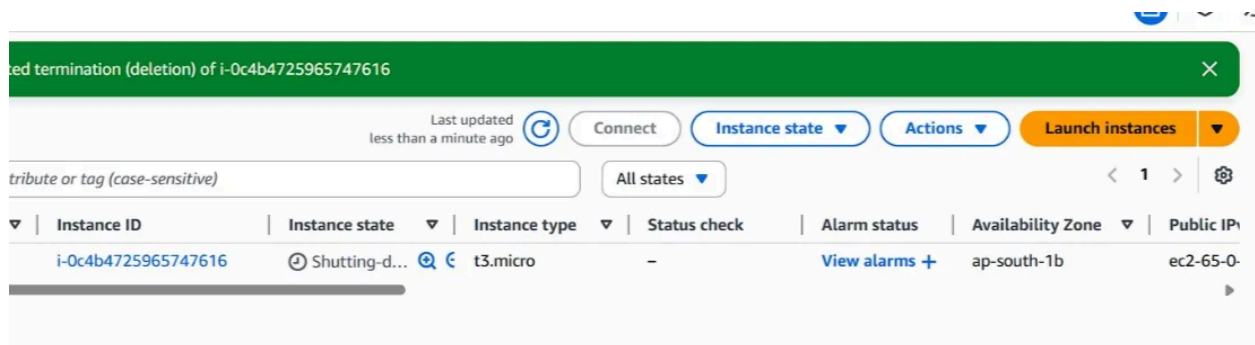


Cycle 08 Homework Documentation

1. AWS Resource Cleanup Practice

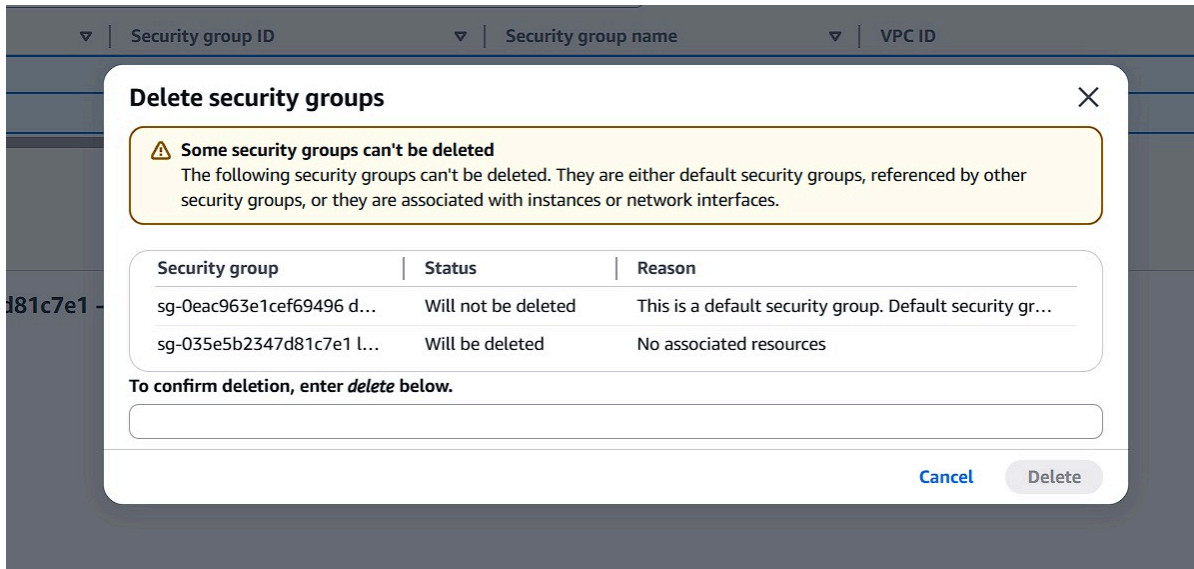
Efficiently manage AWS resources after usage to prevent unnecessary charges and maintain a clutter-free cloud environment.



Steps Performed

1. Delete Security Groups

Security groups associated with terminated instances were removed. Before deletion, all rules were reviewed to ensure no dependencies existed. This step helps avoid potential access loopholes and resource clutter.



