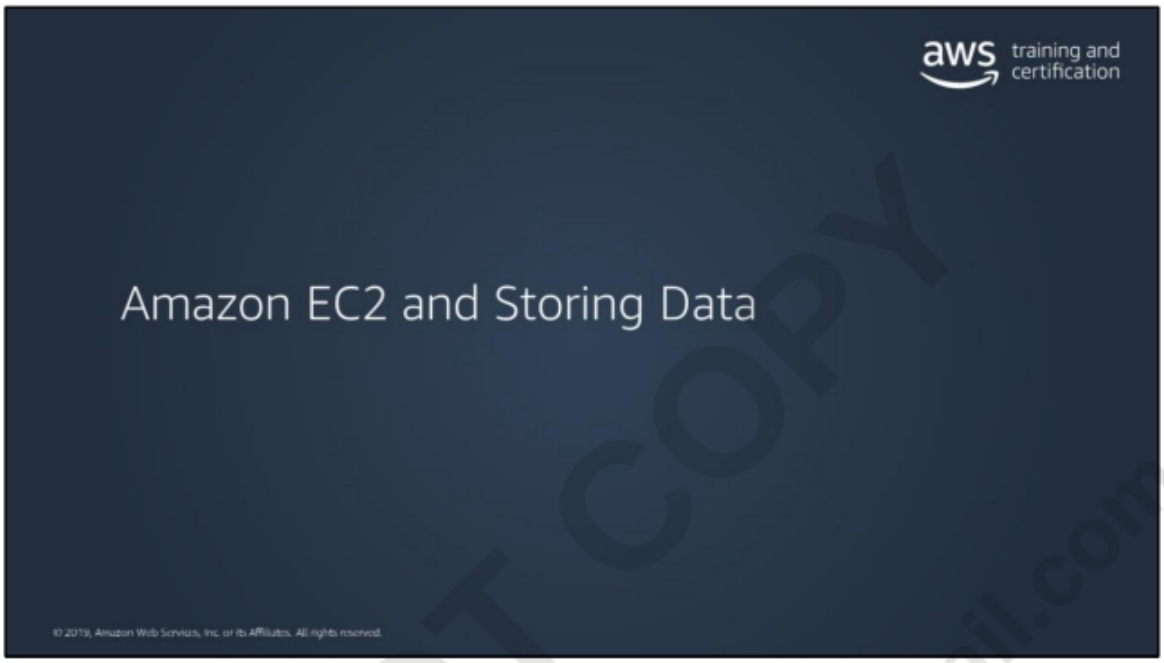
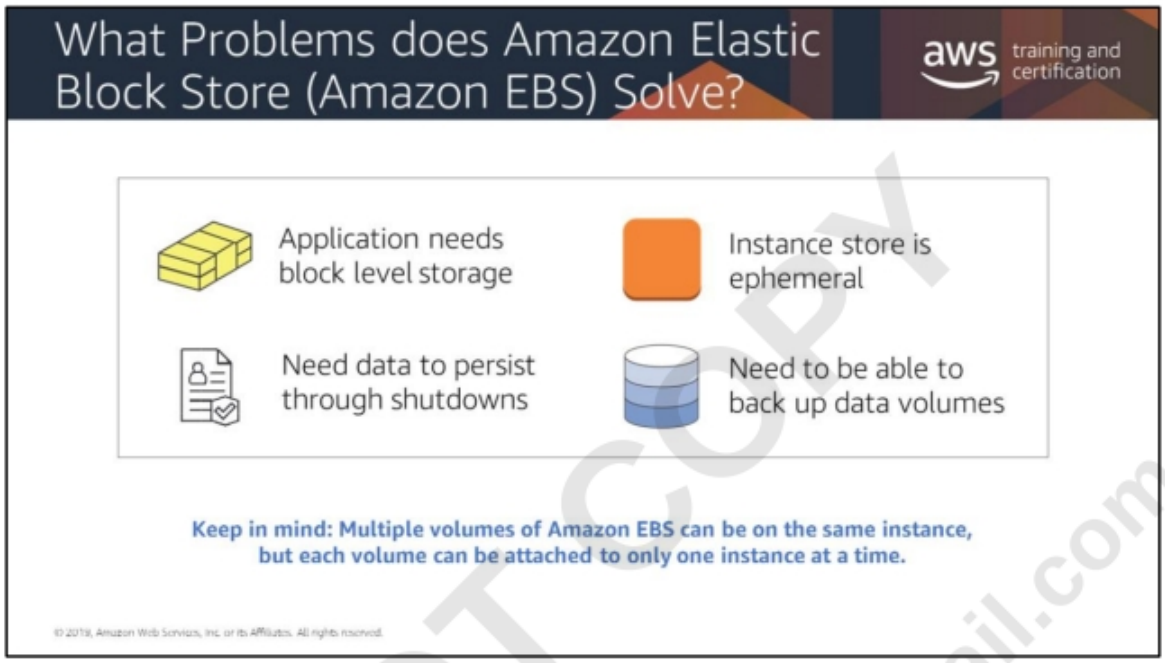


