Cloud Computing Introduction

CS516 – Cloud Computing

Computer Science Department

Maharishi International University

Maharishi International University - Fairfield, Iowa

Content

- What is cloud?
- Cloud computing components (Virtualization)
- Cloud Vendors
- Models of Cloud Services (IaaS, FaaS, serverless, SaaS)
- AWS well architected
- Most common AWS services
- IAM intro

What is Cloud?

The term cloud means storing data over the internet.

Cloud isn't just storage, it’s computing, it’s AI, It’s block chain, it’s satellite, you name it. Cloud is a whole bunch of different types of **services**.

Compute
- EC2
- Lightsail
- Lambda
- Batch
- Elastic Beanstalk
- Serverless Application Repository
- AWS Outposts
- EC2 Image Builder
- AWS App Runner

Containers
- Elastic Container Registry
- Elastic Container Service
- Elastic Kubernetes Service
- Red Hat OpenShift Service on AWS

Storage
- S3
- EFS
- FSx
- S3 Glacier
- Storage Gateway
- AWS Backup

Customer Enablement
- AWS IQ
- Support
- Managed Services
- Activate for Startups

Robotics
- AWS RoboMaker

Blockchain
- Amazon Managed Blockchain

Satellite
- Ground Station

Quantum Technologies
- Amazon Braket

Management & Governance
- AWS Organizations
- CloudWatch
- AWS Auto Scaling
- CloudFormation
- CloudTrail

Machine Learning
- Amazon SageMaker
- Amazon Augmented AI
- Amazon CodeGuru
- Amazon DevOps Guru
- Amazon Comprehend
- Amazon Forecast
- Amazon Fraud Detector
- Amazon Kendra
- Amazon Lex
- Amazon Personalize
- Amazon Polly
- Amazon Rekognition
- Amazon Textract
- Amazon Transcribe
- Amazon Translate
- AWS DeepComposer
- AWS DeepLens
- AWS DeepRacer
- AWS Panorama
- Amazon Monitron
- Amazon HealthLake
- Amazon Lookout for Vision
- Amazon Lookout for Equipment

AWS Cost Management
- AWS Cost Explorer
- AWS Budgets
- AWS Marketplace Subscriptions
- AWS Application Cost Profiler

Front-end Web & Mobile
- AWS Amplify
- Mobile Hub
- AWS AppSync
- Device Farm
- Amazon Location Service

AR & VR
- Amazon Sumerian

Application Integration
- Step Functions
- Amazon AppFlow
- Amazon EventBridge
- Amazon MQ
- Simple Notification Service
- Simple Queue Service
- SWF

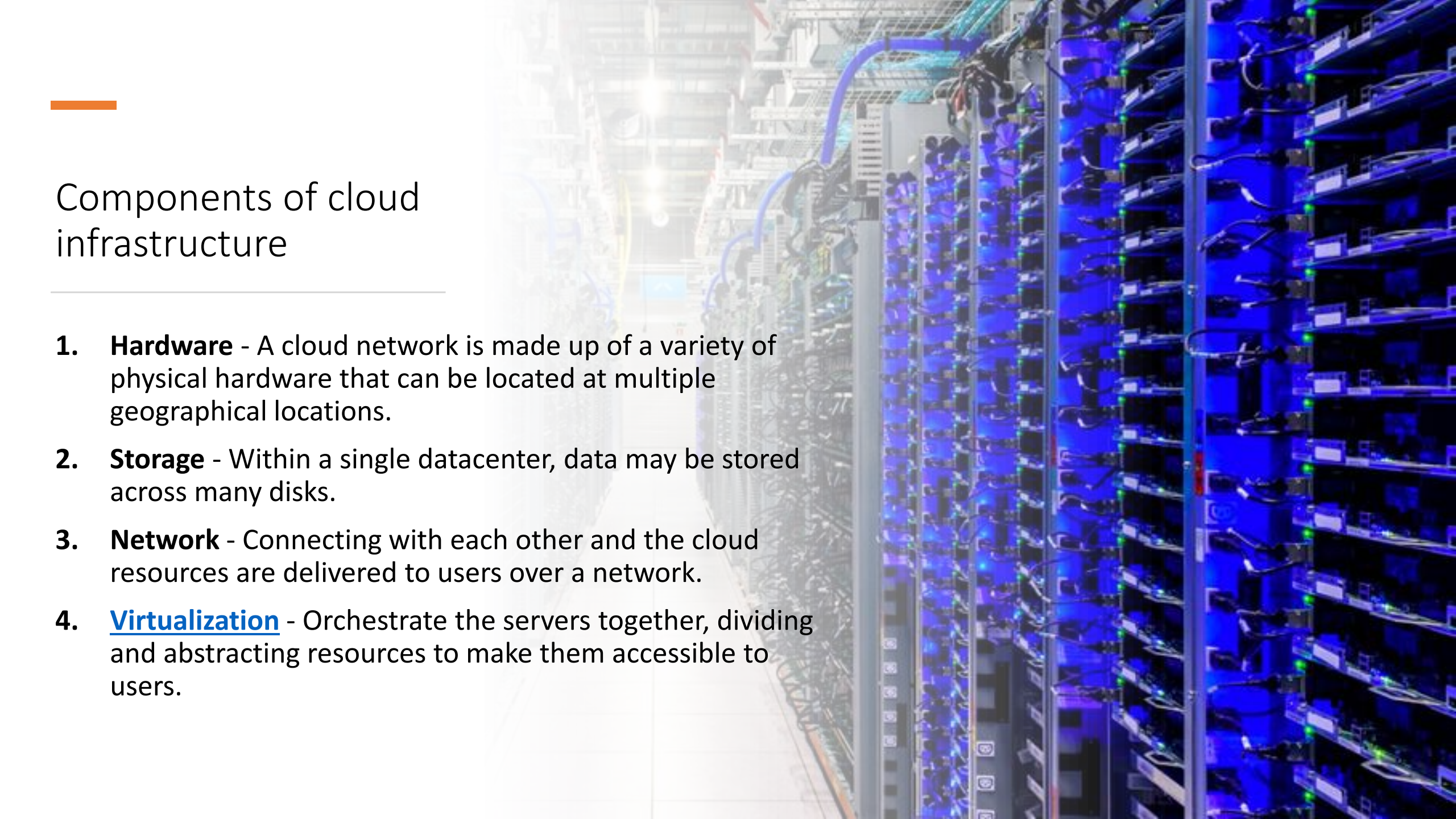
What are services?

