4.2. Student Handout

Amazon EC2 (Elastic Compute Cloud) Student Handout

Overview

Amazon EC2 (Elastic Compute Cloud) is a web service that provides resizable compute capacity in the cloud. It allows you to rent virtual servers, known as EC2 instances, to run your applications.

Key Concepts and Benefits of EC2

1. **Elasticity**:

- Scale resources up or down based on demand.
- Example: Increase instances during a sale event.
- Example: Reduce instances during off-peak hours.
- Example: Automatically adjust resources for a viral campaign.

2. **Pay-as-you-go**:

- Pay only for the resources you use.
- Example: Pay for 5 hours if an instance runs for 5 hours.
- Example: No charge when instances are stopped.
- Example: Cost savings by terminating unused instances.

3. **Global Reach**:

- Launch instances in various regions worldwide.
- Example: Deploy in the US for American users.
- Example: Use the Tokyo region for Japanese customers.
- Example: Choose the Frankfurt region for European users.

4. **Security**:

- Control access with Security Groups and Key Pairs.

- Example: Allow SSH access only from your IP.
- Example: Use Key Pairs for secure logins.
- Example: Restrict HTTP access to specific IP ranges.

EC2 Instance Types

1. **General Purpose Instances**:

- Balanced resources for diverse applications.
- Example: t3.micro for small websites.
- Example: m5.large for application servers.
- Example: a1.medium for development environments.

2. **Compute-Optimized Instances**:

- High computational power for intensive tasks.
- Example: c5.large for gaming servers.
- Example: c6g.medium for scientific modeling.
- Example: c5n.xlarge for machine learning.

3. **Memory-Optimized Instances**:

- High memory for large databases or caches.
- Example: r5.large for in-memory databases.
- Example: x1e.xlarge for SAP HANA workloads.
- Example: z1d.large for high-performance computing.

4. **Storage-Optimized Instances**:

- High-speed storage for data-intensive applications.
- Example: i3.large for big data processing.
- Example: d2.xlarge for data warehousing.
- Example: h1.2xlarge for log processing.

EC2 Pricing Models

1. **On-Demand Instances**:

- Pay by the hour or second with no commitments.
- Example: Short-term testing environments.
- Example: Unpredictable traffic spikes.
- Example: Development and testing workloads.

2. **Reserved Instances**:

- Commit for 1 or 3 years for discounts.
- Example: Long-term web hosting.
- Example: Steady-state applications.
- Example: Predictable workloads.

3. **Spot Instances**:

- Bid for unused capacity at lower prices.
- Example: Batch processing jobs.
- Example: Flexible data analysis tasks.
- Example: Fault-tolerant applications.

Launching Your First EC2 Instance

1. **Choose an Amazon Machine Image (AMI)**:

- Select a pre-configured template.
- Example: Amazon Linux 2 for web servers.
- Example: Windows Server for enterprise applications.
- Example: Ubuntu for open-source projects.

2. **Choose an Instance Type**:

- Select based on application needs.
- Example: t3.micro for low-traffic websites.
- Example: c5.large for compute-heavy tasks.
- Example: r5.large for memory-intensive applications.

3. **Configure Instance Details**:

- Set number of instances and network settings.
- Example: Enable auto-scaling for traffic spikes.

- Example: Configure VPC for network isolation.
- Example: Set IAM roles for instance permissions.

4. **Add Storage**:

- Attach storage volumes to your instance.
- Example: Add EBS for persistent storage.
- Example: Use Instance Store for temporary data.
- Example: Configure RAID for performance.

5. **Configure Security Groups**:

- Define firewall rules for your instance.
- Example: Allow SSH on port 22.
- Example: Permit HTTP on port 80.
- Example: Restrict access to specific IPs.

6. **Review and Launch**:

- Finalize settings and launch the instance.
- Example: Create a Key Pair for secure access.
- Example: Review instance configuration.
- Example: Launch and monitor instance status.

Managing EC2 Instance Storage

1. **EBS (Elastic Block Store)**:

- Persistent storage for instances.
- Example: Store database files.
- Example: Keep application logs.
- Example: Backup critical data.

2. **Instance Store**:

- Temporary storage for ephemeral data.
- Example: Cache temporary files.
- Example: Use for scratch data.
- Example: Store session data.

3. **Adding and Resizing EBS Volumes**:

- Increase storage as needed.
- Example: Add volumes for additional capacity.
- Example: Resize volumes for growing applications.
- Example: Optimize storage for performance.

4. **Backing Up Data with EBS Snapshots**:

- Create backups of EBS volumes.
- Example: Snapshot before major updates.
- Example: Use snapshots for disaster recovery.
- Example: Share snapshots across regions.

Connecting to an EC2 Instance

1. **Linux Instances (SSH)**:

- Connect using Secure Shell.
- Example: Use SSH for remote access.
- Example: Authenticate with a private key.
- Example: Manage instances via command line.

2. **Windows Instances (RDP)**:

- Connect using Remote Desktop Protocol.
- Example: Download RDP file for access.
- Example: Use password for authentication.
- Example: Manage instances via GUI.

Conclusion

Amazon EC2 provides flexible, scalable, and cost-effective cloud computing resources. By understanding instance types, pricing models, and storage options, you can effectively deploy and manage applications in the cloud.

Thank you for participating in this session! If you have any questions, feel free to ask.