**EC2**

**Sections**

1. EC2 Pricing Types
2. EC2 Instance Types
3. Security Group
4. EBS and EBS Types
5. EBS Volumes & Snapshot
6. AMI Types (EBS vs Instance Store)
7. ENI vs ENA vs EFA
8. Encrypt Root Device Volume and Snapshot
9. Spot Instance and Spot Fleet
10. EC2 Hibernate
11. CloudWatch vs. CloudTrail
12. AWS Command Line (CLI)
13. IAM Roles Demo on EC2
14. Using Bootstrap Scripts (User Data)
15. Instance Metadata – 169.254.169.254
16. EFS
17. Amazon FSx for Windows and Amazon FSx for Lustre
18. EC2 Placement Groups
19. HPC (High Performance Computing) on AWS
20. AWS WAF

**EC2 - 101**

1. What is EC2 – Virtual Machines, minutes to provision servers, can be bare metal for special needs
2. EC2 Pricing Types
   1. **On Demand** – **No commitment** Changed by **Hour** or **Second** (**Fix Rate**) (**partial instance-hour consumed** will be billed per-second for Linux, Windows, Windows with SQL Enterprise, Windows with SQL Standard, and Windows with SQL Web Instances, and as a full hour for all other instance types) - <https://aws.amazon.com/ec2/pricing/on-demand/>
   2. **Reserved** – Reserve capacity with **1 Year** or **3 Year** commitment for **Discount** on the **Hourly** charge
      1. **Standard Reserved Instances** – Up to **75%** off. **Pay up front** and **longer contract** get more discount
      2. **Convertible Reserved Instances** – Up to **54%** off. Allowed to change instance type to **Equal or Greater value (Change UP)**
      3. **Scheduled Reserved Instances** – Allows to schedule on a recurring schedule like **a fraction of a day, a week, or a month**
   3. **Spot** – Leftover capacity from AWS for **cheaper price**. Can be **Terminated** by AWS if needed. You **BID** on the price
      1. If terminated by AWS, no partial hour charge
      2. If you terminate it yourself, you will be charged for partial hour
      3. Need to have flexible start and end times to use this option to save money
   4. **Dedicated Hosts** – Physical server dedicated to use. For **Compliance** (No multi-tenant allowed) or special **License** requirement. It can be reserved as well for better price
      1. **On Demand** – (**Hourly**)
      2. **Reserved** – Up to **70%** Off from On-demand
   5. **Dedicated Instances** vs. **Dedicated Hosts**
      1. **Dedicated Instances** are Amazon EC2 instances that run in a VPC on hardware that's dedicated to a single customer. **Your Dedicated instances are physically isolated at the host hardware level from instances that belong to other AWS accounts**. Dedicated instances may share hardware with other instances from the same AWS account that are not Dedicated instances. Pay for Dedicated Instances On-Demand, save up to 70% by purchasing Reserved Instances, or save up to 90% by purchasing Spot Instances
      2. You can also use **Dedicated Hosts** to launch Amazon EC2 instances on physical servers that are dedicated for your use. **Dedicated Hosts give you additional visibility and control over how instances are placed on a physical server**, and you can reliably use the same physical server over time. As a result, Dedicated Hosts enable you to use your existing server-bound **Special** **software licenses** like Windows Server and address corporate compliance and regulatory requirements. Visit this page to compare Dedicated Instances and Dedicated Hosts.
3. EC2 Instance Types
   1. FIGHTDRMCPXAU – with T general purpose like (Like T2 Micro)

**Launch 1st EC2 Instance**

1. **Termination Protection** is **turned off** by default. You can turn it on if you want to
2. **Default** action is to **delete the Root EBS volume** when the instance is terminated. You can uncheck it so it is not deleted.
3. **Default** action on other EBD volumes are not deleted (You can check it so they are deleted)
4. EBS **Root volume** can be **Encrypted.** You can also use 3rd party tool to make the encryption
5. Additional volumes can be **Encrypted**
6. **Security Group** – VPC specific
7. **Key Pair** – Region specific (You can create your own and upload it to all the regions, if you want to use the same Key pair for all the regions)