Web services are just HTTP endpoints that we call using CLI, SDK by passing required parameters, payload, and authorization for building applications by creating resources on the cloud.

AWS (Amazon Web Services) has built UI on top of the services that we call AWS console.

```
const AWS = require("aws-sdk");
const dynamodb = new AWS.DynamoDB({ apiVersion: "2012-08-10" });
const scanParams = {
    TableName: tableName
};
dynamodb.scan(scanParams);
```

Components of cloud infrastructure



1. **Hardware** - A cloud network is made up of a variety of physical hardware that can be located at multiple geographical locations.
2. **Storage** - Within a single datacenter, data may be stored across many disks.
3. **Network** - Connecting with each other and the cloud resources are delivered to users over a network.
4. **Virtualization** - Orchestrate the servers together, dividing and abstracting resources to make them accessible to users.

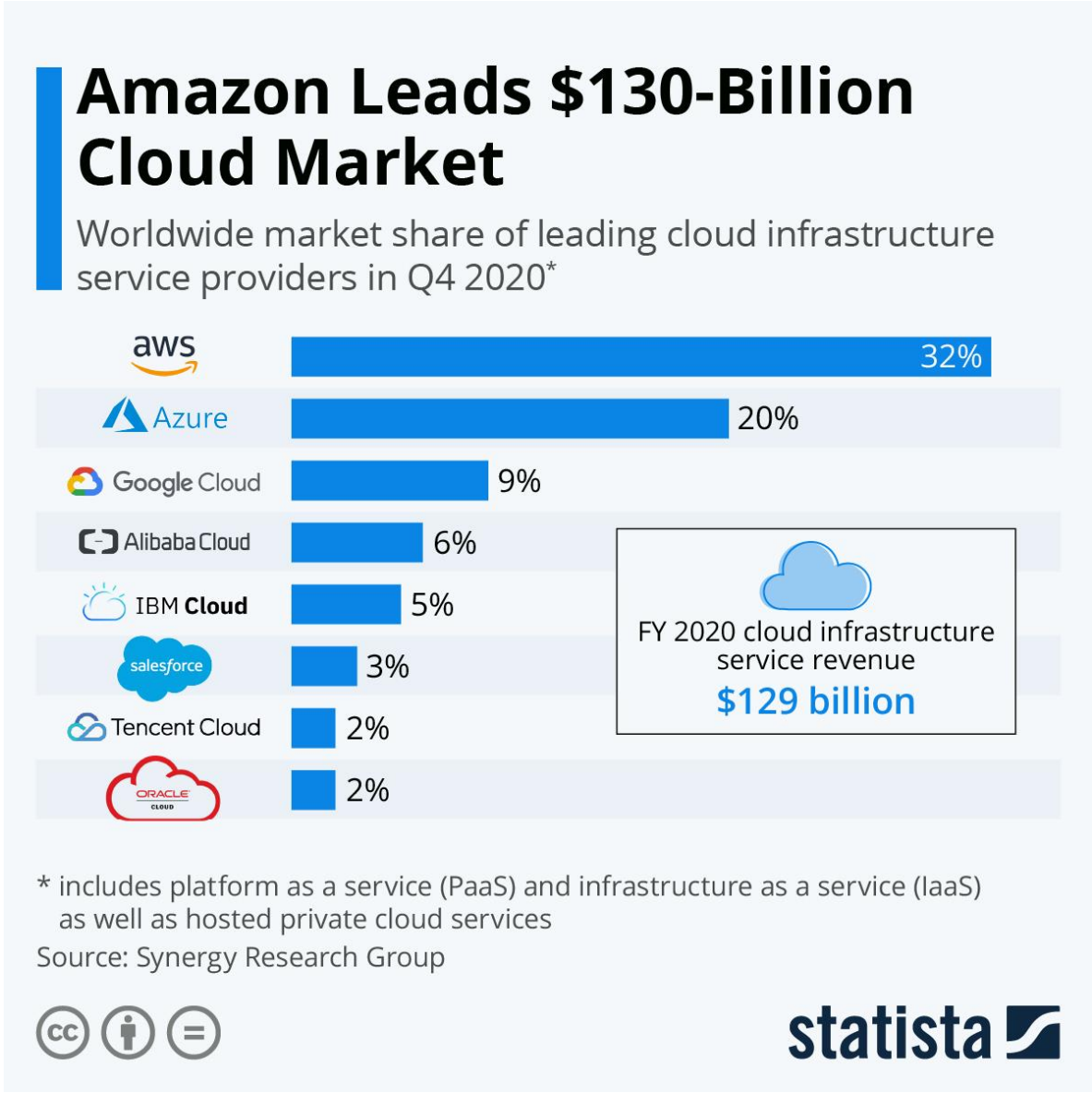
Cloud Vendors

There are 2 types of Cloud Vendors:

