Amazon EC2

- EC2 is one of the most popular of AWS' offering
- EC2 = Elastic Compute Cloud = Infrastructure as a Service
- It mainly consists in the capability of:
    - Renting virtual machines (EC2)
    - Storing data on virtual drives (EBS)
    - Distributing load across machines (ELB)
    - Scaling the services using an auto-scaling group (ASG)
- Knowing EC2 is fundamental to understand how the Cloud works

EC2 sizing & configuration options

- Operating System (OS): Linux, Windows or Mac OS
- How much compute power & cores (CPU)
- How much random-access memory (RAM)
- How much storage space:
    - Network-attached (EBS & EFS)
    - Hardware (EC2 Instance Store)
- Network card: speed of the card, Public IP address
- Firewall rules: security group
- Bootstrap script (configure at first launch): EC2 User Data

EC2 User Data

- It is possible to bootstrap our instances using an EC2 User Data script.
- Bootstrapping means launching commands when a machine starts
- That script is only run once at the instance first start
- EC2 user data is used to automate boot tasks such as:
    - Installing updates
    - Installing software
    - Downloading common files from the internet
    - Anything you can think of
- The EC2 User Data Script runs with the root user

EC2 Instance Types - Overview

- You can use different types of EC2 instances that are optimised for different use cases (link)
- AWS has the following naming convention:

```
m5.2xlarge
```

    - `m` : instance class
    - `5` : generation (AWS improves them over time)
    - `2xlarge` : size within the instance class

EC2 Instance Types – General Purpose

- Great for a diversity of workloads such as web servers or code repositories
- Balance between:
  - Compute
  - Memory
  - Networking
- In the course, we will be using the `t2.micro`, which is a General Purpose EC2 instance

Examples of General Purpose instance families

Mac, T4g, T3, T3a, T2, M6g, M5, M5a, M5n, M5zn, M4, A1

EC2 Instance Types – Compute Optimized

- Great for compute-intensive tasks that require high performance processors:
  - Batch processing workloads
  - Media transcoding
  - High performance web servers
  - High performance computing (HPC)
  - Scientific modeling & machine learning
  - Dedicated gaming servers

Examples of Compute Optimized instance families

C6g, C6gn, C5, C5a, C5n, C4

EC2 Instance Types – Memory Optimized

- Fast performance for workloads that process large data sets in memory
- Use cases:
  - High performance, relational/non-relational databases
  - Distributed web scale cache stores
  - In-memory databases optimized for BI (business intelligence)
  - Applications performing real-time processing of big unstructured data

Examples of Memory Optimized instance families

R6g, R5, R5a, R5b, R5n, R4, X1e, X1, High Memory, z1d

EC2 Instance Types – Storage Optimized

- Great for storage-intensive tasks that require high, sequential read and write access to large data sets on local storage
- Use cases:
  - High frequency online transaction processing (OLTP) systems
  - Relational & NoSQL databases
  - Cache for in-memory databases (for example, Redis)
  - Data warehousing applications
  - Distributed file systems

Examples of Storage Optimized instance families

I3, I3en, D2, D3, D3en, H1

Introduction to Security Groups

- Security Groups are the fundamental of network security in AWS
- They control how traffic is allowed into or out of our EC2 Instances
- Security groups only contain allow rules
- Security groups rules can reference by IP or by security group

Security Groups – Deeper Dive

- Security groups are acting as a "firewall" on EC2 instances
- They regulate:
  - Access to Ports
  - Authorised IP ranges – IPv4 and IPv6
  - Control of inbound network (from other to the instance)
  - Control of outbound network (from the instance to other)

Security Groups – Good to know

- Can be attached to multiple instances
- Locked down to a region / VPC combination
- Does live "outside" the EC2 – if traffic is blocked the EC2 instance won't see it
- It's good to maintain one separate security group for SSH access
- If your application is not accessible (time out), then it's a security group issue
- If your application gives a "connection refused" error, then it's an application error or it's not launched
- All inbound traffic is blocked by default
- All outbound traffic is authorised by default

Classic Ports to know

- 22 = SSH (Secure Shell) – log into a Linux instance
- 21 = FTP (File Transfer Protocol) – upload files into a file share
- 22 = SFTP (Secure File Transfer Protocol) – upload files using SSH
- 80 = HTTP – access unsecured websites
- 443 = HTTPS – access secured websites
- 3389 = RDP (Remote Desktop Protocol) – log into a Windows instance

EC2 Instances Purchasing Options

- On-Demand Instances – short workload, predictable pricing, pay by second
- Reserved (1 & 3 years)
  - Reserved Instances – long workloads
  - Convertible Reserved Instances – long workloads with flexible instances
- Savings Plans (1 & 3 years) – commitment to an amount of usage, long workload
- Spot Instances – short workloads, cheap, can lose instances (less reliable)
- Dedicated Hosts – book an entire physical server, control instance placement
- Dedicated Instances – no other customers will share your hardware
- Capacity Reservations – reserve capacity in a specific AZ for any duration

EC2 On Demand

- Pay for what you use:
  - Linux or Windows – billing per second, after the first minute
  - All other operating systems – billing per hour
- Has the highest cost but no upfront payment
- No long-term commitment
- Recommended for **short-term** and **un-interrupted workloads**, where you can't predict how the application will behave

EC2 Reserved Instances

- Up to 72% discount compared to On-demand
- You reserve a specific instance attributes (Instance Type, Region, Tenancy, OS)
- Reservation Period – 1 year (+discount) or 3 years (+++discount)
- Payment Options – No Upfront (+), Partial Upfront (++), All Upfront (+++)
- Reserved Instance's Scope – Regional or Zonal (reserve capacity in an AZ)
- Recommended for steady-state usage applications (think database)
- You can buy and sell in the Reserved Instance Marketplace