**Security Group**

1. All inbound traffic is blocked by default (No rule for inbound by default).
2. All outbound traffic is allowed (There is a default rule ALL-ALL, you can delete it if you want to)
3. Security Group is **STATEFUL** – If you create an inbound rule allowing traffic in, the **response traffic** is automatically allowed to **BACK-OUT** again.
4. Changes to Security Groups takes effect immediately
5. Multiple Security Groups can attach to EC2 instances
6. You can not block specific IP addresses by Security Group. You need to use NACL (Network Access Lists)
7. There is NO Deny rules (Explicit deny)

**EBS 101**

1. EBS – **Elastic Block Store** (Virtual Hard Drive) – automatically replicated in its Availability Zone, offering high availability and durability
2. **5 Different Types**
   1. General Purpose (SSD) – **gb2**
   2. Provisioned IOPS (SSD) – **Highest Performance** – **io1**
   3. Throughput optimized Hard Disk Drive – **st1**
   4. Cold Hard Disk Drive – **sc1**
   5. Magnetic – Previous Generation - **Standard**

Table

Description automatically generated

**EBS Volumes & Snapshot**

1. Volumes exist on EBS => EBS as Virtual Hard Drive
2. Snapshots exist on **S3**
3. Snapshots are point in time copies of Volumes (**You can take snapshot while the instance is running**) - Snapshots only capture data that has been written to your Amazon EBS volume **at the time the snapshot command is issued**
4. Snapshots are **incremental** – Only the blocks changed since the last snapshot are saved to S3
5. **First snapshot** takes some time to create
6. Best practice is to **stop** the instance for the **ROOT** volume before taking the snapshot
7. You can create AMI from snapshots
8. You can change EBS volume (**Both size and storage type**) on the fly
9. Volumes will ALWAYS in the same **AZ** as the EC2 instance
10. Move EC2 to another AZ => Snapshot => AMI => Create EC2 in a new AZ
11. Move EC2 to another region => Snapshot => AMI => Copy AMI to another region => Launch new instance

We can copy both Snapshots and AMIs to another region

* <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-copy-snapshot.html>
* <https://aws.amazon.com/premiumsupport/knowledge-center/copy-ami-region/>

**AMI Types (EBS vs Instance Store)**

An Amazon Machine Image (AMI) provides the information required to launch an instance. You must specify an AMI when you launch an instance. You can launch multiple instances from a single AMI when you need multiple instances with the same configuration. You can use different AMIs to launch instances when you need instances with different configurations

1. Instance Store Volumes are called **Ephemeral Storage**
2. You can attach additional Instance Store volumes before the instance launches, but after that you can only attach EBS volumes
3. Instance store volumes cannot be stopped. If the underlying host fails, you lose your data
4. EBS backed instances can be stopped, you will not lose the data on the instance if stopped
5. You can **REBOOT** both without losing the data
6. By default, both ROOT volumes will be deleted on termination. But you can configure to keep the EBS root volume when instance terminated

Why still use Instance Store Volumes – **May be faster**

1. **Instance stores** still have value especially when it comes to **massive IOPS at low latency**. The size of the Instance Store Volume depends on the type and cannot be changed freely. The data in the instance store is **LOST** under any of the following circumstances:
   1. The underlying disk drive fails
   2. The instance stops
   3. The instance hibernates
   4. The instance terminates
2. Until you get to the rarified atmosphere of high-performance compute, **EBS storage provides plenty of grunt and a whole bunch of flexibility to meet most of your EC2 needs**

**ENI vs ENA vs EFA**

1. **ENI** – **Elastic Network Interface**
   1. Essentially a **Virtual Network Card**
   2. An ENI is automatically attached to the EC2 instance automatically – It allows IP addresses (Private/public/IPV4/IPV6/Security Groups/MAC/Source-Destination-Check/Description)
   3. Multiple ENI can be attached to an EC2 instance
2. **EN** – **Enhanced Networking** – Used to **High Performance Networking** – Single Root I/O Virtualization (**SR-IOV**) for **High Performance**
   1. Use for High Network Performance
   2. **No additional charge**, but the EC2 instance needs to support it
   3. **ENA** (**Elastic Network Adapter**) (Up to **100Gbps**) vs **Intel** **82599 Virtual Function** (**VF**) (Up to **10Gbps**) [Typically used in **OLD** instances] – Always choose **ENA**. It is designed for the purpose
   4. **Adding more ENI NOT Necessarily** speed up network throughput. Need **ENA**
3. **EFA** – **Elastic Fabric Adapter**
   1. A network device attached to EC2 instance to accelerate **High Performance Computing** (**HPC**) and **Machine Learning**. **OS-Bypass** – Bypass OS Kernel for performance (Only available for **LINUX**)
   2. **EFA** – It is basically **ENA + OS Bypass**. On Windows, it performs as **ENA**
   3. EFA is available as an optional EC2 networking feature that you can enable on any supported EC2 instance at **no additional cost**