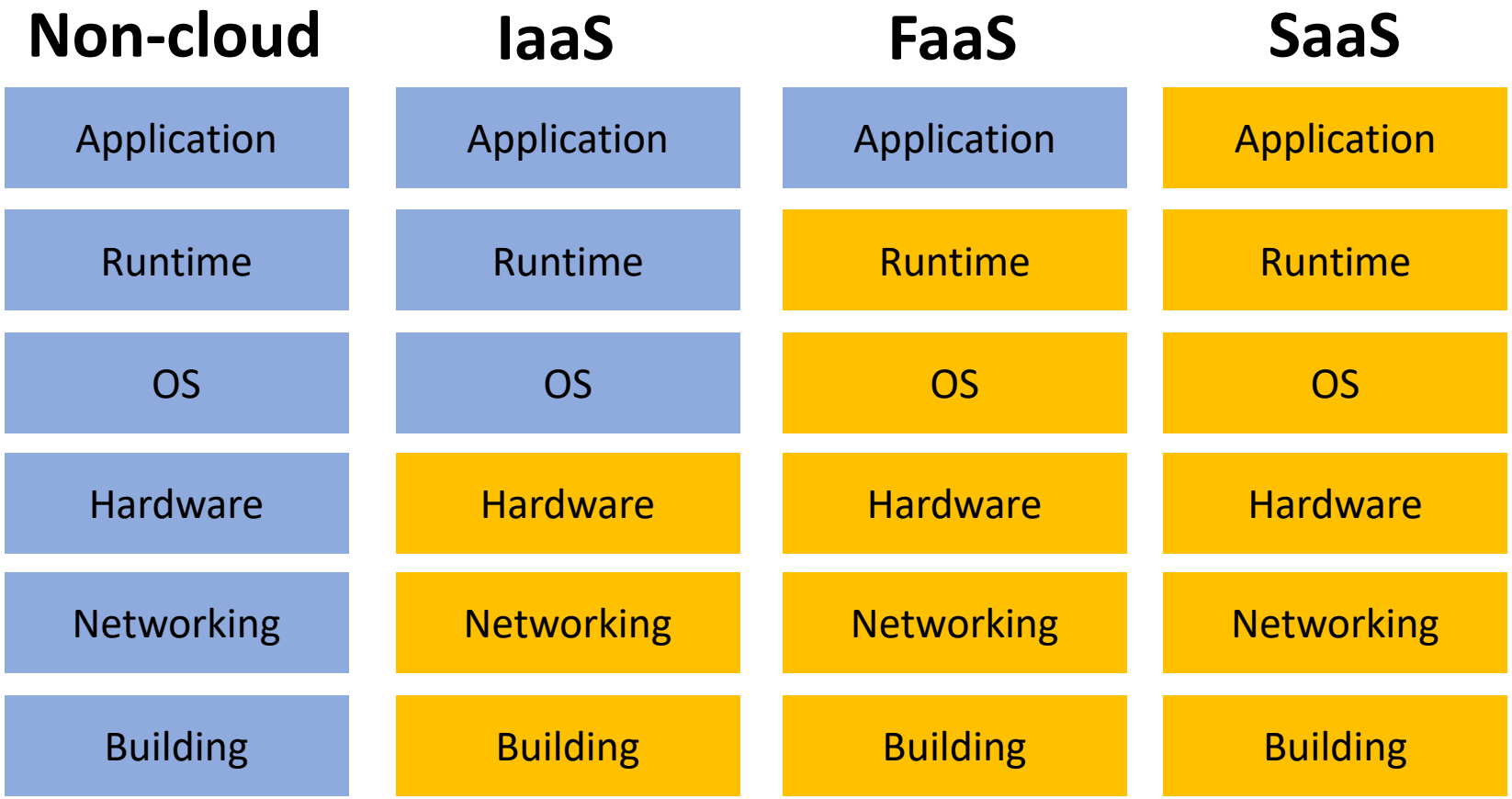
1. Private:
   - Dedicated for use by a single organization.
   - The data center resources may be located on-premise or operated by a third-party vendor off-site.
   - Private cloud is customizable to meet the unique business and security needs of the organization. Examples: VMware, Red Had Cloud, Citrix Cloud.
2. Public
   - Variety of services. Use them as you need.
   - Anyone can use it. Hardware is shared. Examples: AWS, Azure, Google Cloud.

Cloud Vendors

Cloud Computing Industry to Grow from $371.4 Billion in 2020 to $832.1 Billion by 2025, at a Compound Annual Growth Rate 17.5%.



Models of Cloud Services



IaaS

Infrastructure as a Service is where you have a hypervisor, and you can provision **virtual machines**, or it could be where you're renting the physical server itself.

It allows you to host an operating system on the cloud that is unique to you, nobody else can log into it. You go in and then manage it yourself.

With Infrastructure as a Service, you only pay for what you use.

FaaS

Function as a Service is a category of cloud computing services that provides a platform allowing customers to develop, run, and manage **application functionalities without the complexity of building and maintaining the infrastructure**.

Building an application following this model is one way of achieving a **serverless** architecture and is typically used when building event-driven and microservices applications.

