V ANISHA 212224040023

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EXP NO** | 01 | **CLOUD ACCOUNT SETUP**  **AND SERVICES OVERVIEW** | **Year/Sem** | 1 |
| **DATE** | 08/03/2025 | **Branch** | CSE |

**Aim:**

To create an AWS cloud account and explore its various services and to explore various AWS storage services, including Amazon S3, EBS, and EFS, along with best practices for securing, optimizing, and automating storage.

**Procedure:**

* Create an AWS Account
  + Go to AWS Official Website.
  + Click on "Create an AWS Account" and enter the required details (email, password, account name).
  + Choose "Personal" or "Business" account type and provide billing information.
  + Complete identity verification using a phone number.
  + Select a support plan (Free Tier recommended for beginners).
  + Log in to the AWS Management Console.
* Exploring AWS Services
* Navigate through the AWS Management Console to explore services.
* Access different categories such as Compute, Storage, Database, Networking, and Security.
* Experiment with services like EC2 (Elastic Compute Cloud), S3 (Simple Storage Service), and RDS (Relational Database Service) using Free Tier options.

**AWS:**

Amazon Web Services (AWS) is a comprehensive cloud computing platform that offers on-demand computing power, storage, and a wide range of services tailored for businesses and developers. AWS provides a flexible, scalable, and cost-effective solution for various computing needs.

**Services in AWS:**

* **Compute & Networking:**

Services like EC2, Lambda, and VPC for running applications and managing network resources.

* **Storage & Content Delivery:**

Solutions such as S3, EBS, EFS, and CloudFront for storing and delivering content.

* **Database:**

Managed database services like RDS, DynamoDB, and Aurora for structured and unstructured data.

* **Deployment & Management:**

Tools like CloudFormation, Elastic Beanstalk, and AWS OpsWorks for deploying and managing applications.

* **Analytics & AI:**

Services like Amazon Redshift, Athena, and SageMaker for data analysis and machine learning.

* **Mobile & Application Services:**

Services like AWS Amplify and AppSync for building mobile and web applications.

**Applications of AWS:**

* **Web Hosting:**

Hosting scalable websites and applications with high availability.

* **Big Data Analytics:**

Processing large datasets for insights using services like EMR and Redshift.

* **AI & Machine Learning:**

Training and deploying machine learning models with tools like SageMaker.

* **IoT (Internet of Things):**

Connecting and managing IoT devices with AWS IoT Core.

* **Gaming:**

Powering multiplayer and cloud gaming applications with low latency.

* **Backup & Disaster Recovery:**

Secure storage and failover solutions to ensure business continuity.

**Intermediate AWS Storage Guide**

**1. Amazon S3 Storage Class**

Amazon S3 offers multiple storage classes designed for different use cases, balancing cost, durability, and retrieval speed. Choosing the right class helps optimize cost and efficiency.

**S3 Storage Classes Overview**

* S3 Standard – Ideal for frequently accessed data, offering low latency and high throughput. Suitable for use cases like content distribution and big data analytics.
* S3 Intelligent-Tiering – Automatically moves data between two access tiers when access patterns change, optimizing costs without performance impact.
* S3 Standard-IA (Infrequent Access) – Lower cost for data that is accessed less frequently but requires rapid access when needed. Ideal for backups and disaster recovery.
* S3 One Zone-IA – Cost-effective storage for infrequently accessed data that does not require multiple availability zone resilience.
* S3 Glacier – Designed for long-term archival storage, offering retrieval options from minutes to hours. Suitable for compliance and regulatory data.
* S3 Glacier Deep Archive – The most cost-effective storage class for long-term data archiving, with retrieval times ranging from 12 to 48 hours.

**Lifecycle Policies for Cost Optimization**

To minimize costs, configure S3 Lifecycle Rules to automatically transition objects between storage classes. For example:

* Move logs from S3 Standard to Intelligent-Tiering after 30 days.
* Transition backups from S3 Standard-IA to Glacier after 90 days.
* Delete objects that are no longer needed after a specified retention period.

**Use Cases for Different S3 Storage Classes**

* Web applications: Use S3 Standard for fast performance and low latency.
* Backup and recovery: Utilize S3 Standard-IA and Glacier for cost-effective storage solutions.
* Data analytics: I ntelligent-Tiering reduces manual data movement and optimizes costs.
* Compliance storage: Glacier Deep Archive is ideal for long-term retention of compliance data.