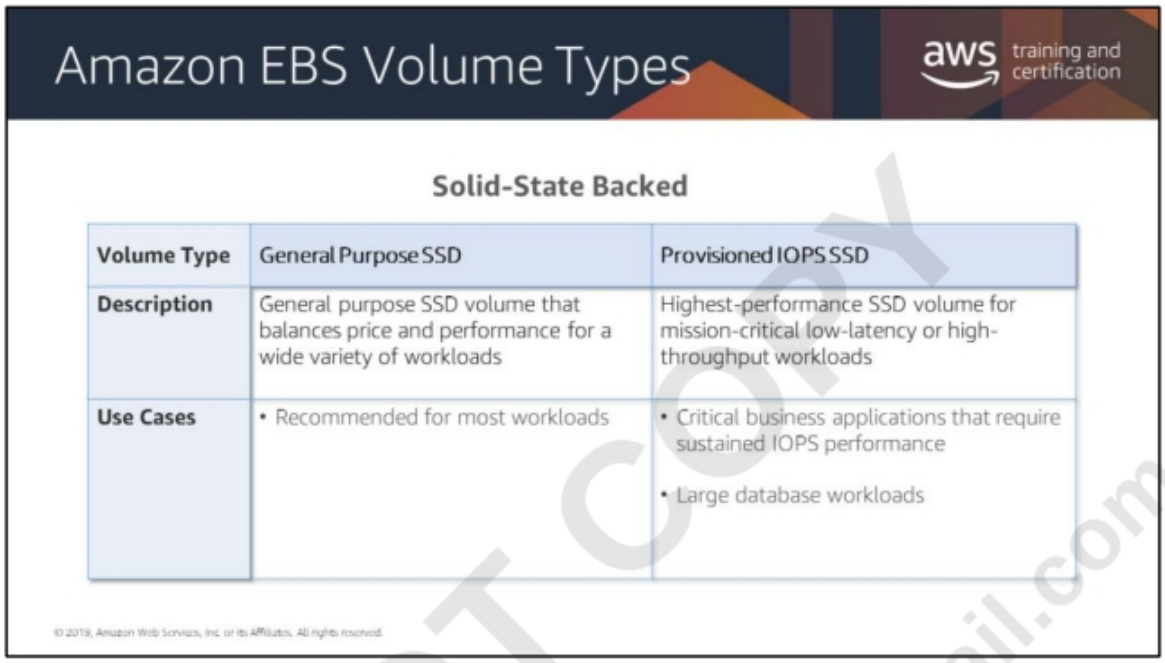


Amazon EBS volumes provide durable, detachable, block-level storage (like an external hard drive) for your Amazon EC2 instances. Because they are mounted to the instances, they can provide extremely low latency between where the data is stored and where it might be used on the instance. For this reason, they can be used to run a database with an Amazon EC2 instance. Amazon EBS volumes can also be used to back up your instances into AMIs, which are stored in Amazon S3 and can be reused to create new Amazon EC2 instances later.

An *instance store* provides **temporary** block-level storage for your instance. This storage is located on disks that are physically attached to the host computer. 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.

For more information on EBS, see:
https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AmazonEBS.html
For more information on Instance Storage, See:
https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/InstanceStorage.html

