Getting Started with AWS

Oussema Miled · Follow
5 min read · May 4, 2021

This article introduce you to AWS covering some of its advantages and some basic topic which can help you start learning and using AWS.

Satisfying the needs

In today's competitive market, customers demand low latency, highly scalable, responsive applications that work all the time. Customers expect to receive the same level of performance and consistency of applications regardless of their location and how often they use those applications.

With the classic infrastructures, there are always problems related to cost, flexibility, scalability and more. The cloud solved all these issues.

The AWS Cloud provides infrastructure services, such as compute, storage, networking, and databases, and a broad set of platform capabilities such as mobile services, analytics, and machine learning (ML). These services are available on demand, through the internet, and with pay-as-you-go pricing. With these features and capabilities, AWS fully satisfies the needs of the companies, developers, users...

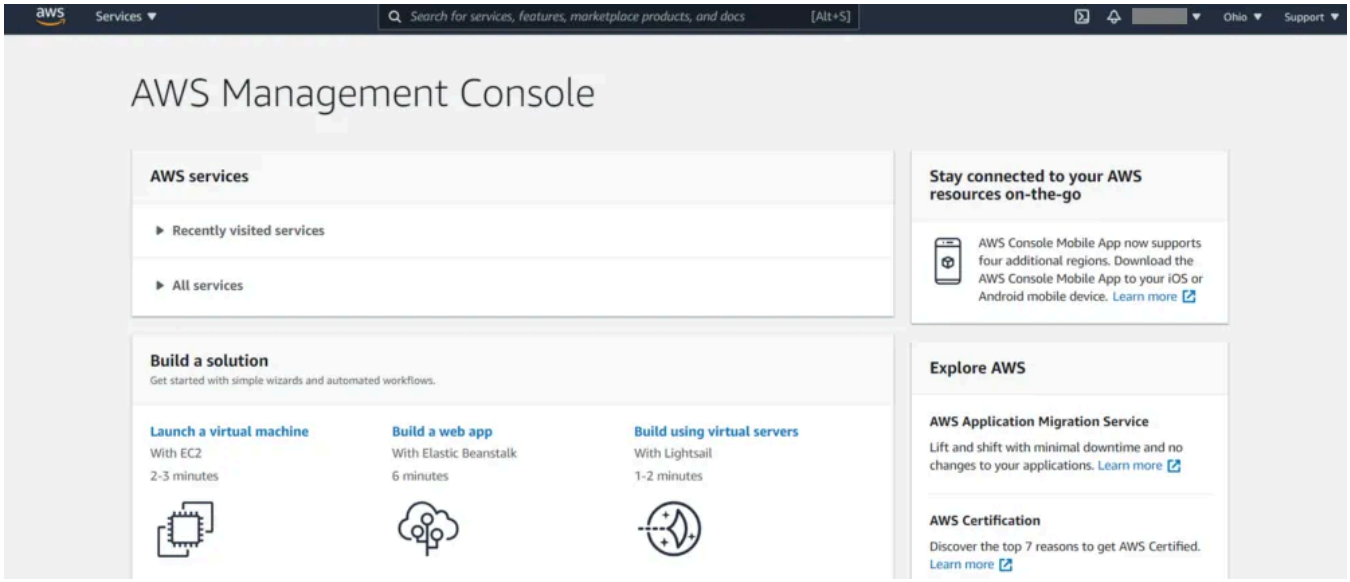
Despite there is other significant and powerful cloud providers such as Microsoft Azure and Google Cloud Platform, AWS proved itself to be the best out there for many reasons which you can learn more about following this link.

Create an account to start

Obviously, so that we can use AWS, we first need to create an account and activate it providing a valid credit card. It's as easy as following some steps and it won't take more than 5 minutes and also don't worry, you don't need to pay anything as long as you use the free tier plan. Jeff is generous enough to provide us with hundreds of services to use for free always or for the first 12 months or via trials. Here's the full list.

