MODULE 3

Amazon Elastic Compute Cloud(EC2) and Elastic Block Store(EBS)

Amazon Elastic Compute Cloud(EC2) and Elastic Block Store(EBS)
Overview of AWS EC2
EC2 Security groups
Amazon Machine Images
EBS Volume Basics and Snapshot

This module is intended to learning how Amazon Elastic Compute Cloud (Amazon EC2) and Amazon Elastic Block Store (Amazon EBS) provide the basic elements of compute and block-level storage to run your workloads on AWS.

Amazon EC2 provides resizable compute capacity in the cloud. Compute refers to the amount of computational power required to fulfill your workload. If your workload is very small, such as a website that receives few visitors, then your compute needs are very small. A large workload, such as screening ten million compounds against a common cancer target, might require a great deal of compute. The amount of compute you need might change drastically over time.

Amazon EC2 allows you to acquire compute through the launching of virtual servers called instances. When you launch an instance, you can make use of the compute as you wish, just as you would with an on-premises server. Because you are paying for the computing power of the instance, you are charged per hour while the instance is running. When you stop the instance, you are no longer charged.

There are two concepts that are key to launching instances on AWS: (1) the amount of virtual hardware dedicated to the instance and (2) the software loaded on the instance. These two dimensions of new instances are controlled, respectively, by the instance type and the AMI.

Instance Types

The instance type defines the virtual hardware supporting an Amazon EC2 instance. There are dozens of instance types available, varying in the following dimensions:

- Virtual CPUs (vCPUs)
- Memory
- Storage (size and type)
- Network performance

Instance types are grouped into families based on the ratio of these values to each other. For instance, the m4 family provides a balance of compute, memory, and network resources, and it is a good choice for many applications. Within each family there are several choices that scale up linearly in size. Figure 3.1 shows the four instance sizes in the m4 family. Note that the ratio of vCPUs to memory is constant as the sizes scale linearly. The hourly price for each size scales linearly as well. For example, an m4.xlarge instance costs twice as much as the m4.large instance.

Instance Family	Current Generation Instance Types
General purpose	t2.nano | t2.micro | t2.small | t2.medium | t2.large | t2.xlarge | t2.2xlarge | m4.large | m4.xlarge | m4.2xlarge | m4.4xlarge | m4.10xlarge | m4.16xlarge | m3.medium | m3.large | m3.xlarge | m3.2xlarge
Compute optimized	c4.large | c4.xlarge | c4.2xlarge | c4.4xlarge | c4.8xlarge | c3.large | c3.xlarge | c3.2xlarge | c3.4xlarge | c3.8xlarge
Memory optimized	r3.large | r3.xlarge | r3.2xlarge | r3.4xlarge | r3.8xlarge | r4.large | r4.xlarge | r4.2xlarge | r4.4xlarge | r4.8xlarge | r4.16xlarge | x1.16xlarge | x1.32xlarge
Storage optimized	i2.xlarge | i2.2xlarge | i2.4xlarge | i2.8xlarge | d2.xlarge | d2.2xlarge | d2.4xlarge | d2.8xlarge
Accelerated computing	p2.xlarge | p2.8xlarge | p2.16xlarge | g2.2xlarge | g2.8xlarge

Figure 3.1

In response to customer demand and to take advantage of new processor technology, AWS occasionally introduces new instance families. Check the AWS website for the current list. Another variable to consider when choosing an instance type is network performance. For most instance types, AWS publishes a relative measure of network performance: low, moderate, or high. Some instance types specify a network performance of 10 Gbps. The network performance increases within a family as the instance type grows.

For workloads requiring greater network performance, many instance types support enhanced networking. Enhanced networking reduces the impact of virtualization on network performance by enabling a capability called Single Root I/O Virtualization (SR-IOV). This results in more Packets Per Second (PPS), lower latency, and less jitter. At the time of this writing, there are instance types that support enhanced networking in the C3, C4, D2, I2, M4 and R3 families (consult the AWS documentation for a current list). Enabling enhanced networking on an instance involves ensuring the correct drivers are installed and modifying an instance attribute. Enhanced networking is available only for instances launched in an Amazon Virtual Private Cloud (Amazon VPC).
Once launched, instances can be managed over the Internet. AWS has several services and features to ensure that this management can be done simply and securely.

