What is Compute?

Essentially, compute is closely related to common server components that many of you will be familiar with such as**central processing units (CPUs)** and **random access memory (RAM)**.

What is Instance?

A cloud instance is a virtual server instance in a cloud computing environment. It is built, hosted and delivered using a cloud computing platform, and can be accessed remotely. ... Cloud instances are also known as cloud servers, virtual servers, virtual machines etc.

Example: MAC Instance which is provided by Apple in AWS cloud which has 12 Vcpus, 32Gib of memory

Compute services:

Amazon EC2

Amazon EC2 Auto Scaling

Amazon EC2 Image Builder

Amazon Lightsail

AWS App Runner

AWS Batch

AWS Elastic Beanstalk

AWS Fargate

AWS Lambda

AWS Outposts

AWS Wavelength

Types of Instances:

What are the different types of EC2 Instances?

Within those greater families are unique instance types. Here’s a list of instance types for just the General Purpose Family:

t2.nano, t2.micro, t2.small, t2.medium, t2.large, t2.xlarge, t2.2xlarge

t3.nano, t3.micro, t3.small, t3.medium, t3.large, t3.xlarge, t3.2xlarge

m4.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m4.10xlarge, m4.16xlarge

m5.large, m5.xlarge, m5.2xlarge, m5.4xlarge, m5.12xlarge, m5.24xlarge

m6.large, m6.xlarge, m6.2xlarge, m6.4xlarge, m6.12xlarge, m6.24xlarge

Notice that T2, T3, M4, M5, and M6 are the different instance types, each having varying sizing options that range from nano to 24xlarge. From left to right, these instance types go from oldest to newest in generation. Over time, AWS deprecates the older generations.

It’s important to pick carefully when choosing your AMI, Instance Family, and Instance type. All three of these components factor greatly when determining the overall price per hour Amazon charges you.

Speaking of pricing, let’s take a look at Amazon’s pricing options.

EC2 Instance Pricing Options

When it comes to paying for Amazon Web Services, there are four main purchasing options: On-demand Instances, Reserved Instances, Spot Instances, and Dedicated Hosts. Each pricing option is tailored to benefit a particular usage behavior.

Properly evaluating your environment to determine the best payment method for each instance is crucial to avoiding wasteful spending. But be careful not to get too aggressive when trying to save–choosing the cheapest available option is not always the smartest choice. Doing so may lead to performance issues for your application that would be difficult to diagnose in production (especially if it involves mis-allocation related to your disk I/O requirements).

After reading the following sections, you’ll have a clear understanding of when to leverage each pricing option

On-Demand Instances

On-Demand is the standard pay-as-you-use payment option. Usage is billed either per-hour or per-second, depending on the kind of instances being run.

On-Demand Instances are great for:

Temporary projects

Testing/Developing with EC2s for the first time

Applications with unpredictable and spiky workloads

Essentially, these are a great first choice for anyone new to AWS who wants to learn without overcommitting. On-Demand Instances are also great for uncertain, flexible projects. This is because you’ll never have to pay for unused resources (which can happen with reservations) and your EC2 can still scale with growing workloads.

Spot Instances

You can obtain deep discounts of up to 90% by leveraging spot instances. How is that possible? It’s simple: spot instances are a limited pooled resource. They aren’t always available for the price you’re willing to pay. And when the price goes above your set bidding threshold, your spot instances are interrupted (via hibernation, stopping, or termination). So when choosing Spot Instances, it’s always best to pick fault-tolerant workloads and stateless distributed applications that can tolerate losing partial or full computing capacity without affecting your business processes.

Spot Instances are great for:

Containerized workloads

Big Data

Test & Development workloads

Reserved Instances

For the forward-thinker, Reserved Instances are a great resource. You can save up to 75% in cost by choosing Reserved Instances over On-Demand ones. When using RIs, you also get the added optional benefit of reserving capacity in your preferred Availability Zones.

That sounds great, right? Why wouldn’t you just use RIs exclusively? Well, the downside is that you have to commit to a long-term contract of at least one year. This means continuing to pay a monthly amount even if your need goes away in six months. Also, to get the most savings possible, you have to pay upfront. That may not fit everyone’s budget.

Don’t worry, though. There’s also partial upfront and no upfront payment options for Reserved Instances–meaning you’ll still save on your EC2s, just not as much.

Reserved Instances are great for:

Customers able to commit to 1 or 3 year contracts

Applications that require steady state resources

Applications that would benefit from reserved capacity

You can modify or exchange your reserved instances with convertible reservations

Standard reserved instances can be put up for sale on the Reserved Instance Marketplace

Dedicated Hosts

Still like the idea of having your own physical server and have full control of your hardware? Dedicated Hosts are your solution. AWS offers dedicated physical EC2 servers through the On-Demand and Reserved Instance payment methods.