Convertible Reserved Instance

- Can change the EC2 instance type, instance family, OS, scope and tenancy
- Up to 66% discount

EC2 Savings Plans

- Get a discount based on long-term usage (up to 72% – same as RIs)
- Commit to a certain type of usage ($10/hour for 1 or 3 years)
- Usage beyond EC2 Savings Plans is billed at the On-Demand price
- Locked to a specific instance family & AWS region (e.g., M5 in us-east-1)
- Flexible across:
  - Instance Size (e.g., m5.xlarge, m5.2xlarge)
  - OS (e.g., Linux, Windows)
  - Tenancy (Host, Dedicated, Default)

EC2 Spot Instances

- Can get a discount of up to 90% compared to On-demand
- Instances that you can "lose" at any point of time if your max price is less than the current spot price
- The MOST cost-efficient instances in AWS

Useful for workloads that are resilient to failure

- Batch jobs
- Data analysis
- Image processing
- Any **distributed** workloads
- Workloads with a flexible start and end time
- Not suitable for critical jobs or databases

EC2 Dedicated Hosts

- A physical server with EC2 instance capacity fully dedicated to your use
- Allows you to address **compliance requirements** and **use your existing server-bound software licenses** (per-socket, per-core, per-VM software licenses)

Purchasing Options

- **On-demand** – pay per second for active Dedicated Host
- **Reserved** – 1 or 3 years (No Upfront, Partial Upfront, All Upfront)
- The most expensive option

Useful for

- Software that has complicated licensing model (BYOL – Bring Your Own License)
- Companies that have strong regulatory or compliance needs

EC2 Dedicated Instances

- Instances run on hardware that's dedicated to you
- May share hardware with other instances in the same account
- No control over instance placement (can move hardware after Stop/Start)

EC2 Capacity Reservations

- Reserve On-Demand instances capacity in a specific AZ for any duration
- You always have access to EC2 capacity when you need it
- No time commitment (create/cancel anytime), no billing discounts
- Combine with Regional Reserved Instances and Savings Plans to benefit from billing discounts
- You're charged at On-Demand rate whether you run instances or not
- Suitable for short-term, uninterrupted workloads that need to be in a specific AZ

Which purchasing option is right for me?

- **On demand**: coming and staying in resort whenever we like, we pay the full price
- **Reserved**: like planning ahead and if we plan to stay for a long time, we may get a good discount
- **Savings Plans**: pay a certain amount per hour for certain period and stay in any room type (e.g., King, Suite, Sea View, ...)
- **Spot instances**: the hotel allows people to bid for the empty rooms and the highest bidder keeps the rooms. You can get kicked out at any time
- **Dedicated Hosts**: We book an entire building of the resort
- **Capacity Reservations**: you book a room for a period with full price even you don't stay in it

AWS charges for IPv4 addresses

- Starting February 1st 2024, there's a charge for all Public IPv4 created in your account
- $0.005 per hour of Public IPv4 (~ $3.6 per month)
- For new accounts in AWS, you have a free tier for the EC2 service: 750 hours of Public IPv4 per month for the first 12 months
- For all other services there is no free tier

What about IPv6?

- Unfortunately, many Internet Service Provider (ISP) around the world don't support IPv6, so the course would not work for some of you
- You can test IPv6 by going to https://test-ipv6.com/
- If you use IPv6 in this course, you're on your own (security groups, networking...) but you can do it!

How to troubleshoot charges?

- Go into your AWS Bill
- Look into the AWS Public IP Insights service

EC2 Spot Instance Requests

- Can get a discount of up to 90% compared to On-demand
- Define **max spot price** and get the instance while **current spot price < max**
    - The hourly spot price varies based on offer and capacity
    - If the current spot price > your max price you can choose to **stop** or **terminate** your instance with a 2 minutes grace period
- Other strategy: **Spot Block**
    - "Block" spot instance during a specified time frame (1 to 6 hours) without interruptions
    - In rare situations, the instance may be reclaimed
- Used for batch jobs, data analysis, or workloads that are resilient to failures
- Not great for critical jobs or databases

How to terminate Spot Instances?

- Spot request parameters:
    - Maximum price
    - Desired number of instances
    - Launch specification
    - Request type: one-time | persistent
    - Valid from, Valid until
- Spot request lifecycle:
    - **Create request** → request becomes **open**
    - **Open** → can go to **active**, **failed**, or **cancelled**
    - **Active** →
        - For **persistent** requests: can become **disabled**
        - For **one-time** requests: can become **closed**
    - **Cancelled** → end state
- **You can only cancel Spot Instance requests that are open, active, or disabled**
- **Cancelling a Spot Request does not terminate instances**
- You must **first cancel** the Spot Request, **then terminate** the associated Spot Instances

Reference: AWS Documentation

Spot Fleets

- **Spot Fleets** = set of Spot Instances + (optional) On-Demand Instances
- The Spot Fleet will try to meet the target capacity with price constraints:
  - Define possible launch pools: instance type (e.g., m5.large), OS, Availability Zone
  - Can have multiple launch pools, so that the fleet can choose
  - Spot Fleet stops launching instances when reaching capacity or max cost
- **Strategies to allocate Spot Instances:**
  - **lowestPrice**: from the pool with the lowest price (cost optimization, short workload)
  - **diversified**: distributed across all pools (great for availability, long workloads)
  - **capacityOptimized**: pool with the optimal capacity for the number of instances
  - **priceCapacityOptimized (recommended)**: pools with highest capacity available, then select the pool with the lowest price (best choice for most workloads)
- **Spot Fleets allow us to automatically request Spot Instances with the lowest price**