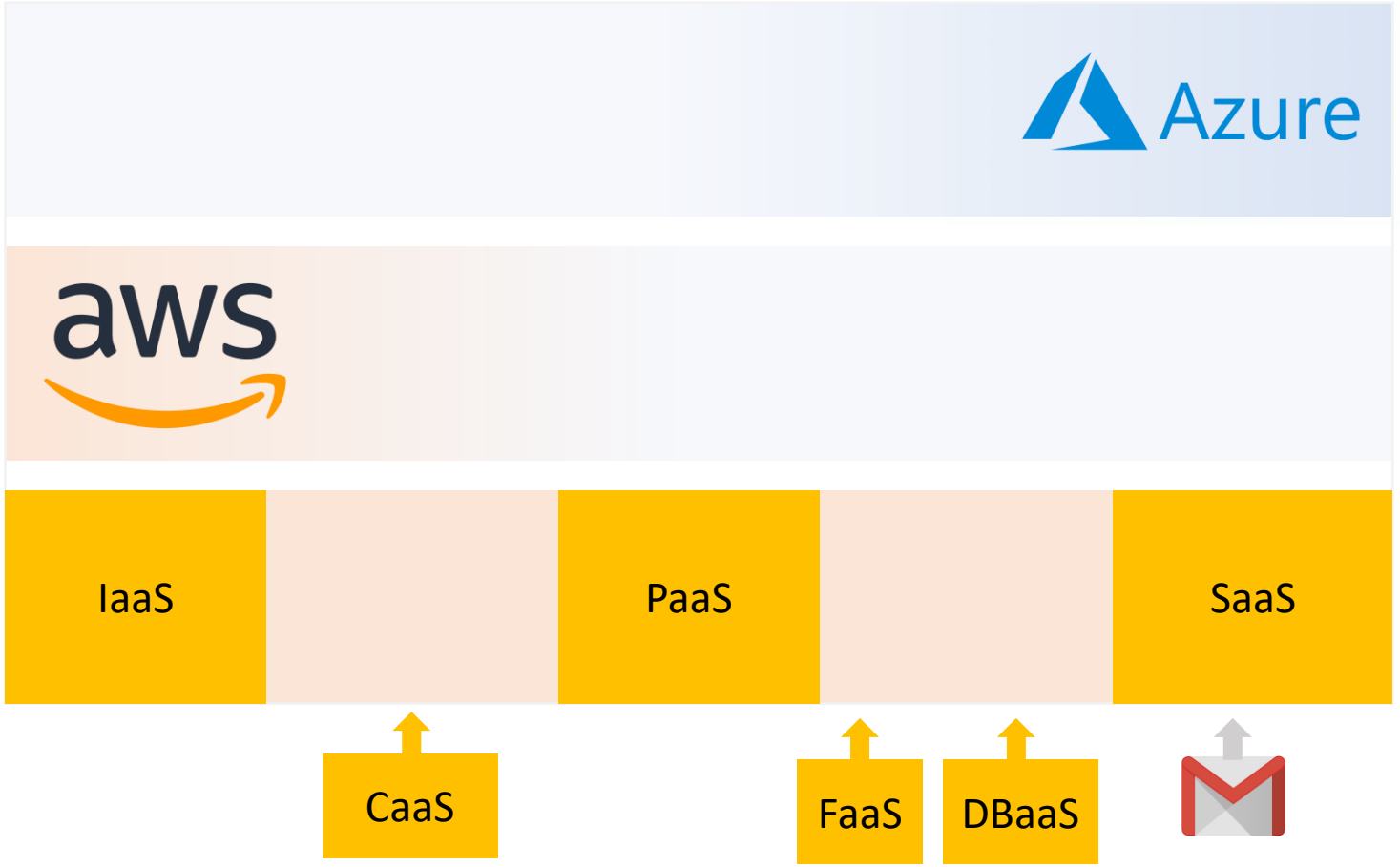
SaaS

Software as a Service is something like Gmail. With Gmail, all you worry about is using the actual software, about creating messages, filtering spam filters.

You're not worried about the underlying servers, how they are load balanced, high availability, DNS resolving etc.

As a Service!

- **Content** as a service
- **Data** as a service
- **Desktop** as a service
- **Function** as a service
- **Infrastructure** as a service
- **Integration** as a service
- **Network** as a service
- **Platform** as a service
- **Security** as a service
- **Software** as a service



PaaS

Platform as a Service is where you don't know what resources you need but you've just got your code and you will then use Platform as a Service to go in and provision those resources for you.

You still have to look after the underlying assets but you don't have to worry about the provisioning of it.

Serverless

Serverless computing is a cloud computing execution model in which the cloud provider allocates machine resources on demand, taking care of the servers on behalf of their customers.

When an app is not in use, there are no computing resources allocated to the app. Pricing is based on the actual amount of resources consumed by an application. Pure pay-as-you-go model. On the other hand, EC2 costs when running even if no one is using.

Benefits of using cloud

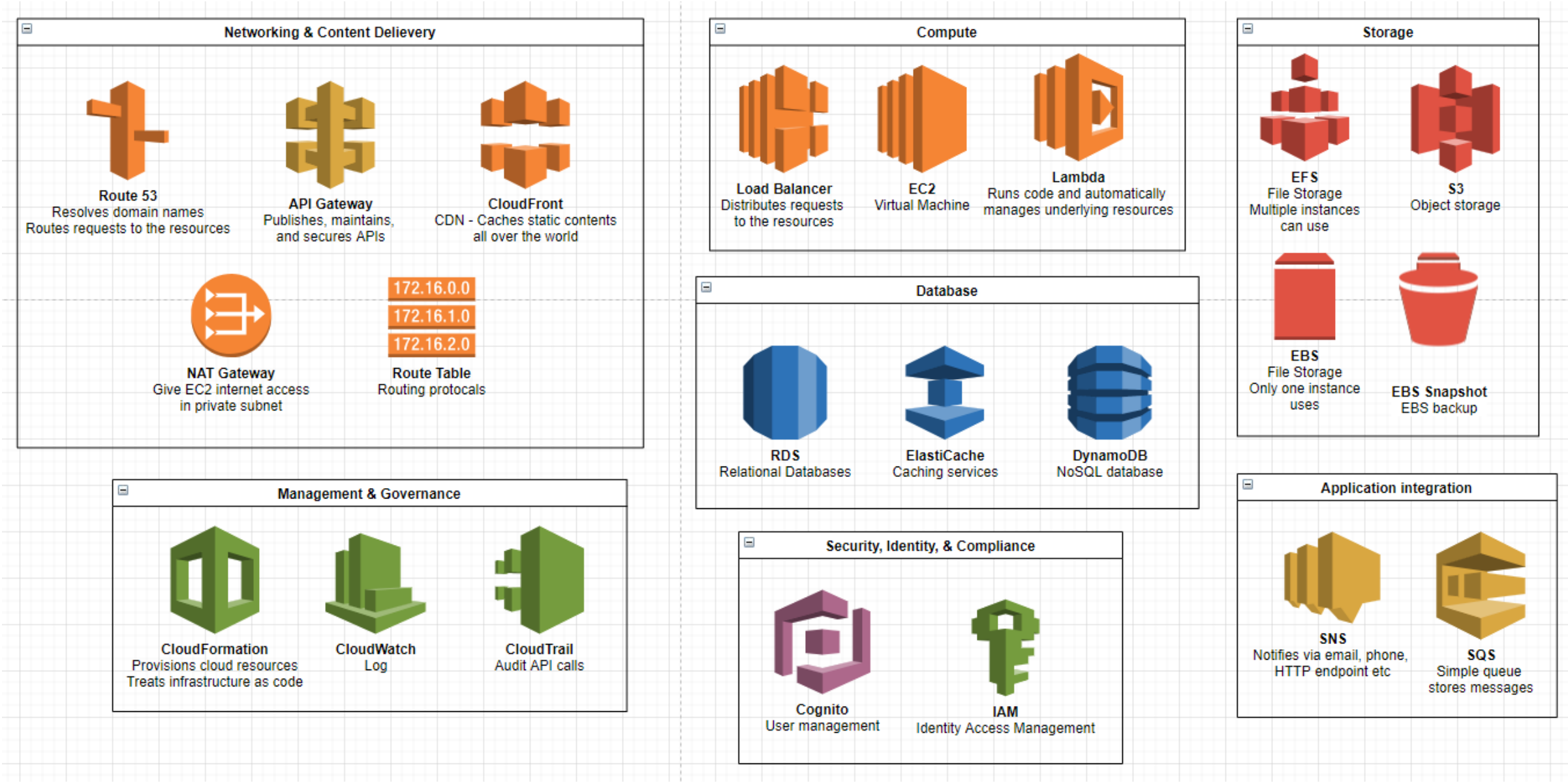
- You don’t need to build a module from scratch that would take lots of time. Potentially, it could have some bugs. Instead, use proven technologies that solve the puzzle and make developers life easier.
- Focus on the business. Cloud providers take care of the technical details.
- There are solutions that you can integrate. That will solve challenges and you don’t need to pay for licenses.