Addressing an Instance
There are several ways that an instance may be addressed over the web upon creation:

Public Domain Name System (DNS) Name—When you launch an instance, AWS creates a DNS name that can be used to access the instance. This DNS name is generated automatically and cannot be specified by the customer. The name can be found in the Description tab of the AWS Management Console or via the Command Line Interface (CLI) or Application Programming Interface (API). This DNS name persists only while the instance is running and cannot be transferred to another instance.

Public IP—A launched instance may also have a public IP address assigned. This IP address is assigned from the addresses reserved by AWS and cannot be specified. This IP address is unique on the Internet, persists only while the instance is running, and cannot be transferred to another instance.

Elastic IP—An elastic IP address is an address unique on the Internet that you reserve independently and associate with an Amazon EC2 instance. While similar to a public IP, there are some key differences. This IP address persists until the customer releases it and is not tied to the lifetime or state of an individual instance. Because it can be transferred to a replacement instance in the event of an instance failure, it is a public address that can be shared externally without coupling clients to a particular instance.

Initial Access

Amazon EC2 uses public-key cryptography to encrypt and decrypt login information. Public-key cryptography uses a public key to encrypt a piece of data and an associated private key to decrypt the data. These two keys together are called a key pair. Key pairs can be created through the AWS Management Console, CLI, or API, or customers can upload their own key pairs. AWS stores the public key, and the private key is kept by the customer. The private key is essential to acquiring secure access to an instance for the first time.

Virtual Firewall Protection

AWS allows you to control traffic in and out of your instances through virtual firewalls called ***security groups***. Security groups allow you to control traffic based on port, protocol, and source/destination. Security groups have different capabilities depending on whether they are associated with an Amazon VPC or Amazon EC2-Classic. Table 3.2 compares these different capabilities (Amazon VPC is discussed in Module 4).

TABLE 3.2 Different Security Groups

Type of Security Group	**Capabilities**
EC2-Classic Security Groups	Control outgoing instance traffic
VPC Security Groups	Control outgoing and incoming instance traffic

Security groups are associated with instances when they are launched. Every instance must have at least one security group but can have more.

A security group is default deny; that is, it does not allow any traffic that is not explicitly allowed by a security group rule. A rule is defined by the three attributes. When an instance is associated with multiple security groups, the rules are aggregated and all traffic allowed by each of the individual groups is allowed. For example, if security group A allows RDP traffic from 72.58.0.0/16 and security group B allows HTTP and HTTPS traffic from 0.0.0.0/0 and your instance is associated with both groups, then both the RDP and HTTP/S traffic will be allowed in to your instance.

A security group is a ***stateful firewall***; that is, an outgoing message is remembered so that the response is allowed through the security group without an explicit inbound rule being required.

Launching
There are several additional services that are useful when launching new Amazon EC2 instances.

A great benefit of the cloud is the ability to script virtual hardware management in a manner that is not possible with on-premises hardware. In order to realize the value of this, there has to be some way to configure instances and install applications programmatically when an instance is launched. The process of providing code to be run on an instance at launch is called ***bootstrapping***.

One of the parameters when an instance is launched is a string value called UserData. This string is passed to the operating system to be executed as part of the launch process the first time the instance is booted. On Linux instances this can be shell script, and on Windows instances this can be a batch style script or a PowerShell script. The script can perform tasks such as:

- Applying patches and updates to the OS
- Enrolling in a directory service
- Installing application software
- Copying a longer script or program from storage to be run on the instance
- Installing Chef or Puppet and assigning the instance a role so the configuration management software can configure the instance

In addition to importing virtual instances as AMIs, ***VM Import/Export*** enables you to easily import Virtual Machines (VMs) from your existing environment as an Amazon EC2 instance and export them back to your on-premises environment. You can only export previously imported Amazon EC2 instances. Instances launched within AWS from AMIs cannot be exported.

