**AWS**

**1. What is Amazon EC2?**

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the

Amazon Web Services (AWS) cloud. Using Amazon EC2 eliminates your need to invest in

hardware up front, so you can develop and deploy applications faster. You can use Amazon EC2

to launch as many or as few virtual servers as you need, configure security and networking, and

manage storage.

**2. Instances and AMIs?**

An Amazon Machine Image (AMI) is a template that contains a software configuration (for

example, an operating system, an application server, and applications). From an AMI, you

launch an instance, which is a copy of the AMI running as a virtual server in the cloud. You can

launch multiple instances of an AMI, as shown in the following figure.

**What is IAM ?**

**AWS** Identity and Access Management (**IAM**) enables you to manage access to **AWS** services and resources securely. Using **IAM**, we can create and manage **AWS** users and groups, and use permissions to allow and deny their access to **AWS** resources.

**s3 (simple storage service)**

---------------------------

S3 is KEY/VALUE store for storing objects

• Amazon S3 allows people to store objects (files) in “buckets” (directories)

• Buckets must have a globally unique name

• Buckets are defined at the region level

• Naming convention

• No uppercase

• No underscore

• 3-63 characters long

• Not an IP

• Must start with lowercase letter or number

Objects (files) have a Key. The key is the FULL path:

• <my\_bucket>/my\_file.txt

• <my\_bucket>/my\_folder1/another\_folder/my\_file.txt

Just keys with very long names that contain slashes (“/”)

• Object Values are the content of the body:

• Max Size is 5TB

• If uploading more than 5GB, must use “multi-part upload”

• Metadata (list of text key / value pairs – system or user metadata)

• Tags (Unicode key / value pair – up to 10) – useful for security / lifecycle

• Version ID (if versioning is enabled)

**IAM Introduction**

-----------------

• IAM (Identity and Access Management)

• Your whole AWS security is there:

• Users

• Groups

• Roles

• Root account should never be used (and shared)

• Users must be created with proper permissions

• IAM is at the center of AWS

• Policies are written in JSON (JavaScript Object Notation)

• IAM has a global view -for all regions

• Permissions are governed by Policies (JSON)

• MFA (Multi Factor Authentication) can be setup

• IAM has predefined “managed policies”

• It’s best to give users the minimal amount of permissions they need to

perform their job (least privilege principles)

**What is load balancing?**

------------------------

• Load balancers are servers that forward internet traffic to multiple

servers (EC2 Instances) downstream.

Why use a load balancer?

------------------------

• **Spread load across multiple downstream instances**

• Expose a single point of access (DNS) to your application

• **Seamlessly handle failures of downstream instances**

• **Do regular health checks to your instances**

• Provide SSL termination (HTTPS) for your websites

• Enforce stickiness with cookies

• High availability across zones

• Separate public traffic from private traffic

**Why use an EC2 Load Balancer?**

-----------------------------

• An ELB (EC2 Load Balancer) is a managed load balancer

• AWS guarantees that it will be working

• AWS takes care of upgrades, maintenance, high availability

• AWS provides only a few configuration knobs

• It costs less to setup your own load balancer but it will be a lot more

effort on your end.

• It is integrated with many AWS offerings / services

**Types of load balancer on AWS**

-----------------------------

• AWS has 3 kinds of Load Balancers

• **Classic Load Balancer** (v1 - old generation) - 2009

• **Application Load Balancer** (v2 - new generation) - 2016

• **Network Load Balancer** (v2 - new generation) - 2017

• Overall, it is recommended to use the newer / v2 generation load

balancers as they provide more features

• You can setup internal (private) or external (public) ELBs

**Health Checks**

---------------

• Health Checks are crucial for Load Balancers

• They enable the load balancer to know if instances it forwards traffic to

are available to reply to requests

• The health check is done on a port and a route (/health is common)

• If the response is not 200 (OK), then the instance is unhealthy

**What’s an Auto Scaling Group?**

----------------------------

• In real-life, the load on your websites and application can change

• In the cloud, you can create and get rid of servers very quickly

• The goal of an Auto Scaling Group (ASG) is to:

• Scale out (add EC2 instances) to match an increased load

• Scale in (remove EC2 instances) to match a decreased load

• Ensure we have a minimum and a maximum number of machines running

• Automatically Register new instances to a load balancer

**AWS RDS**

--------

•RDS stands for Relational Database Service

• It’s a managed DB service for DB use SQL as a query language.

• It allows you to create databases in the cloud that are managed by AWS

• Postgres

• MySQL

• MariaDB

• Oracle

• Microsoft SQL Server

• Aurora (AWS Proprietary database)

**AWS Route 53**

---------------

• Route53 is a Managed DNS (Domain Name System)

• DNS is a set of rules and records which helps clients understand

how to reach a server through URLs. The name is a reference to TCP or UDP port **53**, where DNS server requests are addressed.

• In AWS, the most common records are:

• A: URL to IPv4

• AAAA: URL to IPv6

• CNAME: URL to URL

• Alias: URL to AWS resource.

