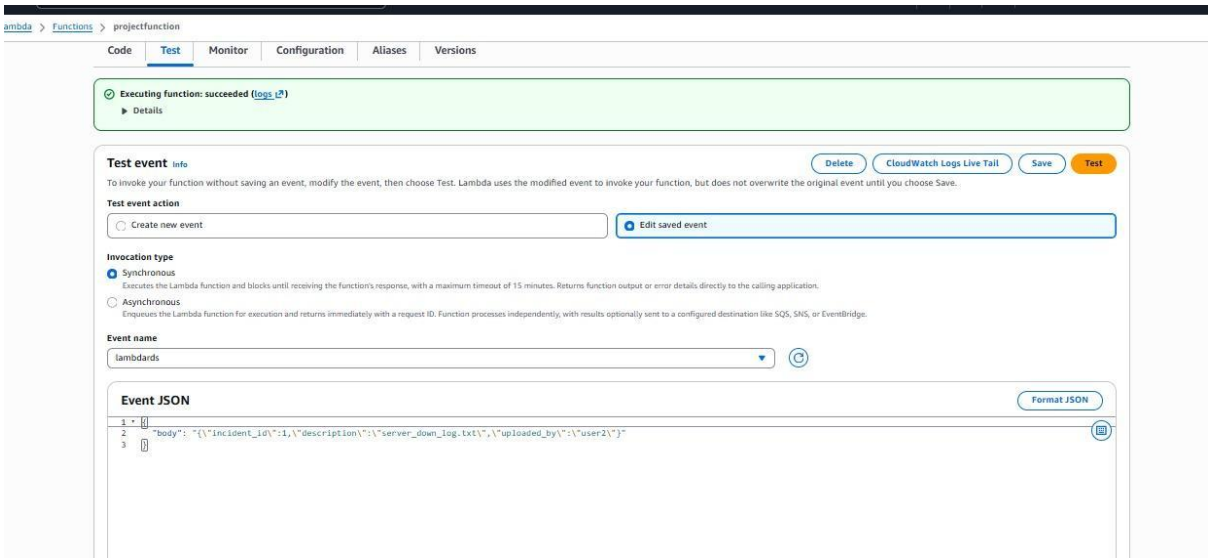
i-014d16b32ef3b6236 (publicprojects)

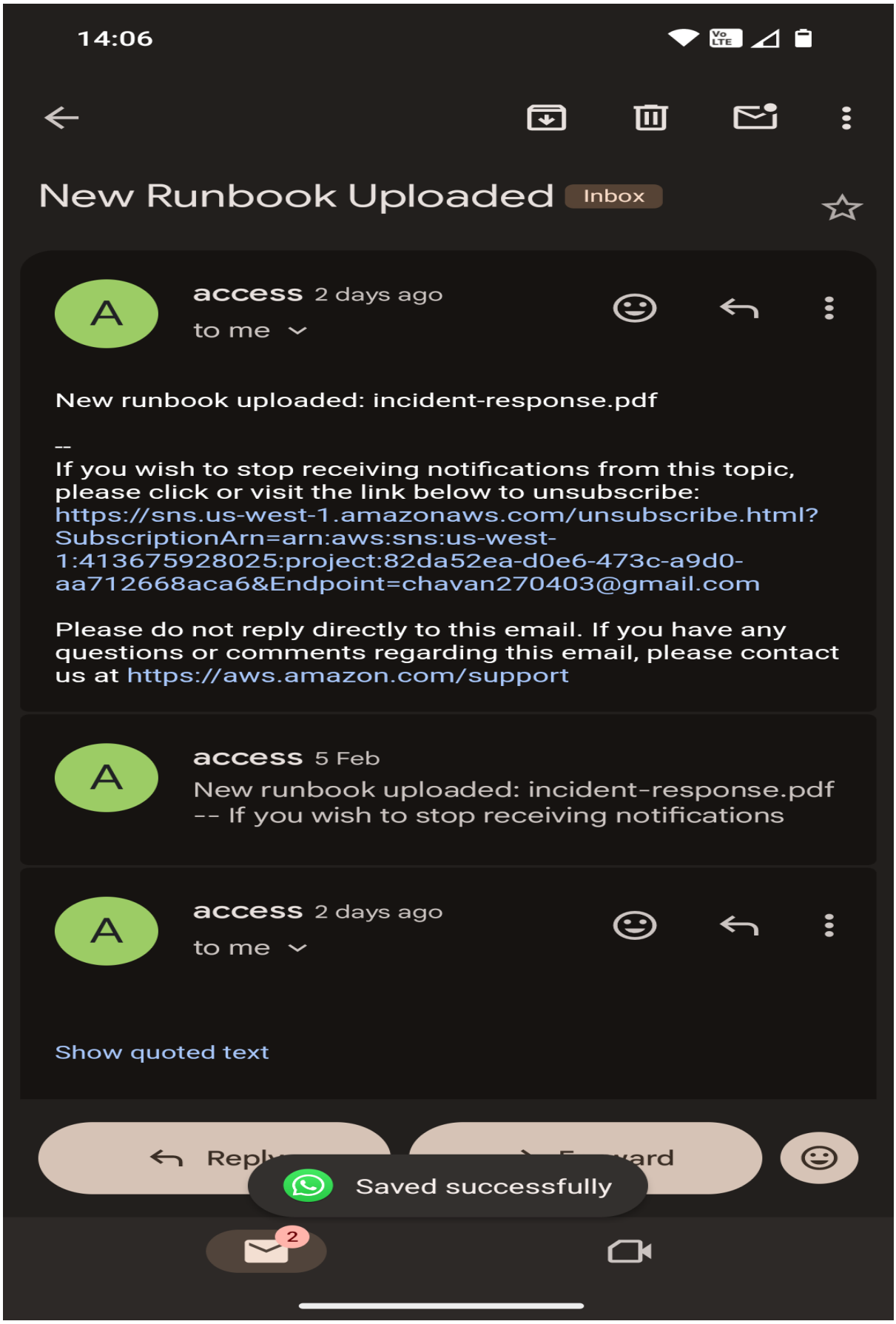
PublicIPs: 54.219.145.15 PrivateIPs: 192.172.52.21

