**AWS** is a platform of web service for storing, networking, and computing that works well together.

**The AWS access key** can verify identity in programmatic calls.

**AWS benefits:**

Services solve common problems.

Automation.

Scalability.

Reliability.

Global infrastructure.

Reducing time.

**Virtual machines** provide a flexible and efficient way to run multiple operating systems and applications on a physical machine.

**Elastic Beanstalk** is a service for deploying and scaling web applications and services.

**Why Elastic Beanstalk:** The fastest way to get web applications up and running on AWS.

**How to deploy an application:**

Create an application.

Explore environment.

Deploy a new version of the application.

Configure environment.

Clean up.

**AWS tools to call API:**

*Command-line interface* (CLI) – Call AWS API from the terminal.

*Software development kit* (SDK) – Call AWS API from the programming language.

*AWS CloudFormation* – Translate the templates used to describe the state of the infrastructure into AWS API.

**AWS account** contains all the AWS resources and comes with a root user.

**IAM user:**

IAM entity assigned to a person.

Password.

Access key.

Belong to a group.

Security Token services are **permanent**.

**IAM role:**

IAM entity assigned to a service.

Security Token services are **temporary**.

Can be associated with an **EC2, Lambda function, or ECS container**.

**IAM policy:**

*Identity policies:* attached to users, groups, or roles.

*Managed policy*: can be reused in your account.

*Inline policy:* belong to a specific IAM role, user, or group.

*Resource policies:* attached to resources.

**IAM is about:**

Allow or deny?

What actions?

On which resources?

Who? Authentication and authorization.

Condition.

**VPC** is a virtual network. If you want to create networked resources in AWS, you will create a VPC first.

**The Internet gateway** is a resource created and added in a VPC. IGW allows for routing traffic between the VPC and the public internet.

**The NAT gateway:** routing traffic to the internet but not back in.

**Route tables:** directs all nonlocal traffic from the subnet to the gateway.

**The security group** is a set of rules determining what network traffic is allowed in and out of an instance.

Inbound

Outbound

**S3:** Amazon Simple Storage Service is a typical web service that lets you store and retrieve data organized as objects via API reachable over HTTPS.

**S3 use cases:**

Storing and delivering static web content.

Backing up data.

Storing structured data for analytics, also called a data lake.

Storing and delivering user-generated content.

**S3 uses buckets** with globally unique names to group objects.

**RDS**: Amazon Relation Database Service is a managed service operating a relational database system (Aurora, PostgreSQL, MySQL, MariaDB, Oracle Database, and Microsoft SQL Server).

**Controlling Access to RDS:**

*Configuration access management*: **IAM policies**

*Network access management:* **Firewall rules**

*Database access management:* **Database** user and access management itself

*Data:* **Sensitive data** must be protected from unauthorized access

**DynamoDB** is a fully managed NoSQL database service provided by AWS. It is designed for applications that require low latency and high scalability, closed source key-value store with document support in handling large amounts.

**DynamoDB** dose not run in VPC. It is accessed via an API.