**AWS Route 53 Overview**

----------------------

• Route53 can use:

• public domain names you own (or buy)

application1.mypublicdomain.com

• private domain names that can be resolved by our instances in our VPCs.

application1.company.internal

• Route53 has advanced features such as:

• Load balancing (through DNS – also called client load balancing)

• Health checks (although limited…)

• Routing policy: simple, failover, geolocation, latency, weighted, multi value

• You pay $0.50 per month per hosted zone

**AWS CloudWatch Metrics**

------------------------

• CloudWatch provides metrics for every services in AWS

• Metric is a variable to monitor (CPUUtilization, NetworkIn…)

• Metrics belong to namespaces

• Dimension is an attribute of a metric (instance id, environment, etc…).

• Up to 10 dimensions per metric

• Metrics have timestamps

• Can create CloudWatch dashboards of metrics

**AWS CloudWatch Logs**

--------------------

• Applications can send logs to CloudWatch using the SDK

• CloudWatch can collect log from:

• Elastic Beanstalk: collection of logs from application

• ECS: collection from containers

• AWS Lambda: collection from function logs

• VPC Flow Logs: VPC specific logs

• API Gateway

• CloudTrail based on filter

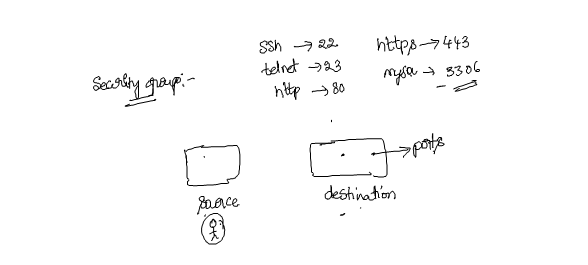
• CloudWatch log agents: for example on EC2 machines

• Route53: Log DNS queries

• CloudWatch Logs can go to:

• Batch exporter to S3 for archival

• Stream to ElasticSearch cluster for further analytics



**What is VPN in network?**

A **VPN**, or **Virtual Private Network**, allows you to create a secure connection to another **network** over the Internet. **VPNs** can be used to access region-restricted websites.

**VPCs and subnets**

A virtual private cloud (**VPC**) is a virtual network dedicated to our **AWS** account. It enables us to launch **AWS** resources into a virtual network that you've defined. This virtual network closely resembles a traditional network that we would operate in our own data center.

Default VPC Walkthrough

-----------------------

• All new accounts have a default VPC

• New instances are launched into default VPC if no subnet is specified

• Default VPC have internet connectivity and all instances have public IP

• We also get a public and a private DNS name

Understanding CIDR - IPv4

-------------------------------

(**Classless Inter-Domain Routing**)

• CIDR are used for Security Groups rules, or **AWS networking** in general

• They help to **define an IP address range**

• We’ve seen WW.XX.YY.ZZ/32 == one IP

• We’ve seen 0.0.0.0/0 == all IPs

• But we can define for ex: 192.168.0.0/26: 192.168.0.0 – 192.168.0.63 (64 IP)

Understanding CIDR

---------------------

• A CIDR has two components:

• The base IP (XX.XX.XX.XX)

• The Subnet Mask (/26)

• The base IP represents an IP contained in the range

• The subnet masks defines how many bits can change in the IP

• The subnet mask can take two forms. Examples:

• 255.255.255.0 --> less common

• /24 --> more common

Understanding CIDRs Subnet Masks

--------------------------------

• The subnet masks basically allows part of the underlying IP to get additional next values from the base IP

• /32 allows for 1 IP = 2^0

• /31 allows for 2 IP = 2^1

• /30 allows for 4 IP = 2^2

• /29 allows for 8 IP = 2^3

• /28 allows for 16 IP = 2^4

• /27 allows for 32 IP = 2^5

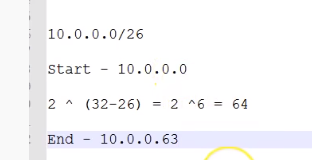
• /26 allows for 64 IP = 2^6

• /25 allows for 128 IP = 2^7

• /24 allows for 256 IP = 2^8

• /16 allows for 65,536 IP = 2^16

• /0 allows for all IPs = 2^32



**What is the difference between Amazon S3, Amazon EBS and Amazon EFS storage**

* **Amazon S3 (Simple storage services)**provides simple object storage, useful for hosting website images and videos, data analytics and all data types which are stored in their native formats. Data objects can be distributed across several machines. We can access the S3 service from anywhere on the internet.
* **AWS EBS (Elastic Block Store)**it act as separate hard drives; block storage devices are more flexible and offer higher performance than regular file storage. We need to mount EBS onto an Amazon EC2 instance.
* **AWS EFS(Elastic File System)**is a shared, elastic file storage system that grows and shrinks as we add and remove files. EFS is useful for SaaS applications and content management systems. We can mount EFS into several EC2 instances at the same time.