**Encrypt Root Device Volume and Snapshot**

1. **Root Volume** – Volume with the OS
2. **You can choose to encrypt ROOT Volume when you provision the EC2**, but if you did not, you need to go through the steps
3. Steps
   1. Create a **Snapshot**
   2. **Copy the snapshot and encrypt**, there is an option to encrypt the new snapshot (Give a description to the new snapshot and use default key – aws/ebs [May have other key options])
   3. **Create an AMI** on the encrypted snapshot
   4. **Launch a new EC2 instance** on the encrypted AMI
4. You **CANNOT** create an **UN-ENCRYPTED volume** from an **Encrypted AMI**
5. Volumes restored from Encrypted snapshot are automatically encrypted
6. You can **share** snapshot only if they are **NOT encrypted**

**Spot Instance and Spot Fleet**

1. Spot instances can get up to **90%** discount. If you application is flexible for termination (**A Spot Instance interruption notice** is a warning that is issued **two minutes** before Amazon EC2 stops or terminates your Spot Instance) - <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/spot-interruptions.html>
2. Spot Price – You set you MAX-SPOT Price
3. Spot Block – Stop instance termination even the SPOT PRICE above your MAX (Between 1 – 6 Hours currently)
4. SPOT NOT Good for – Persistent Workload / Critical Jobs / Databases
5. **SPOT FLEET** – **Combination** of SPOT and ON-DEMAND instances for desired capacity
   1. Set up different **launch pools**. Define things like EC2 instance type, OS, and AZ
   2. You can have **multiple** pools, and the fleet will choose best way to implement depending on the strategy defind
   3. Spot fleet will **STOP LAUNCHING** instances once the **Price Threshold** or **capacity requirement** is met
6. **SPOT fleet strategies**
   1. **CapacityOptimized** – Optimize Capacity for the number of instances launching
   2. **LowerestPrice** – **DEFAULT** **Strategy** – choose the lowest price
   3. **Diversified** – **Distribute** across all pools
   4. **InstancePoolsToUseCount** – The spot instances are distributed across the number of spot instance pools you specify. The parameter is valid only when used in combination with **LowestPrice**

**TIPS**

1. SPOT instances (BID) can save up to 90% from ON-DEMAND
2. USEful for type of computing that can be terminated
3. You can **BLOCK SPOT** instances from terminating SPOT BLOCK (**1-6 hours** only currently)
4. SPOT FLEET is a **collection** of SPOT instances and **optionally** **ON-DEMAND** instances

**EC2 Hibernate**

1. When launching an instance, you have the option to **check** the “**ENABLE Hibernation**” option (It is a **checkbox** on the AWS console)
2. You need to **encrypt** the **ROOT Volume** and you need to make sure the ROOT Volume is **large enough** to save the content in the RAM
3. Hibernate is NOT Re-start the instance
4. EC2 Hibernate preserves RAM on the ROOT EBS Volume
5. It is much **fast** that Regular OS Boot-up to make available from Hibernating
   1. Instance RAM needs to be **less than 150 GB**
   2. Only available on certain instance types
   3. Certain OSs
   4. Cannot hibernate more than **60 days**
   5. For **ON-DEMAND** and **Reserved Instances**

**CloudWatch 101 (Important for SAA) CloudWatch vs. CloudTrail**

1. **CloudWatch** tool can monitor **almost everything** (Mostly **AWS Performance and health**). **CloudWatch** monitor **EC2** for every **5 minutes** by DEFULT (Can be adjusted – **Detailed Monitoring** – **1 minute** interval)
2. **CloudWatch Alarms** can be turned on to Trigger **Notifications**
3. **CloudWatch** -> Dashboard / Alarms / Logs / Metrics / Events / Application Monitoring / Insights
4. **CloudWatch** – Logs are Region specific and Billing Alarms is in US-EAST-1
5. **CloudWatch** or **CloudTrail**
   1. **CloudWath** is **monitoring** tool
   2. **CloudTrail** – CloudTrail to meet your **governance**, **compliance**, and **operational** **auditing** and **risk auditing** needs for your AWS accounts, **Continuously log** your AWS account activities, either through console, API (CLI or other client), identify user account, IP, which call and the time when it is made.
   3. **CloudTrail** – Logs API calls in the AWS platform (Who set up S3? Who create an EC2 instance?)
6. **CONFUSING part is** – **Logs/Log Groups** are under the **CloudWatch**