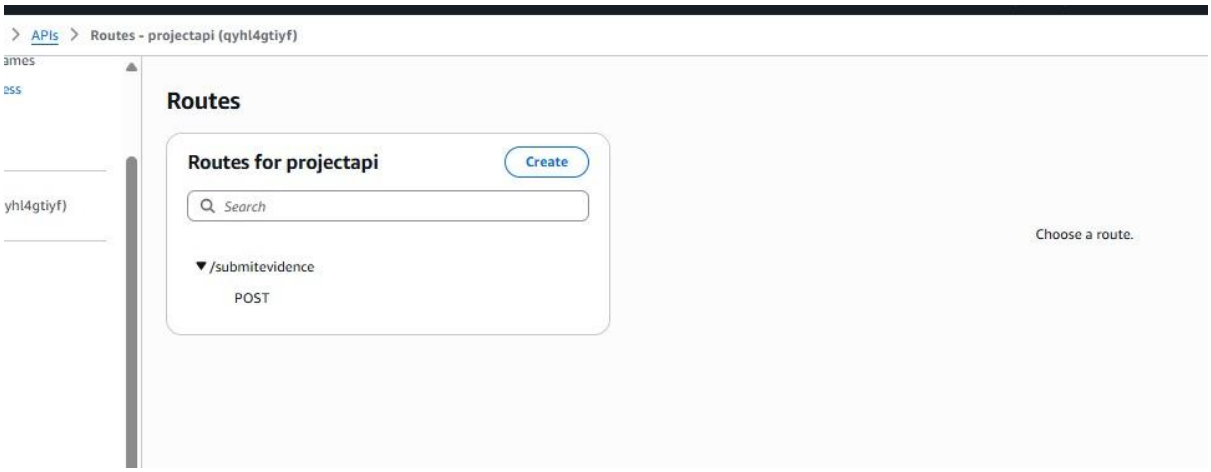
5.17 LAMBDA AND API GATEWAY INTEGRATION LOGIC