AWS Management Console, SDKs & CLI

After creating an account, we can start to explore the AWS Management Console



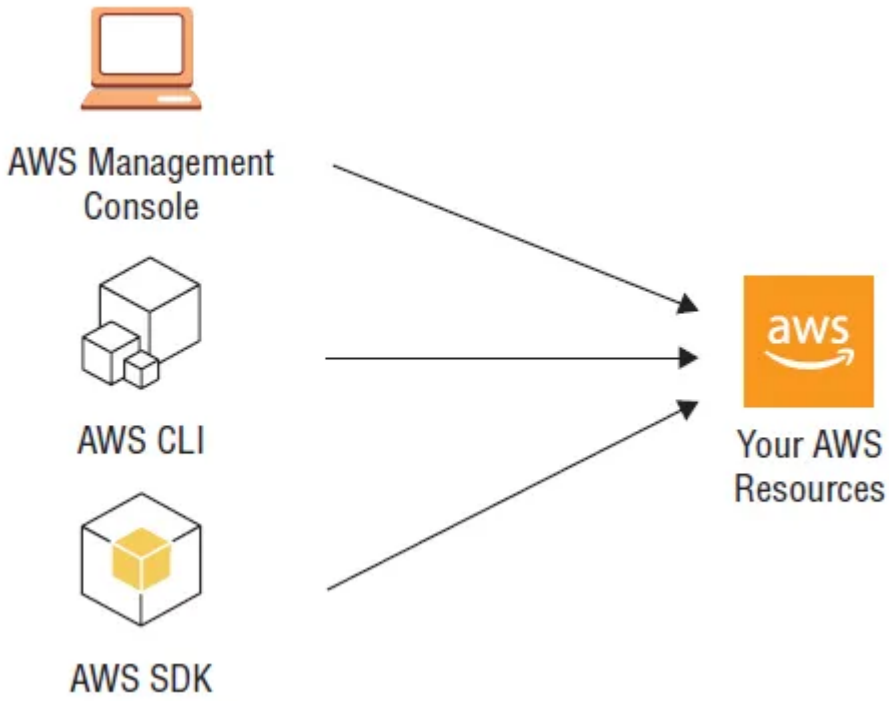
Previewing AWS MC

Because all the functionalities of AWS is exposed through APIs, AWS provides more than only the web interface for managing resources. For example, the console is

also available as a mobile app for iOS and for Android.

I recommend that you take the time to explore the management console, discover what you can do on it, list and learn more about AWS services etc.

After you become familiar with a service, you can manage AWS resources programmatically through either the AWS Command Line Interface (AWS CLI) or the AWS software development kits (AWS SDKs), as shown below.



Options for managing AWS resources

AWS SDKs

If you're not familiar with SDKs, basically a software development toolkit (SDK) is a set of software tools and programs provided by hardware and software vendors that developers can use to build applications for specific platforms.

AWS SDKs are available in many popular programming languages such as Java, .NET, JavaScript, PHP, Python, Ruby, Go, and C++. AWS also provides specialty SDKs such as the AWS Mobile SDK and AWS Internet of Things (IoT) Device SDK.

AWS CLI Tools

In addition to the AWS Management Console and SDKs, AWS provides tools to manage AWS resources from the command line. One such tool is the AWS CLI, which is available on Windows, Linux/Unix, and macOS.

The AWS CLI allows you to perform actions similar to those from the SDKs but in an interactive scripting environment. Because the AWS CLI is interactive, it is a good environment for experimenting with AWS features. Also, the AWS CLI and the SDK on the same server can share configuration settings.

Working with Regions

AWS operates facilities in multiple regions across the world, as shown below. Each AWS Region is designed to be isolated from the other AWS Regions. This design achieves the greatest possible fault tolerance and stability.