AWS Well Architected – 5 pillars

AWS Well-Architected helps cloud architects build secure, high-performing, resilient, and efficient infrastructure for their applications and workloads. Based on five pillars.

Read More about: [AWS Well Architected]

Amazon Web Services



AWS Global Infrastructure Map

https://aws.amazon.com/about-aws/global-infrastructure/

Identity & Access Management (IAM)

AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources.

You use IAM to control who (**user** or **role**) is authenticated (signed in) and authorized (has **permissions**) to use resources.

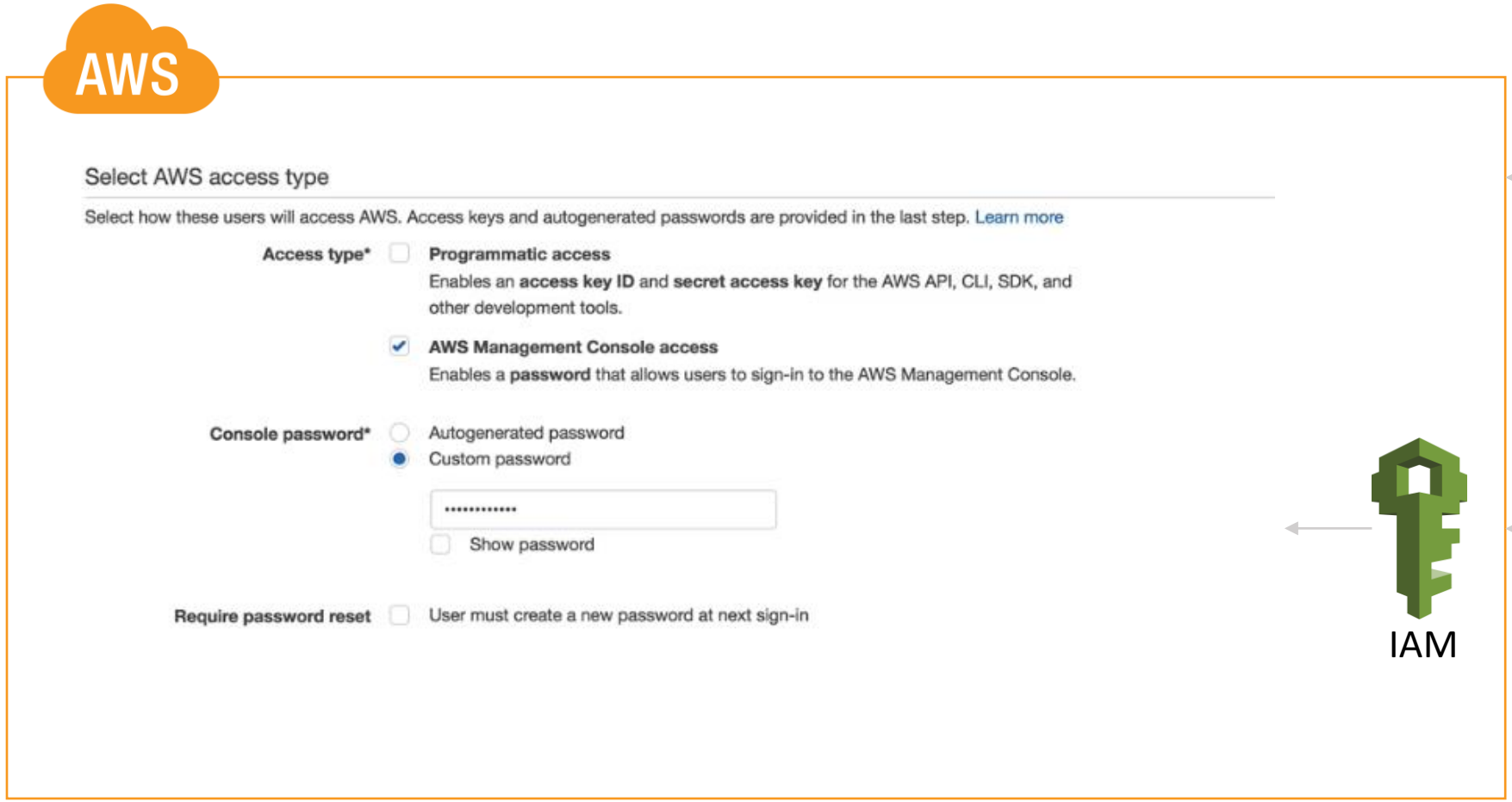
AWS Identity and Access Management - IAM