**CloudWatch Demo**

1. EC2 Standard Monitoring – 5 Minutes
2. EC2 Detailed Monitoring – 1 Minute
3. There are other monitoring intervals for other things
4. **Dashboard** - You can use CloudWatch to create Dashboard to see what happening in your AWS Account
5. **Alarms** – Allow you to set Alarms to notify you when a threshold is reached
6. **Events** – Allow you to respond to state changes in your AWS resources
7. **Logs** – CloudWatch logs helps you to aggregate, monitor, and store logs

**AWS Command Line (CLI) Demo**

1. You can inter-act with AWS from anywhere using AWS CLI (Almost everything)
2. You need program access and the IAM permissions
3. Key / Secret-access-key and “.aws” directory

**IAM Roles Demo**

1. Roles are **more secure** than storing access key and secret access key on individual EC2 instances
2. Roles are easier to manage
3. Roles can be assigned to an EC2 instance after it is created using both console or CLI
4. **Roles are Universal** – You can use them in any region
5. **ONLY 1 role** – Per EC2 instance

**Using Bootstrap Scripts (User Data) Demo**

1. Run commands on your Linux instance **at Launch (First BOOT)** – There are many ways to run a script at **Every Restart** on **the Operating System level**
2. When configure EC2 instance => Advanced Details => **User Data**

**Instance Metadata - Lab**

1. Metadata is used to GET information about the Instance (such as Public IP)
2. curl <http://169.254.169.254/latest/meta-data>
3. curl <http://169.254.169.254/latest/user-data> - It is the bootstrap script

**EFS - Lab**

1. **EFS** – **Elastic File System**. Easy to share file systems among EC2 instances (Mont to the file system path in EC2 instances)
2. EFS needs to provision
   1. **No Need to specify size**
   2. Choose **VPC** and collection of **Subnets** (So EC2 instances in the **subnets** can access the EFS System)
   3. Choose **Security Group** for each subnet, so the permissions to access the EFS is controlled (Granted)
   4. Other options – Life cycle management, Throughput mode, Performance mode, Encryption, etc.
3. **EFS Security Group** needs allow **NFS inbound rule** (allow the EC2 to connect)**,** so it is accessible from the EC2 instances
4. Use “**amazon-efsutils**” [Help web link available on the AWS console on the EFS page] command-line utility to **MOUNT** (TLS option available) EFS to the **local file system** on the EC2 instances. Then use it as a directory (**sudo** may be needed)
5. EFS can be **shared** by **multiple** EC2 instances
6. EFS Suppots **Network File System Version 4** (**NFSv4**) protocol
7. **Pay** for the storage you use
8. Scale up to Petabytes
9. Support thousands of concurrent NFS connections
10. Data is **stored Multi-AZ** within a region
11. **Read After Write** consistency
12. **EFS is LINUX only**

**Amazon FSx for Windows and Amazon FSx for Lustre**

1. Amazon **FSx** is a **Windows File Server**, Designed to work with **Windows based applications**, SQL Server, Active Directory, IIS, etc.
2. **FSx** runs **Windows Server Message Block** (**SMB**) based file services
3. **Amazon FSx for Lustre** is a FULLY Managed **file system** that is optimized for **compute-intensive workloads**, such as high performance computing, machine learning, media data processing, and electronic design automation (EDA)
4. With Amazon FSx, you can launch and run a **Lustre** **file system** that can process massive data set at up to hundreds of gigbytes per second of throughput, millions of IOPS, and sub-millisecond latencies
5. **FSx for Lustre** can store data directly to **S3**