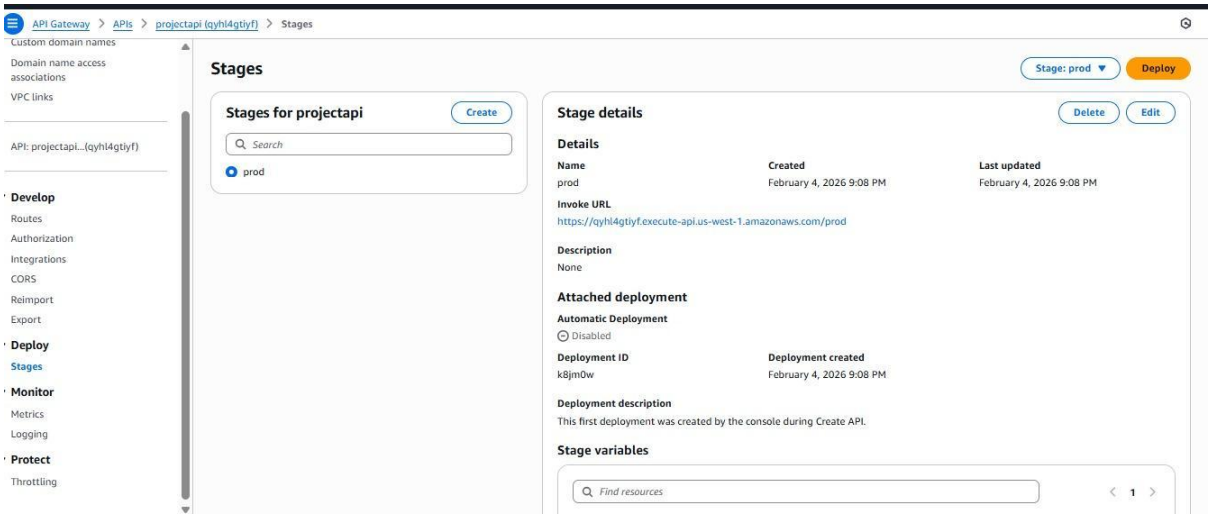
- S3 event triggers invoked Lambda functions on uploads.
- Lambda processed events and sent notifications.
- API Gateway exposed /submitEvidence endpoint.
- Metadata was stored securely in RDS.