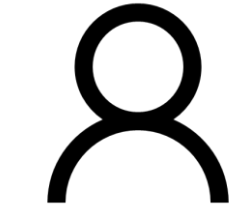
IAM is used to manage:

- IAM Access **Policies** *We attach **IAM Policies** to Users, Groups, and Roles.*
- **Users & Groups**
- **Roles**

Note: The user created when you created the AWS account is called the **root user**. By default, the root user has **full administrative** rights and access to every part of the account. Any new or additional users you create in the AWS account are created with **no access** to anything by default.

Creating an IAM user





AWS Account Owner

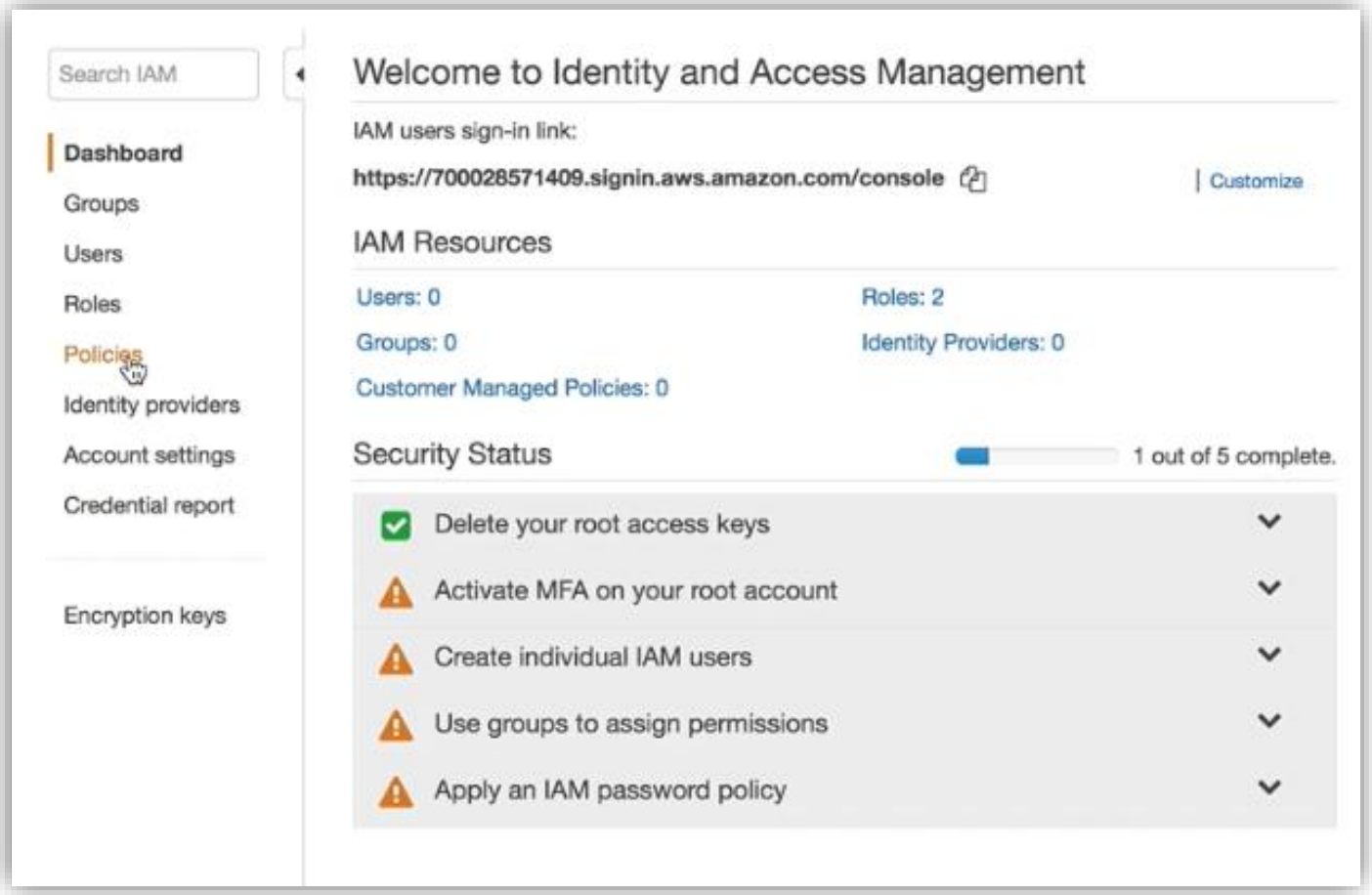
IAM

George
Mike
(developers)

IAM user configurations

IAM users can have **Access Key ID** and **Secret Access Key**. Both are used when you grant the user programmatic access for the AWS API via CLI and SDK.

IAM users have **Password** that allows them to sign-in to the AWS Management Console.