AWS Global Infrastructure Map

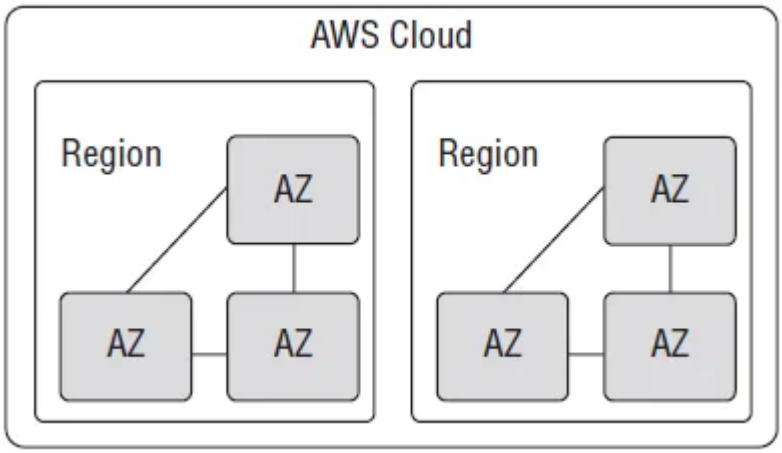
When you view your resources, you see only the resources that are tied to the AWS Region that you specified (on the top right of the management console).

AWS has 80 Availability Zones (we'll talk about it in a moment) across 25 geographic regions, with plans to launch 15 more Availability Zones and five more AWS Regions. The number is growing each year.

Availability Zones

Each AWS Region contains multiple data centers, grouped together to form Availability Zones. Regions are composed of multiple Availability Zones, which allows AWS to provide highly available services in a way that differentiates them from traditional architectures with single or multiple data centers.

Availability Zones are physically separated from each other and are designed to operate independently from each other in the case of a fault or natural disaster.



Regions and Availability Zones

AWS customers can improve the resilience of their applications by deploying a copy of each application to a second Availability Zone within the same region. This allows the application to remain available to customers even in the face of events that could disrupt an entire data center. Similarly, many of the AWS services automatically replicate data across multiple Availability Zones within an AWS Region to provide high availability and durability of the data.

An example of an AWS service that replicates data across Availability Zones within a region is Amazon Simple Storage Service (Amazon S3). Amazon S3 enables you to upload files and store those files as objects within a bucket. By default, Amazon S3 automatically replicates objects across a minimum of three Availability Zones within the region hosting the bucket. This design protects data even against the loss of one entire Availability Zone.

Choosing a Region

One factor for choosing an AWS Region is the availability of the services required by your application. Other aspects to consider when choosing a region include latency, price, and data residency. The image below describes selection criteria to include when choosing an AWS Region.

Selection Criteria	Description
Service availability	Choose a region that has all or most of the services you intend to use. Each region exposes its own AWS Cloud service endpoints, and not all AWS services are available in all regions.
Proximity and latency	Choose a region closer to application users, on-premises servers, or your other workloads. This allows you to decrease the latency of API calls.
Data residency	Choose a region that allows you to stay compliant with regulatory or contractual requirements to store data within a specific geographic region.
Business continuity	Choose a pair of regions based on any specific requirements regarding data replication for disaster recovery. For example, you may select a second AWS Region as a target for replicating data based on its distance from the primary AWS Region.
Price	AWS service prices are set per region. Consider cost when service availability and latency are similar between candidate regions.

Selecting an AWS Region

That's it for now

I guess the mentioned informations and topics that I included in this article are enough as an introduction and to stimulate your curiosity so you learn more about it. Personally, I found AWS very interesting and fun to work with and I am happy to share what I learned with you from time to time. Stay Tuned! 😄

PS: If you like what I do and want to support me, you can do that by becoming a medium member using this link

Cloud Computing AWS Learning Amazon Programming

Follow

Written by Oussema Miled

97 Followers · 5 Following

A Computer Science Engineer who loves to talk about Web Development and sometimes other stuff!