**Best Practices for Using S3**

* Data Organization: Use a logical folder structure and naming conventions to organize data effectively.
* Versioning: Enable versioning to protect against accidental deletions and overwrites.
* Data Lifecycle Management: Implement lifecycle policies to manage data retention and optimize costs.
* Monitoring and Logging: Use S3 Access Logs and AWS CloudTrail to monitor access and changes to your S3 buckets.

**2. How to Secure Your Amazon S3 Buckets**

Many security breaches happen due to misconfigured S3 buckets. Implement these best practices to ensure data security:

1. Block Public Access

* AWS allows you to configure Block Public Access Settings at the bucket or account level to prevent accidental public exposure.
* This feature is crucial for protecting sensitive data.

2. IAM Policies for Access Control Use AWS Identity and Access Management (IAM) to define precise access permissions.

* Principle of Least Privilege: Only grant necessary permissions to users and roles. Regularly review and adjust permissions as needed.
* Use Bucket Policies: Define who can access your bucket and under what conditions. For example, restrict access to specific IP addresses or VPCs.
* Enable MFA Delete: Prevent accidental deletion by requiring multi-factor authentication for delete operations.

3. Encryption Best Practices

* Server-Side Encryption (SSE): Encrypts objects before storing them in S3.
* SSE-S3 – Managed by AWS, automatically encrypts data at rest.
* SSE-KMS – Uses AWS Key Management Service for encryption, allowing for more control over keys.
* SSE-C – User-provided encryption keys, giving you full control over the encryption process.Client-Side Encryption: Encrypt data before uploading to S3.
* Client-Side Encryption: Encrypt data before uploading to S3, ensuring that data is protected before it reaches AWS.

4. Logging & Monitoring for Security

* Enable AWS CloudTrail for tracking API requests and changes to your S3 buckets.
* Use Amazon Macie to detect sensitive data exposure, such as personally identifiable information (PII).
* Activate S3 Access Logs to monitor unauthorized access attempts and analyze usage patterns.

**3. Setting Up Automated Backups with AWS Backup & S3 Versioning**

AWS provides automated solutions to ensure data backups are maintained securely.

**1. Enabling S3 Versioning**

S3 Versioning allows you to retain multiple versions of an object, preventing accidental overwrites or deletions.

Steps to Enable Versioning:

1. Open the S3 Console.

2. Select a Bucket.

3. Click Properties > Versioning > Enable.

4. Save changes.

**2. Using AWS Backup for EBS, RDS, and S3 Backups**

AWS Backup centralizes backup management across AWS services.

* Create Backup Plans to automate snapshot schedules.
* Define Retention Policies to manage backup lifecycles.
* Cross-Region Replication enables disaster recovery strategies.

**3. Lifecycle Policies for Backup Retention**

* Keep daily backups for 7 days.
* Transition backups older than 30 days to Glacier.
* Automatically delete backups after 1 year to save costs.

**4. Disaster Recovery Strategies with AWS Storage**

* Multi-region backups: Ensure redundancy in case of failure.
* Cross-Account Backup Strategies: Protect against insider threats.
* AWS Backup Audit Manager: Automate compliance audits.

**4. How to Mount Amazon EFS for Scalable Shared Storage**

Amazon Elastic File System (EFS) provides scalable, shared file storage for AWS services, particularly useful for applications requiring concurrent access from multiple EC2 instances.

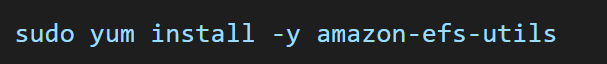
**Steps to Set Up and Mount EFS:**

1. Create an EFS File System

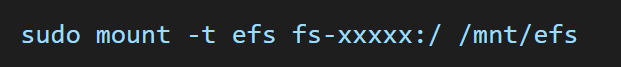
* Navigate to the EFS Console.
* Click Create File System.
* Choose a VPC and Availability Zones.
* Enable automatic backup (optional).
* Click Create.