**EC2 Placement Groups**

1. **Clustered** – Group instances within a **single AZ**. Recommended for applications need **low network latency**, **high network throughput**, or both. Only certain instances can be launched in a **Clustered Placement Group**
2. **Spread** – Instances are each placed **on distinct RACKs**, with each rack having its own **network** and **power source**. Recommended for applications that have a small number of **CRITICAL Instances** that should be kept **separate** from each other
   1. A spread placement group **CAN** span **multiple Availability Zones in the same Region**
   2. You can have a maximum of **seven** running **instances** per Availability Zone per group)
3. **Partitioned** – Amazon divides each group into different partitions. Amazon EC2 ensures that **each partition** within a placement group has its **own SET of racks**. Each rack has its own network and power source. No two partitions within a placement group share the same racks, allowing you to isolate the impact of hardware failure within your application
   1. A partition placement group can have partitions in **multiple Availability Zones** **in the same Region**.
   2. A partition placement group can have a maximum of **seven partitions** per Availability Zone.
   3. The number of instances that can be launched into a partition placement group is limited only by the limits of your account

TIP

1. Clustered – Low Network Latency & High Network Throughput
2. Spread – **Individual** **Critical** EC2 instances
3. Partitioned – Multiple EC2 instances **HDFS**, **HBase** and **Cassandra** partitions
4. A clustered group does not span multiple AZ
5. Spread and Partitioned groups span multiple AZ in the same region
6. The name of the placement group is unique in the AWS account
7. Only certain types of instances can be launched in placement groups (Compute optimized, GPU, Memory Optimized, Storage Optimized)
8. AWS recommend homogenous instances placed in clustered placement groups
9. You cannot merge placement groups
10. You can move existing EC2 into a placement group, but it must be stopped first. You can do it with CLI and SDK, but not in the AWS console yet

**HPC (High Performance Computing) on AWS**

1. Data Transfer
   1. Snowball, Snowmobile
   2. AWS Datasync to s3, EFS, FSx for Windows, etc,
   3. Storage Gateway - Integration
   4. Direct Connect – A dedicated network connection to AWS from premises to AWS
2. Compute & Networking
   1. EC2 GPU or CPU Optimized
   2. EC2 Fleet (Spot instances or spot fleet)
   3. Placement Groups (Cluster placement group)
   4. Enhanced Networking
   5. Elastic Network Adapter – (SR-IOV)
   6. Elastic Fabric Adapter – (ENA + OS Bypass) – Support HPC and Machine Learning Applications. Linux only
3. Storage
   1. Instance Attached
      1. EBS – 64000 IOPS with provisioned IOPs
      2. Instance Store – Scale to millions of IOPs, low latency
   2. Network Attached
      1. Amazon S3
      2. EFS – Scaled IOPs on total size, or use provisioned IOPs - Linux
      3. Amazon FSx – For Windows and Linux - Server Message Block (SMB)
      4. FSx for Lustre – HPC-optimized. Millions of IOPs, backed by S3
4. Orchestration & Automation
   1. AWS Batch
      1. Enables to run hundreds of thousands of batch computing jobs on AWS
      2. Support multi-node parallel jobs, allow a single job to span multiple instances
      3. Easily schedule job and launch EC2 instances according to need
   2. AWS ParallelCluster
      1. Open0source cluster management tool. Make it easy to deploy and manage HPC clusters on AWS
      2. It uses simple text file to model and provision all the resources needed for HPC applications in an automated and secure manner
      3. Automate creation of VPC, subnet, cluster type and instance types

**AWS WAF**

1. WAF – **Web Application Firewall**. It monitors the HTTP/HTTPS requests to **Cloudfront**, Application Load Balancer (**ALB**) or **API Gateway**. AWF allows control access to the content. HTTP 403 is returned (Forbidden if not allowed)
2. Behaviors
   1. Allow all except the ones specified
   2. Block all expect the ones specified
   3. Passive mode – Count the request that match the properties specified
3. Layer 7 firewall – Can see query string (**7 Kinds** of checks)
   1. IP address that requests originate from
   2. Country
   3. **Values in the header**
   4. **String** that appears in the request, string or regex pattern
   5. **Length** of requests
   6. Presence of SQL that likely be malicious (SQL injection)
   7. Presence of script that likely be malicious (Cross-site scripting)

Text

Description automatically generated

**LAB – Using WAS Tags and Resource Groups**

No

**LAB – Create and Work with an EC2 Instance in AWS**

No

**Using EC2 Roles and Instance Profiles in AWS**

No