Homework Documentation

1. AWS Free Tier Account Setup

Objective:

To set up a secure and cost-effective personal AWS environment using the AWS Free Tier.

Steps Followed:

- Created a new AWS account using a **non-primary email ID** and **separate credit/debit card**.
- Enabled **Multi-Factor Authentication (MFA)** on the root account to enhance login security.
- Created an **IAM user with AdministratorAccess** named `YourName_Dev`.
- IAM user credentials used for all activities instead of the root account.

Key Learnings:

- Importance of using IAM users for day-to-day operations.
- How MFA protects the root account from unauthorized access.
- Billing alerts and cost monitoring through AWS Billing Dashboard.

---

2. EC2 Hands-On Practice

Goal:

To gain hands-on experience with launching and managing EC2 instances on the Free Tier.

a) Linux EC2 Instance Setup via AWS Console

1. Logged into AWS Console with IAM user credentials.
2. Navigated to EC2 → Launched Instance.
3. Selected Amazon Linux 2 AMI, t2.micro instance (Free Tier eligible).
4. Created a new key pair (.pem) and downloaded it.
5. Configured security group to allow SSH (Port 22).
6. Launched instance in **Mumbai/Hyderabad** region.
7. Connected via SSH:

```
ssh -i "key.pem" ec2-user@<public-IP>
```

1. Verified connectivity with basic Linux commands.

b) Windows EC2 Instance Setup via AWS Console

1. Launched another instance with **Microsoft Windows Server 2022 Base** AMI.
2. Selected **t2.micro**, same region.
3. Used existing or new key pair.
4. Configured security group to allow **RDP (Port 3389)**.
5. Launched the instance.
6. Retrieved admin password using the key pair.
7. Connected using **Remote Desktop Client (RDP)**.

---

3. Security Best Practices & Resource Cleanup

Security Measures:

- Kept the `.pem` file secure.
- Avoided sharing of RDP credentials.

- Restricted access via specific ports only (SSH: 22, RDP: 3389).
- Used a single region to avoid cross-region data transfer costs.

Cleanup:

- Terminated both Linux and Windows EC2 instances.
- Deleted:
  - Key pairs
  - Custom security groups
  - Unused volumes and snapshots

Billing Monitoring:

- Checked Billing Dashboard.
- Verified no active resources or charges post-lab.

Conclusion

Cycle 08 offered a practical foundation in AWS, covering secure account setup, launching EC2 instances, connectivity, and best practices. The experience reinforced real-world cloud usage principles and resource management within Free Tier limits.

Cycle 08 Homework Documentation

---

1. AWS Free Tier Account Setup

Objective:

To set up a secure and cost-effective personal AWS environment using the AWS Free Tier by creating an isolated user account, ensuring strong authentication, and preparing for billing awareness and resource control.

Steps Followed:

- Created a new AWS account using a **non-primary email ID** and a **separate credit/debit card** for better isolation and cost tracking.
- Logged in to the AWS Management Console and enabled **Multi-Factor Authentication (MFA)** for the root account using an authenticator app. This ensures two-layer protection against unauthorized access.
- Created a new **IAM user** named `YourName_Dev` with full **AdministratorAccess** policy attached.
- Generated access keys for the IAM user and used it to log in for all future operations.

Key Learnings:

- Learned how to segregate user roles and why it's critical to avoid using the root user for daily tasks.
- Understood the importance of **MFA** and how it can prevent potential unauthorized logins.
- Gained awareness of how to check the **Billing Dashboard**, set up **alerts**, and ensure usage remains within Free Tier limits.

---

2. EC2 Hands-On Practice

Goal:

To gain experience deploying, accessing, and managing EC2 virtual machines (instances) on AWS, with a focus on both Linux and Windows environments using Free Tier eligible options.

a) Linux EC2 Instance Setup via AWS Console

1. Logged into the AWS Console using the IAM user.
2. Navigated to **EC2 Dashboard > Launch Instance**.
3. Selected **Amazon Linux 2 AMI (HVM), SSD Volume Type**.

4. Chose **t2.micro** instance type (eligible for Free Tier).
5. Configured instance details including network (VPC and subnet) settings.
6. Created a **new key pair (RSA)** and downloaded the `.pem` file securely.
7. Added a **security group rule** to allow SSH access on **Port 22** from the required IP range.
8. Launched the instance in **Mumbai/Hyderabad region**.
9. After the instance entered "running" state, noted the **public IP** address.
10. Connected using SSH from terminal:

```
ssh -i "your-key.pem" ec2-user@<public-IP>
```

1. Ran basic Linux commands (e.g., `uname -a`, `df -h`, `top`) to verify system operation.

b) Windows EC2 Instance Setup via AWS Console

1. Repeated the above steps with adjustments:
2. Selected **Microsoft Windows Server 2022 Base** as the AMI.
3. Again selected **t2.micro** instance type.
4. Used the **same or new key pair**.
5. Added **security group rule** to allow **RDP (Port 3389)** access.
6. Launched instance in the same region for cost consistency.
7. Once running, selected the instance and used the **Get Password** option to decrypt admin password using the `.pem` key.
8. Opened the **Remote Desktop Client**, entered public IP and decrypted password.
9. Successfully connected to Windows instance, accessed Server Manager, and performed a basic configuration check.

Key Learnings:

- Understood AMI types, instance families, and configuration parameters.
- Gained confidence in launching and connecting to both Linux and Windows servers.
- Practiced working with key pairs and managing security groups.

---

3. Security Best Practices & Resource Cleanup

Security Measures:

- `.pem` key was stored securely and not shared. Ensured only SSH/RDP access was granted.
- All inbound rules in security groups were reviewed and set to allow access only from required IPs.
- Ensured all access credentials were deleted after lab completion.
- Selected and used only **one region** (Mumbai or Hyderabad) to avoid multi-region charges.

Cleanup Process:

- **Terminated both Linux and Windows EC2 instances** after completing hands-on tasks.
- Deleted:
  - Associated key pairs from the AWS Console.
  - Custom-created security groups.
  - Any EBS volumes or snapshots not automatically deleted.

Billing Monitoring:

- Navigated to the **Billing Dashboard** and confirmed no residual costs.
- Enabled **Free Tier usage alerts** and reviewed the **Cost Explorer**.

Conclusion

- Understanding and applying cloud security measures
- Using IAM roles responsibly
- Practicing with real EC2 Linux and Windows instances
- Monitoring and controlling costs effectively

This lab built real-world readiness for cloud projects, improved practical confidence with AWS Console navigation, and introduced essential cloud hygiene practices necessary for enterprise environments.