Instance Metadata Instance metadata is data about your instance that you can use to configure or manage the running instance. This is unique in that it is a mechanism to obtain AWS properties of the instance from within the OS without making a call to the AWS API. An HTTP call to *http://169.254.169.254/latest/meta-data/* will return the top node of the instance metadata tree. Instance metadata includes a wide variety of attributes, including:

- The associated security groups
- The instance ID
- The instance type
- The AMI used to launch the instance

This only begins to scratch the surface of the information available in the metadata.

Pricing Options

You are charged for Amazon EC2 instances for each hour that they are in a running state, but the amount you are charged per hour can vary based on three pricing options: *On-Demand Instances, Reserved Instances,* and *Spot Instances*.

On-Demand Instances. The price per hour for each instance type published on the AWS website represents the price for On-Demand Instances. This is the most flexible pricing option, as it requires no up-front commitment, and the customer has control over when the instance is launched and when it is terminated. It is the least cost effective of the three pricing options per compute hour, but its flexibility allows customers to save by provisioning a variable level of compute for unpredictable workloads.

Reserved Instances. The Reserved Instance pricing option enables customers to make capacity reservations for predictable workloads. By using Reserved Instances for these workloads, customers can save up to 75 percent over the on-demand hourly rate. When purchasing a reservation, the customer specifies the instance type and Availability Zone for that Reserved Instance and achieves a lower effective hourly price for that instance for the duration of the reservation. An additional benefit is that capacity in the AWS data centers is reserved for that customer. There are two factors that determine the cost of the reservation: the term ***commitment*** and the payment option. The term commitment is the duration of the reservation and can be either one or three years. The longer the commitment, the bigger the discount.

There are three different payment options for Reserved Instances:

All Upfront—Pay for the entire reservation up front. There is no monthly charge for the customer during the term.
Partial Upfront—Pay a portion of the reservation charge up front and the rest in monthly installments for the duration of the term.
No Upfront—Pay the entire reservation charge in monthly installments for the duration of the term. The amount of the discount is greater the more the customer pays up front.

Spot Instances

For workloads that are not time critical and are tolerant of interruption, Spot Instances offer the greatest discount. With Spot Instances, customers specify the price they are willing to pay for a certain instance type. When the customer's bid price is above the current Spot price, the customer will receive the requested instance(s). These instances will operate like all other Amazon EC2 instances, and the customer will only pay the Spot price for the hours that instance(s) run. The instances will run until:

- The customer terminates them.
- The Spot price goes above the customer's bid price.
- There is not enough unused capacity to meet the demand for Spot Instances.

If Amazon EC2 needs to terminate a Spot Instance, the instance will receive a termination notice providing a two-minute warning prior to Amazon EC2 terminating the instance. Because of the possibility of interruption, Spot Instances should only be used for workloads tolerant of interruption. This could include analytics, financial modeling, big data, media encoding, scientific computing, and testing.

Architectures with Different Pricing Models. It's important to know how to take advantage of the different pricing models to create a cost-efficient architecture. Such an architecture may include different pricing models within the same workload. For instance, a website that averages 5,000 visits a day, but ramps up to 20,000 visits a day during periodic peaks, may purchase two Reserved Instances to handle the average traffic, but depend on On-Demand Instances to fulfill compute needs during the peak times.

Placement Groups

A placement group is a logical grouping of instances within a single Availability Zone. Placement groups enable applications to participate in a low-latency, 10 Gbps network. Placement groups are recommended for applications that benefit from low network latency, high network throughput, or both. Remember that this represents network connectivity between instances. To fully use this network performance for your placement group, choose an instance type that supports enhanced networking and 10 Gbps network performance.

Instance Stores

An instance store (sometimes referred to as *ephemeral storage*) provides temporary block-level storage for your instance. This storage is located on disks that are physically attached to the host computer. An instance store is ideal for temporary storage of information that changes frequently, such as buffers, caches, scratch data, and other temporary content, or for data that is replicated across a fleet of instances, such as a load-balanced pool of web servers. The size and type of instance stores available with an Amazon EC2 instance depend on the instance type. At this writing, storage available with various instance types ranges from no instance stores up to 24 2 TB instance stores. The instance type also determines the type of hardware for the instance store volumes. While some provide Hard Disk Drive (HDD) instance stores, other instance types use Solid State Drives (SSDs) to deliver very high random I/O performance.