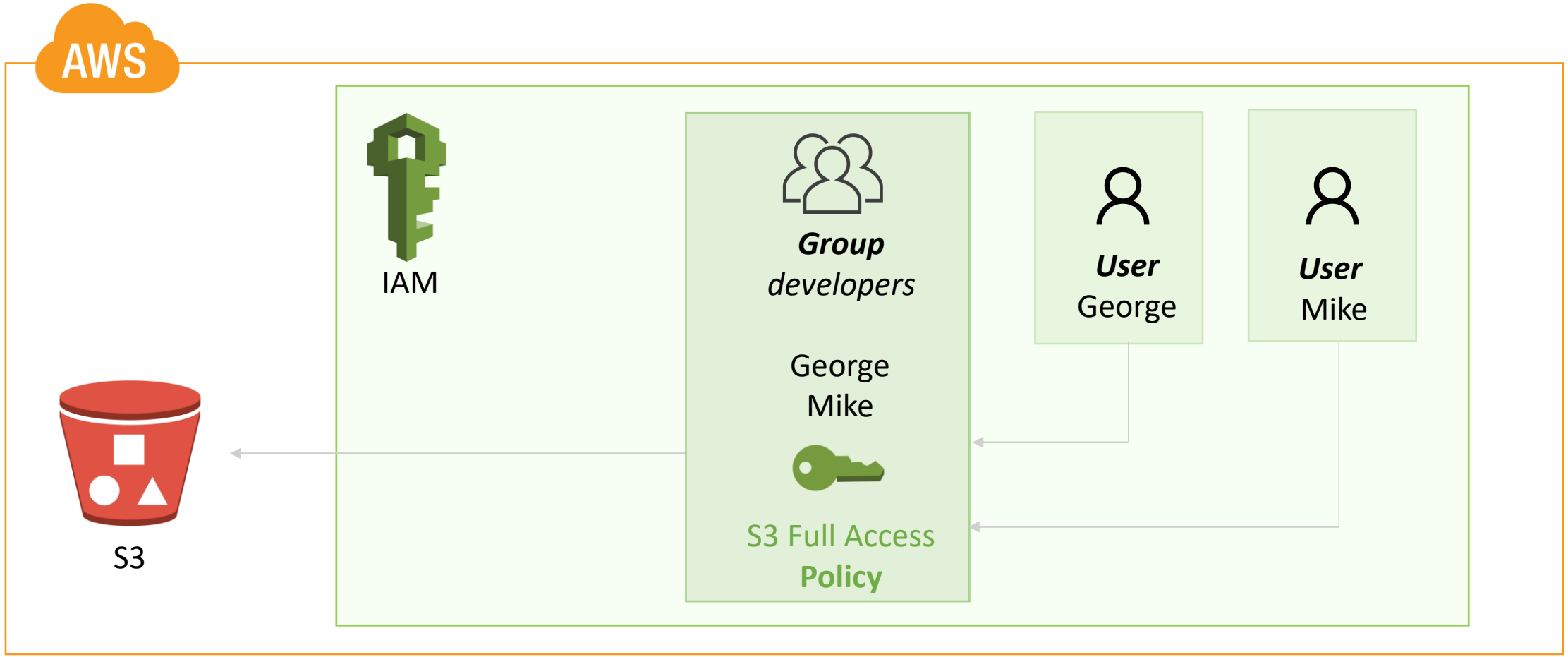


IAM Policies

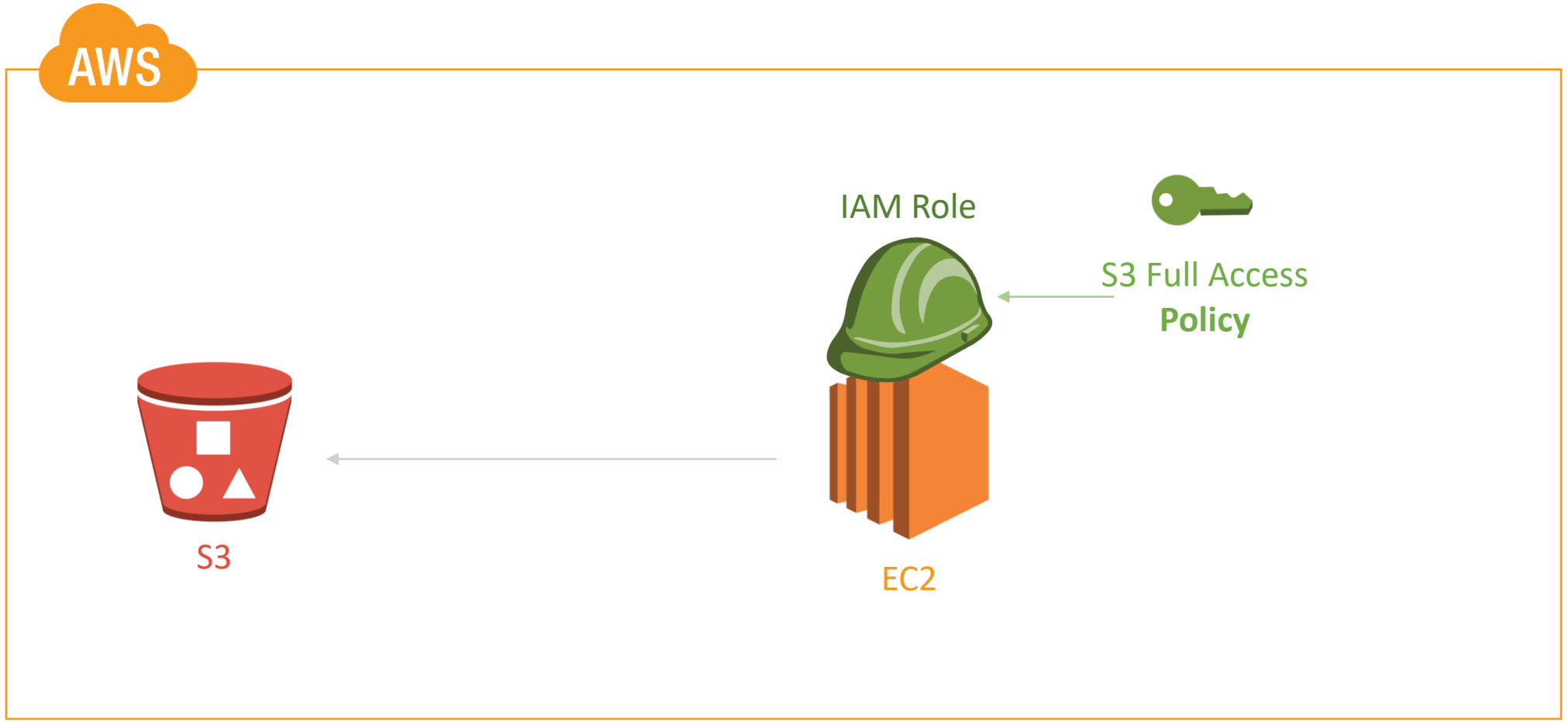
- **IAM Policies** are **permissions** that you can assigned to any User, Group, and Roles.
- It is best to provide IAM Policies to Groups rather than Users.
- Users can be members up to 10 groups.
- When any user tries to access an AWS Service, **IAM Policy** will be checked to see if has access permission.
- We don't attach a IAM Policy to a Service, instead we would need to use a **Role**.

