EC2 Fundamentals

AWS Budget Setup

- To activate billing information: Root User > Billing dashboard > IAM User and Role Access to Billing Information > Edit > checkbox Activate.
- See attached

Amazon EC2

- A popular AWS offering
- Elastic Compute Cloud = Infrastructure as a Service
  - Consists of:
    - Rent Virtual Machines (EC2)
    - Store data on virtual drives (EBS)
    - Distribute load across machines (ELB)
    - Scale 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

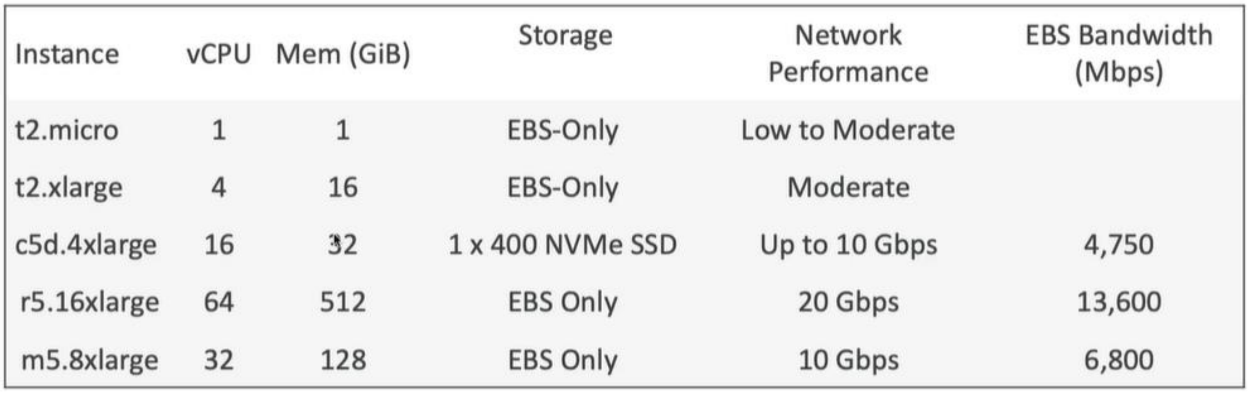
EC2 User Data:

- Bootstrap our instances using an EC2 User data script
- Bootstrapping means launching command when a machine starts
- The script is **one 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 internet
- ANYTHING

** The EC2 User Data Scripts runs with the root user

EC2 Instance Types: Five Examples



Instance	vCPU	Mem (GiB)	Storage	Network Performance	EBS Bandwidth (Mbps)
t2.micro	1	1	EBS-Only	Low to Moderate	
t2.xlarge	4	16	EBS-Only	Moderate	
c5d.4xlarge	16	32	1 x 400 NVMe SSD	Up to 10 Gbps	4,750
r5.16xlarge	64	512	EBS Only	20 Gbps	13,600
m5.8xlarge	32	128	EBS Only	10 Gbps	6,800

EC2 > Instance > Launch Instances

- Step 1: Choose AMI > (64-bit)
- Step 2: Choose an Instance Type > Free tier eligible
- Step 3: Configure Instance > Number of instances: 1 > copy script

Copy + paste script into "optional box"

```
#!/bin/bash

# Use this for your user data (script from top to bottom)

# install httpd (Linux 2 version)

yum update -y

yum install -y httpd

systemctl start httpd

systemctl enable httpd

echo "<h1>Hello World from $(hostname -f)</h1>"> /var/www/html/index.html
```

- Step 4: Add Storage: check size, check delete on termination, etc.
- Step 5: Add Tags: Add name > Name, My First Instance, Department, Finance
- Step 6: Configure Security Group: Add SHH & HTTPS
- Step 7: Review: Create a new key pair > RSA (key pair type), Key Pair name: EC2 Tutorial

Once RC2 is created, there are many options and visuals to see when it is running in the cloud. It is VERY normal to terminate the instance when working it. Right click on selected EC2 name and select terminate. In the cloud, everything is disposal.