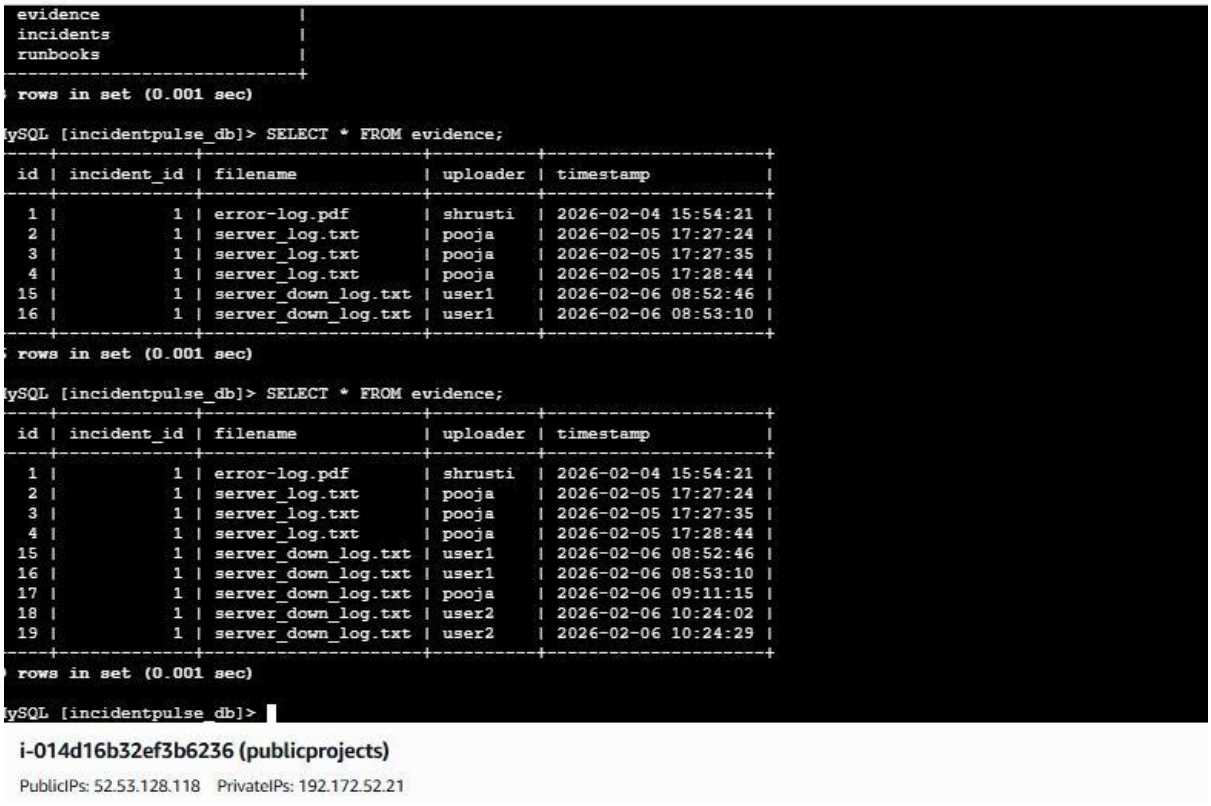



ambda > Functions > projectfunction
Code
Test
Monitor
Configuration
Aliases
Versions
Executing function: succeeded (logs)
Details
Test event Info
Delete
CloudWatch Logs Live Tail
Save
Test
To invoke your function without saving an event, modify the event, then choose Test. Lambda uses the modified event to invoke your function, but does not overwrite the original event until you choose Save.
Test event action
Create new event
Edit saved event
Invocation type
Synchronous
Executes the Lambda function and blocks until receiving the function's response, with a maximum timeout of 15 minutes. Returns function output or error details directly to the calling application.
Asynchronous
Enqueues the Lambda function for execution and returns immediately with a request ID. Function processes independently, with results optionally sent to a configured destination like SQS, SNS, or EventBridge.
Event name
lambdards
Event JSON
Format JSON
1 {
2 "body": "{\"incident_id\":1,\"description\":\"server_down_log.txt\",\"uploaded_by\":\"user2\"}"
3 }


14:06
New Runbook Uploaded
Inbox
access 2 days ago
to me
New runbook uploaded: incident-response.pdf
--
If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:
https://sns.us-west-1.amazonaws.com/unsubscribe.html?SubscriptionArn=arn:aws:sns:us-west-1:413675928025:project:82da52ea-d0e6-473c-a9d0-aa712668aca6&Endpoint=chavan270403@gmail.com
Please do not reply directly to this email. If you have any questions or comments regarding this email, please contact us at https://aws.amazon.com/support
access 5 Feb
New runbook uploaded: incident-response.pdf
-- If you wish to stop receiving notifications
access 2 days ago
to me
Show quoted text
Saved successfully


APIs > Routes - projectapi (qyhl4gtiyf)
Routes
Routes for projectapi
Create
Search
/submitevidence
POST
Choose a route.


API Gateway > APIs > projectapi (qyhl4gtiyf) > Stages
Custom domain names
Domain name access associations
VPC links
API: projectapi...(qyhl4gtiyf)
Develop
Routes
Authorization
Integrations
CORS
Reimport
Export
Deploy
Stages
Monitor
Metrics
Logging
Protect
Throttling
Stages
Stage: prod
Deploy
Stages for projectapi
Create
Search
prod
Stage details
Delete
Edit
Details
Name
prod
Created
February 4, 2026 9:08 PM
Last updated
February 4, 2026 9:08 PM
Invoke URL
https://qyhl4gtiyf.execute-api.us-west-1.amazonaws.com/prod
Description
None
Attached deployment
Automatic Deployment
Disabled
Deployment ID
k8jm0w
Deployment created
February 4, 2026 9:08 PM
Deployment description
This first deployment was created by the console during Create API.
Stage variables
Find resources
1


evidence
incidents
runbooks
rows in set (0.001 sec)
ySQL [incidentpulse_db]> SELECT * FROM evidence;
id | incident_id | filename | uploader | timestamp
1 | 1 | error-log.pdf | shrusti | 2026-02-04 15:54:21
2 | 1 | server_log.txt | pooja | 2026-02-05 17:27:24
3 | 1 | server_log.txt | pooja | 2026-02-05 17:27:35
4 | 1 | server_log.txt | pooja | 2026-02-05 17:28:44
15 | 1 | server_down_log.txt | user1 | 2026-02-06 08:52:46
16 | 1 | server_down_log.txt | user1 | 2026-02-06 08:53:10
rows in set (0.001 sec)
ySQL [incidentpulse_db]> SELECT * FROM evidence;
id | incident_id | filename | uploader | timestamp
1 | 1 | error-log.pdf | shrusti | 2026-02-04 15:54:21
2 | 1 | server_log.txt | pooja | 2026-02-05 17:27:24
3 | 1 | server_log.txt | pooja | 2026-02-05 17:27:35
4 | 1 | server_log.txt | pooja | 2026-02-05 17:28:44
15 | 1 | server_down_log.txt | user1 | 2026-02-06 08:52:46
16 | 1 | server_down_log.txt | user1 | 2026-02-06 08:53:10
17 | 1 | server_down_log.txt | pooja | 2026-02-06 09:11:15
18 | 1 | server_down_log.txt | user2 | 2026-02-06 10:24:02
19 | 1 | server_down_log.txt | user2 | 2026-02-06 10:24:29
rows in set (0.001 sec)
ySQL [incidentpulse_db]>
i-014d16b32ef3b6236 (publicprojects)
PublicIPs: 52.53.128.118 PrivateIPs: 192.172.52.21