2. Mount EFS to an EC2 Instance

* 1. Connect to an EC2 instance via SSH.
* 2. Install NFS utilities:



* 3. Mount the EFS file system:



* 4. Verify the mount:



3. EFS Performance & Cost Optimization

* Use EFS Standard for high-performance applications.
* Move infrequently accessed data to EFS Infrequent Access (IA) for cost savings.
* Use AWS Auto Scaling to manage cost based on workload.

4. EFS Security Best Practices

* IAM Policies: Use IAM policies to control access to EFS file systems.
* Encryption: Enable encryption at rest and in transit to protect sensitive data.
* Monitoring: Use Amazon CloudWatch to monitor EFS performance and set alarms for unusual activity.

5. Use Cases for EFS

* Content Management Systems: Store and serve media files

**5. Comparing AWS Storage Solutions: S3 vs. EBS vs. EFS**

**Amazon S3**

* Object storage for static files and backups.
* Best for web hosting, data lakes, and archives.

**Amazon EBS**

* Block storage for persistent disk volumes.
* Best for databases, high-speed applications.

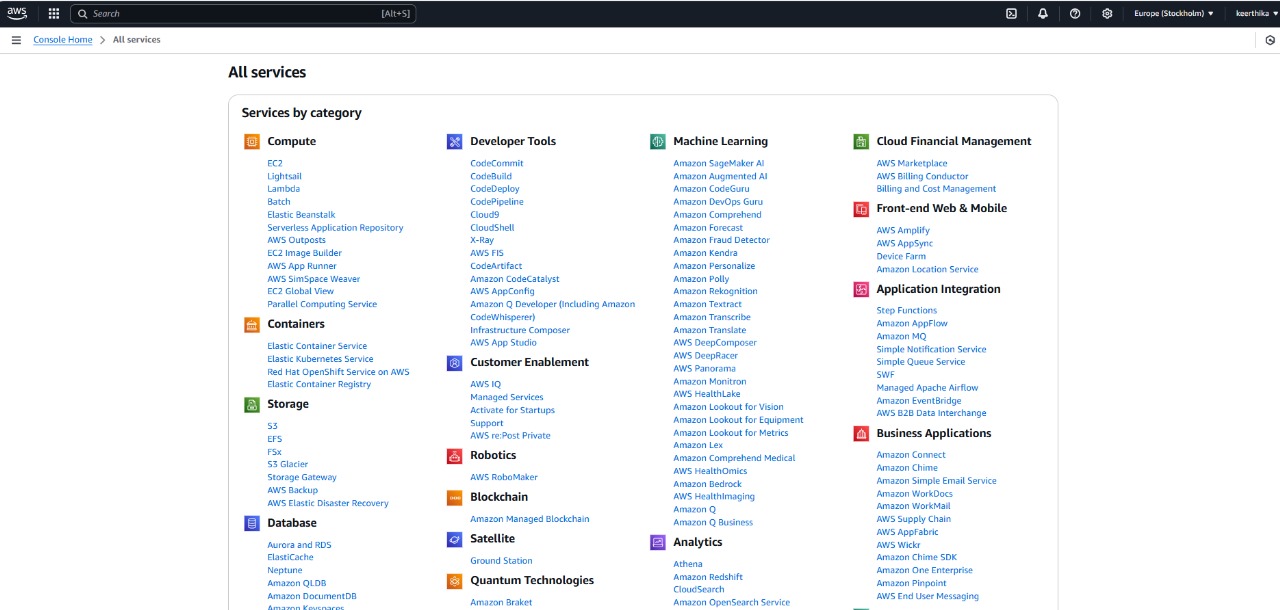
**Amazon EFS**

* File storage for multiple EC2 instances.
* Best for big data processing, web applications.

**Choosing the Right Storage Service**

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **S3** | **EBS** | **EFS** |
| Access | Global | Single Instance | Multi-Instance |
| Use Case | Backup, Archive | Databases | Shared Storage |
| Performance | Moderate | High | Scalable |
| Cost | Low | Medium | High |

**Output:**



**Conclusion**

AWS provides a variety of storage solutions for different use cases. Understanding how to secure, optimize, and automate AWS storage can lead to better cost efficiency, security, and performance. By implementing best practices like IAM policies, encryption, versioning, and lifecycle policies, businesses can maximize AWS storage capabilities while reducing risks and expenses.

**Result:**

Successfully created an AWS account and explored various AWS services.