Instance stores are included in the cost of an Amazon EC2 instance, so they are a very cost-effective solution for appropriate workloads. The key aspect of instance stores is that they are temporary. Data in the instance store is lost when:

- The underlying disk drive fails.
- The instance stops (the data will persist if an instance reboots).
- The instance terminates.

Therefore, do not rely on instance stores for valuable, long-term data. Instead, build a degree of redundancy via RAID or use a file system that supports redundancy and fault tolerance such as Hadoop's HDFS. Back up the data to more durable data storage solutions such as Amazon Simple Storage Service (Amazon S3) or Amazon EBS often enough to meet recovery point objectives.

Amazon Elastic Block Store (Amazon EBS)

While instance stores are an economical way to fulfill appropriate workloads, their limited persistence makes them ill-suited for many other workloads. For workloads requiring more durable block storage, Amazon provides Amazon EBS.

Elastic Block Store Basics
Amazon EBS provides persistent block-level storage volumes for use with Amazon EC2 instances. Each Amazon EBS volume is automatically replicated within its Availability Zone to protect you from component failure, offering high availability and durability. Amazon EBS volumes are available in a variety of types that differ in performance characteristics and price. Multiple Amazon EBS volumes can be attached to a single Amazon EC2 instance, although a volume can only be attached to a single instance at a time.

Types of Amazon EBS Volumes
Amazon EBS volumes are available in several different types. Types vary in areas such as underlying hardware, performance, and cost. It is important to know the properties of the different types so you can specify the most cost-efficient type that meets a workload's performance demands on the exam.

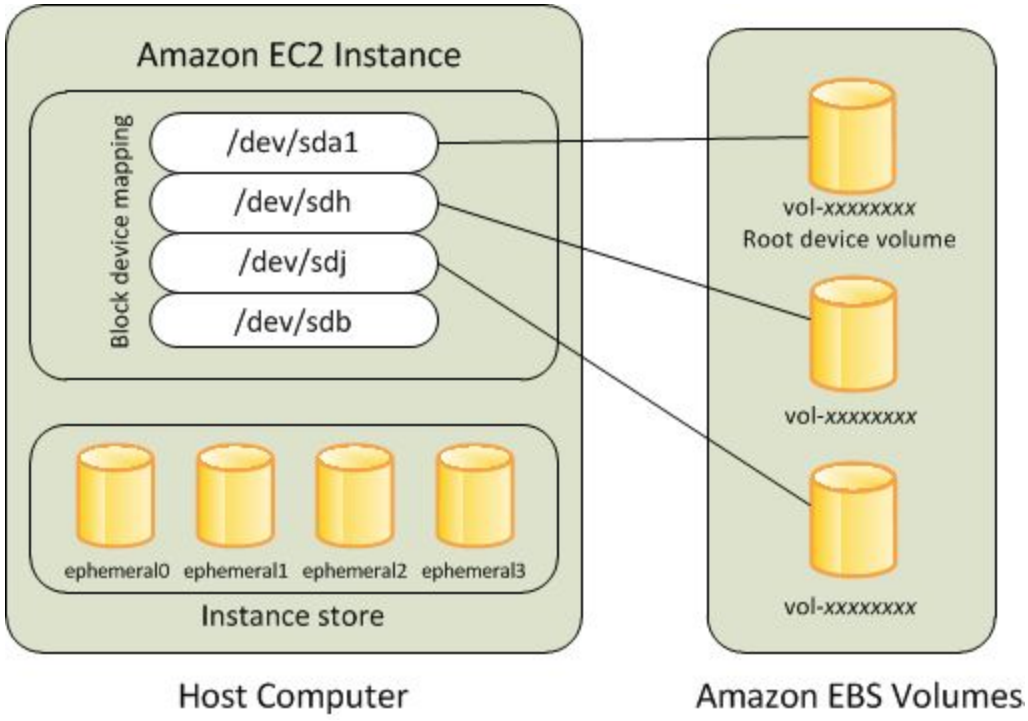

Figure 3.2 AWS EC2 storage types

Magnetic Volumes

Magnetic volumes have the lowest performance characteristics of all Amazon EBS volume types. As such, they cost the lowest per gigabyte. They are an excellent, cost-effective solution for appropriate workloads.

