Case Study on Amazon EC2

Amazon EC2 (Elastic Compute Cloud) is a core part of Amazon Web Services (AWS), providing resizable compute capacity in the cloud.

It is designed to make web-scale computing easier for developers. Here's an overview and case study on Amazon EC2:

Overview of Amazon EC2

Amazon EC2 provides virtual servers, also known as instances, in the cloud. Users can launch, configure, and scale EC2 instances

based on their computing needs. EC2 offers several benefits:

- Scalability: EC2 instances can be scaled up or down based on demand. Users can launch new instances as needed, helping to handle varying workloads.

- Cost Efficiency: EC2 follows a pay-as-you-go pricing model, allowing users to only pay for the compute power they use, which helps reduce costs.

- Flexibility: EC2 supports multiple operating systems, including Linux and Windows. Users can also choose from a variety of instance types optimized for different use cases (compute, memory, storage).

- Security: EC2 instances are secured by AWS's Virtual Private Cloud (VPC), and users can configure firewalls, access controls, and encryption.

- Integrated Services: EC2 integrates with other AWS services like S3 (storage), RDS (database), and CloudWatch (monitoring), providing a comprehensive cloud solution.

Key Features of Amazon EC2:

1. Instance Types: EC2 offers different instance types suited for various workloads:

   - General Purpose (e.g., t2, m5 instances)

- Compute Optimized (e.g., c5 instances)

- Memory Optimized (e.g., r5 instances)

- Storage Optimized (e.g., i3 instances)

- Accelerated Computing (e.g., p3 instances for machine learning)

2. Elastic Load Balancing (ELB): Automatically distributes incoming application traffic across multiple instances to ensure high availability.

3. Auto Scaling: Automatically adjusts the number of EC2 instances in response to traffic, ensuring consistent performance while optimizing costs.

4. Amazon Machine Images (AMIs): Pre-configured templates to quickly launch EC2 instances with specific operating systems, applications, or software stacks.

5. Elastic Block Store (EBS): Provides persistent storage for EC2 instances, with the ability to attach and detach volumes as needed.

6. Key Pair Authentication: EC2 instances are accessed using SSH (for Linux) or RDP (for Windows) via secure key pairs, providing an additional layer of security.

7. Spot Instances: Allows users to bid for unused EC2 capacity at lower prices, which is ideal for cost-sensitive workloads.

Case Study: Using Amazon EC2 for Hosting a Web Application

Imagine a company, TechSmart, that wants to build and host a web application for its customers. The application needs to handle varying levels of traffic, store user data securely, and scale dynamically as traffic fluctuates. TechSmart decides to use AWS EC2 for its backend.

Steps TechSmart Took:

1. Instance Selection: TechSmart chose an m5.large EC2 instance, as it provided a good balance of compute, memory, and storage for their web application.

2. Configuration: They set up an Amazon Machine Image (AMI) with Ubuntu, Apache Web Server, and MySQL. This provided a pre-configured environment that allowed them to deploy their

application quickly.

3. Scaling: TechSmart set up Auto Scaling to handle the variable traffic. During peak times, additional instances were automatically launched, and during off-peak hours, instances were terminated to save costs.

4. Load Balancing: They configured an Elastic Load Balancer (ELB) to distribute incoming traffic across multiple EC2 instances, ensuring that the application remained highly available even when traffic surged.

5. Persistent Storage: TechSmart used Elastic Block Store (EBS) to store customer data, ensuring the data was persistent even when instances were stopped or restarted.

6. Security: They set up a Virtual Private Cloud (VPC) to isolate their instances and used Security Groups to control inbound and outbound traffic to their EC2 instances. Additionally, they used Key Pairs for secure SSH access to their instances.

7. Monitoring: TechSmart integrated Amazon CloudWatch to monitor the performance of their EC2 instances. This allowed them to track CPU utilization, memory usage, and disk I/O to optimize their infrastructure.

8. Cost Management: By using Spot Instances during non-critical operations (e.g., batch processing), they were able to significantly reduce costs while maintaining reliable performance during high-traffic periods.

Results:

- The application was highly available and scalable, automatically adjusting to varying traffic loads.

- TechSmart saved money by only paying for the compute capacity they used and by utilizing spot instances when possible.

- The infrastructure could be easily managed and monitored through AWS's tools, providing deep insights into the health and performance of the system.

Conclusion

Amazon EC2 is a powerful and flexible cloud computing service that helps businesses scale their infrastructure easily while keeping costs under control. The case study of TechSmart shows how EC2, along with other AWS services, can be leveraged to build highly scalable, secure, and cost-efficient web applications.