Dedicated Hosts are great for:

Avoiding the sharing of physical computing resources with other AWS customers

Using your existing server-bound software licenses

Meeting compliance requirements

Note that AWS also offers Dedicated Instances (vs. Hosts).

The difference is that Dedicated Instances may be provisioned on a different physical server each time (preventing you from using your own software licenses tied to a specific socket, core, or CPU serial number).

How Does AWS Charge for EC2s?

In October of 2017, AWS announced that Amazon EC2 usage of Linux-based instances launched in On-Demand, Reserved, and Spot form (along with provisioned EBS volumes) will be billed on one second increments, with a minimum of 60 seconds. This means that you would no longer pay for a full hour if you only use the instance in a running state for a few minutes. You also get the benefit of reserved instance pricing for multiple instances used within the hour. Remember that AWS EC2 pricing may still be expressed in the form of instance-hour on many of their pricing pages to preserve consistency.

Dedicated Instances, EBS Snapshots, and products in AWS Marketplace are still billed on an hourly basis.

Data Transfer Costs

Data transfer costs can be divided into two general categories:

Data transfers in between AWS services (like from EC2 to S3)

Across Availability Zones in Same Region: $0.01/GB each direction

Within Same Availability Zone: typically free

Across Regions:varies by region

Data transfers from AWS to the public internet

IPv4 (Public or Elastic) $0.01/GB either direction

IPv6 in Different VPS: $0.01/GB either direction

Regional Out To Internet: varies by region

The more data that is sent out from AWS, the less expensive it gets. Here are the current pricing tiers: 1GB, 9.999TB, 40TB, 100TB, 150TB, 500TB. AWS allows you to commit to a volume of data transfer per month in exchange for a more favorable pricing.

Remember, each of these tiers will have different pricing values for each region.

One-Time Dedicated Instance Costs

Dedicated Instances have two main costs:

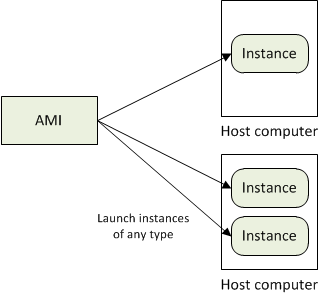
Per hour instance usage fees for each instance running

Price varies by Region and AMI

Per hour dedicated region fee for each region being used

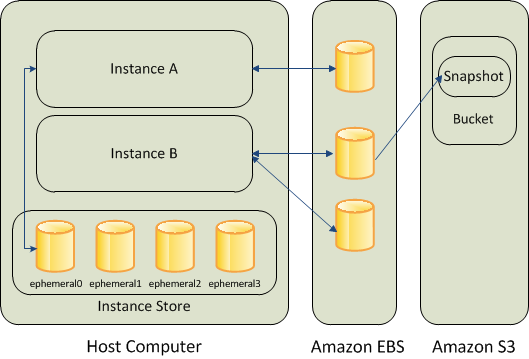
$2 per region used regardless of the number of instances in them

* Amazon Machine Image (AMI) is a template for software configuration (Operating System, Application Server, and Applications)
* Instance is a AMI running on virtual servers in the cloud
* Each *instance type* offers different compute and memory facilities



Storage:

* Amazon EC2 provides three type of storage option
  + Amazon EBS
  + Amazon S3
  + Instance Storage



Amazon EBS (Elastic Block Store) provides with persistent, block-level storage. Basically additional Hard Disk that you can attach to instance.

Suitable for apps which require database , filesystem , block level storage.

Create, Attach , Detach, Delete

- Standard volumes and Provisioned IOPS volumes.

Take snapshots that is stored in S3 , a new EBS can be re-created using the snapshot

Instance store comes with each instance except the micro-one , temporary block level storage.

Storage physically attached to the computer

S3 simple storage service

storage for the Internet

web service interface that enables you to store and retrieve any amount of data from anywhere on the web

Root device storage: contains image to boot the system

AMI categorized as “Backed by Amazon EBS” or “Backed by instance store”

VPC - launch Amazon Web Services (AWS) resources into a virtual network that you've defined.

Configure VPC: select its IP address range, create subnets, and configure route tables, network gateways, and security settings.

Security Group - enables you to specify the protocols, ports, and source IP ranges that are allowed to reach your instances

Create multiple security groups , assign instance to a particular group , determine the traffic