A magnetic Amazon EBS volume can range in size from 1 GB to 1 TB and will average 100 IOPS, but has the ability to burst to hundreds of IOPS. They are best suited for:

- Workloads where data is accessed infrequently
- Sequential reads
- Situations where low-cost storage is a requirement

Magnetic volumes are billed based on the amount of data space provisioned, regardless of how much data you actually store on the volume.

General-Purpose SSD

General-purpose SSD volumes offer cost-effective storage that is ideal for a broad range of workloads. They deliver strong performance at a moderate price point that is suitable for a wide range of workloads.

A general-purpose SSD volume can range in size from 1 GB to 16 TB and provides a baseline performance of three IOPS per gigabyte provisioned, capping at 10,000 IOPS. For instance, if you provision a 1 TB volume, you can expect a baseline performance of 3,000 IOPS. A 5 TB volume will not provide a 15,000 IOPS baseline, as it would hit the cap at 10,000 IOPS. General-purpose SSD volumes under 1 TB also feature the ability to burst to up to 3,000

IOPS for extended periods of time. For instance, if you have a 500 GB volume you can expect a baseline of 1,500 IOPS. Whenever you are not using these IOPS, they are accumulated as I/O credits. When your volume then has heavy traffic, it will use the I/O credits at a rate of up to 3,000 IOPS until they are depleted. At that point, your performance reverts to 1,500 IOPS. At 1 TB, the baseline performance of the volume is already at 3,000 IOPS, so bursting behavior does not apply.

General-purpose SSD volumes are billed based on the amount of data space provisioned, regardless of how much data you actually store on the volume. They are suited for a wide range of workloads where the very highest disk performance is not critical, such as:

- System boot volumes
- Small- to medium-sized databases
- Development and test environments

Provisioned IOPS SSD

Provisioned IOPS SSD volumes are designed to meet the needs of I/O-intensive workloads, particularly database workloads that are sensitive to storage performance and consistency in random access I/O throughput. While they are the most expensive Amazon EBS volume type per gigabyte, they provide the highest performance of any Amazon EBS volume type in a predictable manner.

A Provisioned IOPS SSD volume can range in size from 4 GB to 16 TB. When you provision a Provisioned IOPS SSD volume, you specify not just the size, but also the desired number of IOPS, up to the lower of the maximum of 30 times the number of GB of the volume, or 20,000 IOPS. You can stripe multiple volumes together in a RAID 0 configuration for larger size and greater performance. Amazon EBS delivers within 10 percent of the provisioned IOPS performance 99.9 percent of the time over a given year.

Pricing is based on the size of the volume and the amount of IOPS reserved. The cost per gigabyte is slightly more than that of general-purpose SSD volumes and is applied based on the size of the volume, not the amount of the volume used to store data. An additional monthly fee is applied based on the number of IOPS provisioned, whether they are consumed or not.

Provisioned IOPS SSD volumes provide predictable, high performance and are well suited for:

- Critical business applications that require sustained IOPS performance
- Large database workloads

Characteristic	General-Purpose SSD	Provisioned IOPS SSD	Magnetic
Use cases	• System boot volumes • Virtual desktops • Small-to-medium sized databases • Development and test environments	• Critical business applications that require sustained IOPS performance or more than 10,000 IOPS or 160MB of throughput per volume • Large database workloads	• Cold workloads where data is infrequently accessed • Scenarios where the lowest storage cost is important
Volume size	1 GiB–16TiB	4 GiB–16TiB	1 GiB–1TiB
Maximum throughput	160MB	320MB	40–90MB
IOPS performance	Baseline performance of 3 IOPS/GiB (up to 10,000 IOPS) with the ability to burst to 3,000 IOPS for volumes under 1,000 GiB	Consistently performs at provisioned level, up to 20,000 IOPS maximum	Averages 100 IOPS, with the ability to burst to hundreds of IOPS