EC2 Instance Types Basics

1. General Purpose
2. Compute Optimized
3. Memory Optimized
4. Accelerated Computing
5. Storage Optimized
6. Instance Features
7. Measuring Instances Performance

Naming convention

m5.2xlarge

M = instance class

5 = generation (AWS improves them over time)

2xlarge = size within the instance class

General Purpose

Great for diversity of workloads such as web servers or code repositories

- Balance between:
  - Compute
  - Memory
  - Networking
- *Example: t2.micro which is a General Purpose EC2 instance*

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 and machine learning
- Dedicated gaming servers

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

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) system
  - Relational & NoSQL databases
  - Cache for in-memory databases (for example, Redis)
  - Data warehousing applications
  - Distributed file systems

Here’s a list of all the available instances:
[Amazon EC2 Instance Comparison (vantage.sh)]

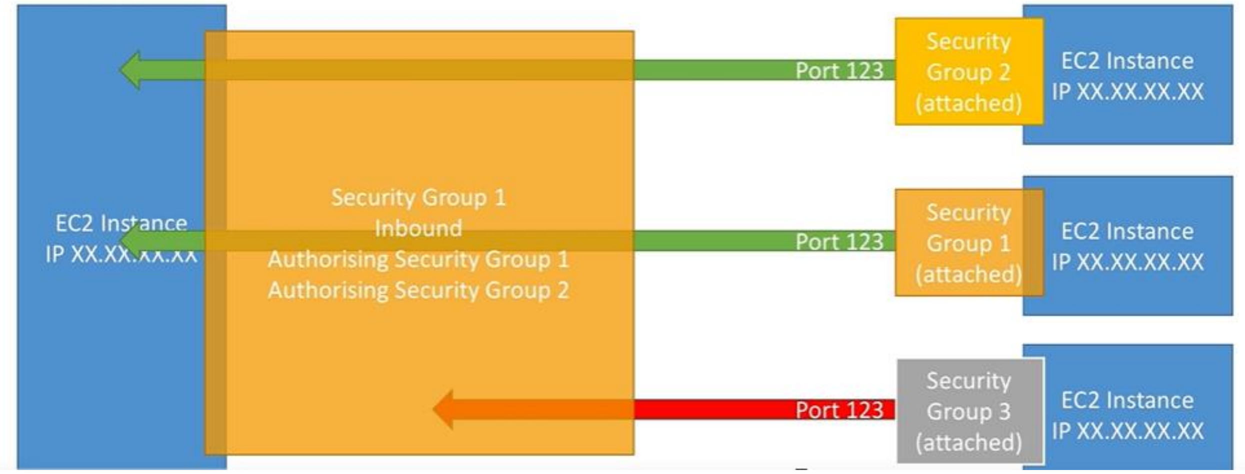
Security Groups & Classic Ports

- Security groups are fundamentals of network security in AWS
- 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

Deeper Dive of Security Groups

- Security groups are acting as a “firewall” on EC2 instances

- They regulate:
    - Access to Ports
    - Authorized IP ranges – Ipv4 and IPv6
    - Control of inbound network (from other to the instance)
    - Control of outbound network (from the instance to other)
- 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 SHH 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 its an application error or it’s not launched
- All inbound traffic is **blocked** by default
- All outbound traffic is **authorized** 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

How to get into the server of SSH computer server:

- ~/aws-course ssh ec2-user@*ipaddress 01.234.567.890*

- Ls
- Ssh -I EC2Tutorial.pem ec2-user@01.234.567.890
- (Will provide warnings)
- Chmod 0400 EC2Tutorial.pem
- Ssh -I EC2Tutorial.pem ec2-user@01.234.567.890
- “Amazon Linux 2 AMI”

EC2 Instance Connect via AWS
EC2 > Instances > Select > Connect > AWS instance screen