Read More about: IAM Users, IAM Policies, IAM Roles

AWS IAM Group



AWS IAM Role



IAM Configurations

When a new AWS root account is created, it is best practice to complete the tasks listed in IAM under "Security Status":

- **Activate MFA on your root account** (Multi-Factor Authentication)
- **Create individual IAM users** (never use your root account for daily use. Instead, create an IAM user and attach the *AdministratorAccess* policy to it, they will have a different Console link to sign-in).
- **Use groups to assign permissions** (IAM group is a collection of IAM users. Groups allow you to set and manage permissions for multiple users at the same time).
- **Apply an IAM password policy**

aws

Contact Us | Support | English | My Account | Sign In to the Console

Products | Solutions | Pricing | Documentation | Learn | Partner Network | AWS Marketplace | Customer Enablement | Events | Explore More

Customer Success Stories

Learn how organizations of all sizes use AWS to increase agility, lower costs, and accelerate innovation in the cloud.

Search customer stories

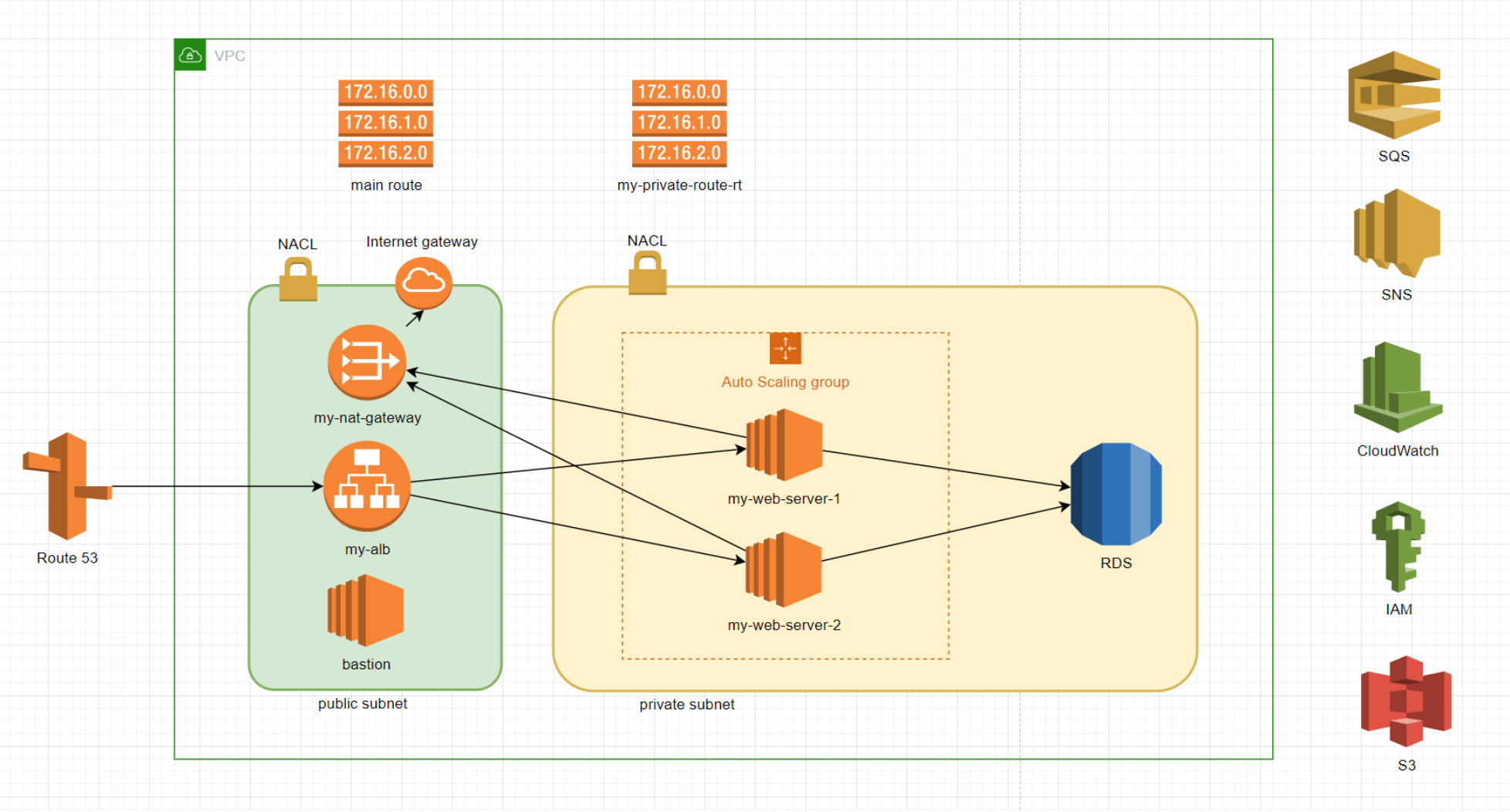
AWS Customer News

Sun Life taps AWS as its long-term strategic cloud provider to innovate for clients, develop employees, and become a more agile business in the cloud. Read the press release »

Featured Customers

https://aws.amazon.com/solutions/case-studies/

The first half of the course (IaaS)



The second half of the course (FaaS)