Amazon EBS-Optimized Instances

When using any volume type other than magnetic and Amazon EBS I/O is of consequence, it is important to use Amazon EBS-optimized instances to ensure that the Amazon EC2instance is prepared to take advantage of the I/O of the Amazon EBS volume. An Amazon EBS-optimized instance uses an optimized configuration stack and provides additional, dedicated capacity for Amazon EBS I/O. This optimization provides the best performance for your Amazon EBS volumes by minimizing contention between Amazon EBS I/O and other traffic from your instance. When you select Amazon EBS-optimized for an instance, you pay an additional hourly charge for that instance. Check the AWS documentation to confirm which instance types are available as Amazon EBS-optimized instance.

Protecting Data

Over the lifecycle of an Amazon EBS volume, there are several practices and services that you should know about when taking the exam.

Backup/Recovery (Snapshots)

You can back up the data on your Amazon EBS volumes, regardless of volume type, by taking point-in-time snapshots. Snapshots are incremental backups, which means that only the blocks on the device that have changed since your most recent snapshot are saved.

Taking Snapshots

You can take snapshots in many ways:

- Through the AWS Management Console

- Through the CLI
- Through the API
- By setting up a schedule of regular snapshots

Data for the snapshot is stored using Amazon S3 technology. The action of taking a snapshot is free. You pay only the storage costs for the snapshot data. When you request a snapshot, the point-in-time snapshot is created immediately and the volume may continue to be used, but the snapshot may remain in pending status until all the modified blocks have been transferred to Amazon S3.

It's important to know that while snapshots are stored using Amazon S3 technology, they are stored in AWS-controlled storage and not in your account's Amazon S3 buckets. This means you cannot manipulate them like other Amazon S3 objects. Rather, you must use the Amazon EBS snapshot features to manage them. Snapshots are constrained to the region in which they are created, meaning you can use them to create new volumes only in the same region. If you need to restore a snapshot in a different region, you can copy a snapshot to another region.

Creating a Volume from a Snapshot

To use a snapshot, you create a new Amazon EBS volume from the snapshot. When you do this, the volume is created immediately but the data is loaded lazily. This means that the volume can be accessed upon creation, and if the data being requested has not yet been restored, it will be restored upon first request. Because of this, it is a best practice to initialize a volume created from a snapshot by accessing all the blocks in the volume.

Snapshots can also be used to increase the size of an Amazon EBS volume. To increase the size of an Amazon EBS volume, take a snapshot of the volume, then create a new volume of the desired size from the snapshot. Replace the original volume with the new volume.

Recovering Volumes

Because Amazon EBS volumes persist beyond the lifetime of an instance, it is possible to recover data if an instance fails. If an Amazon EBS-backed instance fails and there is data on the boot drive, it is relatively straightforward to detach the volume from the instance. Unless the DeleteOnTermination flag for the volume has been set to false, the volume should be detached before the instance is terminated. The volume can then be attached as a data volume to another instance and the data read and recovered.

Encryption Options

Many workloads have requirements that data be encrypted at rest, either because of compliance regulations or internal corporate standards. Amazon EBS offers native encryption on all volume types.

When you launch an encrypted Amazon EBS volume, Amazon uses the AWS Key Management Service (KMS) to handle key management. A new master key will be created unless you select a master key that you created separately in the service. Your data and associated keys are encrypted using the industry-standard AES-256 algorithm. The encryption occurs on the servers

that host Amazon EC2 instances, so the data is actually encrypted in transit between the host and the storage media and also on the media. (Consult the AWS documentation for a list of instance types that support Amazon EBS encryption.) Encryption is transparent, so all data access is the same as unencrypted volumes, and you can expect the same IOPS performance on encrypted volumes as you would with unencrypted volumes, with a minimal effect on latency. Snapshots that are taken from encrypted volumes are automatically encrypted, as are volumes that are created from encrypted snapshots.