CHAPTER 6

RESULTS / OUTPUT

6.1 FINAL OUTCOME ACHIEVED

The IncidentPulse – DevOps Incident & Patch Evidence Portal was successfully designed and implemented using Linux and AWS services.
The system enables secure collection, storage, validation, and archival of incident evidence while enforcing strict role-based access control.

The final solution provides:

- Secure evidence uploads through FTP with TLS
- Controlled access to runbooks using Linux permissions and ACLs
- Automated backups, archival, and notifications
- Scalable and highly available AWS infrastructure
- Secure data storage with encryption and disaster recovery

The project successfully simulates a real-world enterprise DevOps incident management workflow.

6.2 OBSERVATIONS BASED ON EXECUTION

- Linux role-based access control is critical for maintaining security and separation of duties.
- Automation using cron jobs significantly reduces manual effort and operational errors.
- AWS services such as ALB, Auto Scaling, and S3 improve system reliability and scalability.
- Placing backend services in private subnets enhances security.
- Encryption (LUKS) is essential for protecting sensitive incident data.
- Monitoring storage usage and auto-expanding volumes prevents downtime during incidents.

CHAPTER 7

CHALLENGES FACED

7.1 ISSUES ENCOUNTERED

1. Managing permissions across multiple Linux users and groups.
2. Restricting engineers strictly to FTP access without shell access.
3. Configuring ACLs correctly for fine-grained directory permissions.
4. Setting up secure FTP with TLS and chroot restrictions.
5. Designing correct VPC routing between public and private subnets.
6. Ensuring EC2 instances in private subnets had internet access.
7. Integrating multiple AWS services without using additional services.
8. Automating storage expansion using LVM without downtime.

7.2 HOW THE ISSUES WERE RESOLVED

1. Proper Linux group design and consistent permission testing were implemented.
2. FTP chroot configuration was enforced to prevent directory traversal.
3. ACLs were tested using different user logins to validate access control.
4. TLS certificates were correctly configured for encrypted FTP communication.
5. Route tables were carefully reviewed and validated for correct traffic flow.
6. NAT Gateway was used to provide secure outbound internet access for private subnet instances.
7. IAM policies were designed using the least privilege principle.
8. LVM monitoring scripts and thresholds were tested before deployment.

CHAPTER 8

CONCLUSION

The *IncidentPulse* project demonstrates the effective use of Linux system administration and AWS cloud services to build a secure, scalable, and automated incident management platform. By combining strong access control, encrypted storage, automation, and cloud scalability, the system closely reflects real-world DevOps and enterprise operational practices.

This project provided valuable hands-on experience in managing production-like environments and handling critical system incidents.

8.1 SUMMARY OF LEARNING

Through this project, the following key skills were developed:

- Linux user and group management
- File permissions, ACLs, and secure file handling
- FTP and NFS configuration
- LVM and encrypted storage management using LUKS
- Cron job automation
- AWS VPC design and subnet segregation
- EC2, S3, IAM, ALB, Auto Scaling, RDS, CloudFront, Lambda, and API Gateway usage
- Secure and scalable cloud architecture design
- Real-time DevOps incident handling workflows

8.2 OVERALL OUTCOME

The project successfully met all functional and security requirements defined in the problem statement.

It provided practical exposure to enterprise-level DevOps operations using Linux and AWS, enhancing technical proficiency and confidence in designing real-time cloud-based systems.

The *IncidentPulse* portal stands as a complete and well-structured solution for incident evidence management and operational audits.