Intro to Amazon Web Services

Cloud Computing

Brenden west

Contents

Learning Outcomes

- What is Amazon Web Services (AWS)
- AWS responsibility model
- AWS Identity & Access (IAM)
- Core AWS services
- Ways to access AWS

Reading

- Cloud Computing: Concepts ...,

What is AWS

- A secure cloud platform offering a broad set of services
- Provides on-demand access to virtualized IT resources and management tools
- Provides services on pay-as-you-go model
- Services designed to work together for composing complex systems
- Services are categorized (e.g. compute, storage, database, identity, etc)
- Services are supported by globally distributed & redundant physical infrastructure

AWS Global Infrastructure

- Scalable, fault-tolerant, highly available
- AWS infrastructure is provided in globally distributed **regions**
- Each region is a physical geographic location optimal for proximity to end users
- Regions are isolated from each other, but developers can replicate data across regions
- Each region consists of two or more **Availability Zones** connected by high-speed private networks
- Availability zones each have own power infrastructure & are physically separated
- Developers can replicate data & resources between zones for resiliency
- Each availability zone consists of one or more physical data centers

AWS Shared Responsibility Model

AWS

- Operate physical infrastructure. Ensure reliability and physical security

Customer

- Choose regions, availability zones
- Replicate data as needed
- Configure appropriate user access

Selecting a Region

- Data governance & legal requirements - e.g. EU data
- Proximity to customers (low latency)
- Service availability
- Regional costs

Core AWS Services

AWS has a wide & sometimes confusing array of services. Some core services a new cloud developer will encounter:

- **Compute** - EC2, ECS, ELB, Lambda
- **Storage** - S3, EBS
- **Networking** - VPC, Route 53, CloudFront
- **Database** - RDS, DynamoDB
- **Identity** - IAM, Artifact, KMS
- **Management** - Config, Console, CloudWatch, CLI
- **Cost Management** - Cost & Usage Report, Cost Explorer, Budgets

Accessing AWS

Before using AWS:

- Create an AWS account (aka root user)
- Create an Identity & Access Management (IAM) user
- Set up IAM user permissions

Interacting with AWS

AWS supports 3 means for interacting with services through a REST-like API:

- Management Console
- Command Line Interface (CLI)
- Software Development Kits (SDKs)

AWS also provides several browser-based tools for using the CLI or SDKs:

- AWS CloudShell - a browser-based terminal for running CLI commands
- AWS Cloud9 - an integrated development environment (IDE)
-

AWS Management Console

- Browser-based UI for managing AWS resources
- Provides access to development tools (e.g. CloudShell, Cloud9)

AWS CLI

- AWS service access through terminal-based commands or scripts
- Can run on Windows, MacOS, Linux operating systems
- Uses AWS credentials stored on client computer

AWS SDKs

- Language or platform-specific libraries for accessing AWS services programmatically (e.g. Python, Java, Android, etc)
- SDK mediates request / response between a client application and AWS services

AWS CloudShell

- Available in the AWS Console
- Inherits credentials from AWS Console
- Runs fully-managed Amazon Linux 2 instance with AWS CLI & SDKs pre-installed
- Has per-region persistent storage
- Pay only for AWS resources you create & run