2. Release Elastic IPs

Any allocated Elastic IPs not in use were released. Since AWS charges for Elastic IPs when they're not associated with a running instance, this step helps eliminate idle cost accumulation.

3. Remove Unused Key Pairs

Key pairs that were not linked to active instances were deleted. This improves account security and reduces clutter in the EC2 Key Pairs section.

4. Delete Detached EBS Volumes


Detached volumes were identified and deleted manually. These volumes may remain after instance termination if not configured for auto-deletion, leading to storage costs.


5. Verify Clean Dashboard

A final inspection of the EC2 dashboard, EBS volumes, Elastic IPs, snapshots, and AMIs confirmed that no chargeable resources were running.

Resources

EC2 Global View





You are using the following Amazon EC2 resources in the Asia Pacific (Mumbai) Region:

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

Purpose

This practice ensures efficient cloud resource management, prevents billing surprises, and reinforces real-world AWS cleanup techniques.

2. Web Server Deployment

Goal

Deploy web servers on both Windows and Linux EC2 instances and verify their functionality by accessing them through public IPs.

A. Windows EC2 Instance Deployment with IIS

- **AMI Used:** Microsoft Windows Server 2019 Base
- **Instance Type:** t2.micro (Free Tier eligible)
- **Access Method:** Remote Desktop Protocol (RDP, port 3389)
- **Steps Followed:**
 1. Launched a new EC2 instance using the Windows AMI.
 2. Allowed RDP access by modifying security group rules.
 3. Downloaded the RDP file and decrypted the administrator password.
 4. Logged in and installed IIS using Windows Server Manager.
 5. Verified IIS by entering the public IP in a browser — saw the default IIS landing page.

B. Linux EC2 Instance Deployment with Apache

- **AMI Used:** Amazon Linux 2
- **Instance Type:** t2.micro
- **Access Method:** SSH (port 22)
- **Steps Followed:**
 1. Created and launched a new Linux EC2 instance.
 2. Opened port 22 (for SSH) and port 80 (for HTTP) in the security group.
 3. Connected via SSH using the private key and terminal.
 4. Installed Apache HTTP Server:

```
sudo yum update -y
sudo yum install httpd -y
sudo systemctl start httpd
sudo systemctl enable httpd
```

5. Verified Apache by visiting the instance's public IP in a browser.

Validation

Both servers responded successfully via their respective IPs, proving proper deployment and network configuration.

3. PAN Card & FI Money/Jupiter Setup

Steps Taken

- Ensured that a valid **PAN card** is linked to an active **FI Money or Jupiter** account.
- Verified the ability to perform **international transactions**, as AWS billing systems often require international payment authorization.
- This setup is essential to ensure billing functionality for AWS Free Tier and paid services, especially when local payment cards face issues with foreign

transactions.

Notes

Encountered temporary issues during the practical AWS setup due to a blocked or unsupported payment method. Resolved by enabling international usage in the banking app settings.

4. Research Findings

A. AWS Free Tier Limits

Always Free Services

- AWS Lambda: 1M free requests/month
- DynamoDB: 25GB storage with read/write capacity
- S3: 1000 PUT, 1000 GET, 1GB/month

12-Month Free Services

- EC2: 750 hours/month (t2.micro or t3.micro)
- RDS: 750 hours/month (db.t2.micro or db.t3.micro)
- S3: 5GB Standard Storage

Observations

- The 750-hour/month limit means you can run **one EC2 instance full-time**.
 - Running multiple instances can exceed the free tier, leading to billing.
 - Tracking usage through AWS Billing Dashboard and Free Tier alerts is highly recommended.
-

B. Deep Dive into Protocols & Ports

Port and Protocol Understanding

Protocol	Port	Purpose
HTTP	80	Web traffic (Apache, IIS)
HTTPS	443	Secure web traffic (SSL)
SSH	22	Linux server remote access
RDP	3389	Windows remote desktop connection

Security Group Configuration

- Created custom security groups for each instance.
 - Allowed only necessary inbound ports to minimize security risks.
 - Temporarily restricted RDP/SSH for testing security behavior.
 - Observed that without open ports, instances are unreachable externally, reinforcing the importance of proper security group setup.
-

Challenges Faced

- Initial payment method setup failed due to card restrictions.
 - Minor troubleshooting required during Apache installation on Linux.
-