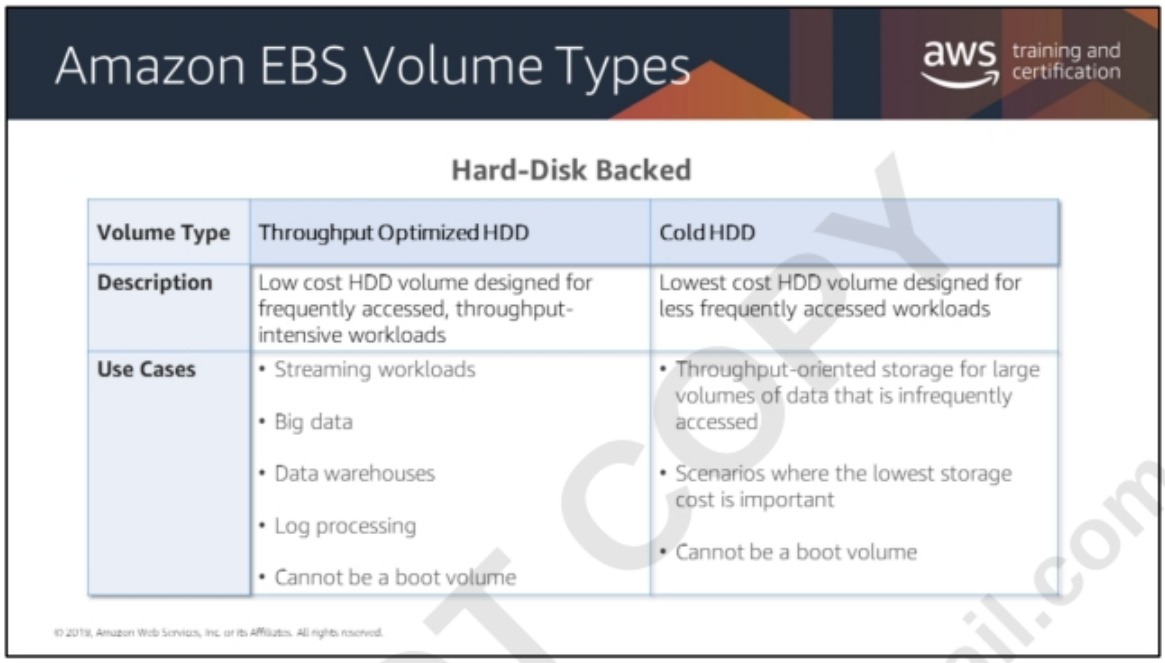
/

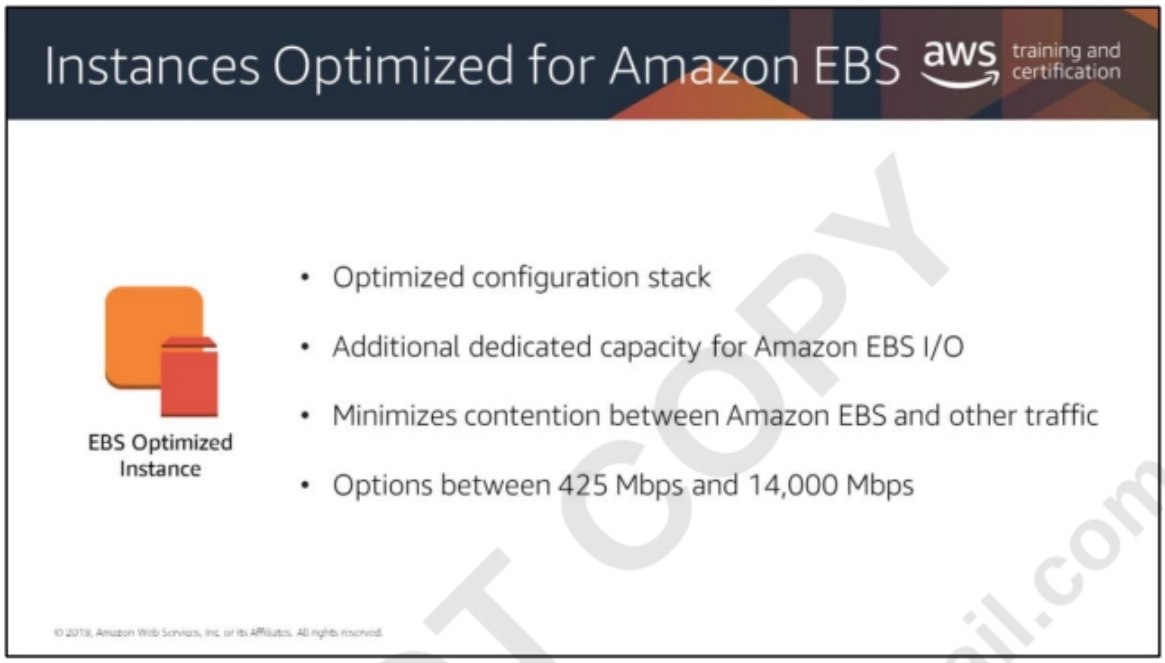


SSD-backed volumes are optimized for transactional workloads involving frequent read/write operations with small I/O size, where the dominant performance attribute is IOPS.

HDD-backed volumes are optimized for large streaming workloads where throughput (measured in MiB/s) is a better performance measure than IOPS.

For more information about Amazon EBS volume types, see https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSVolumeTypes.html



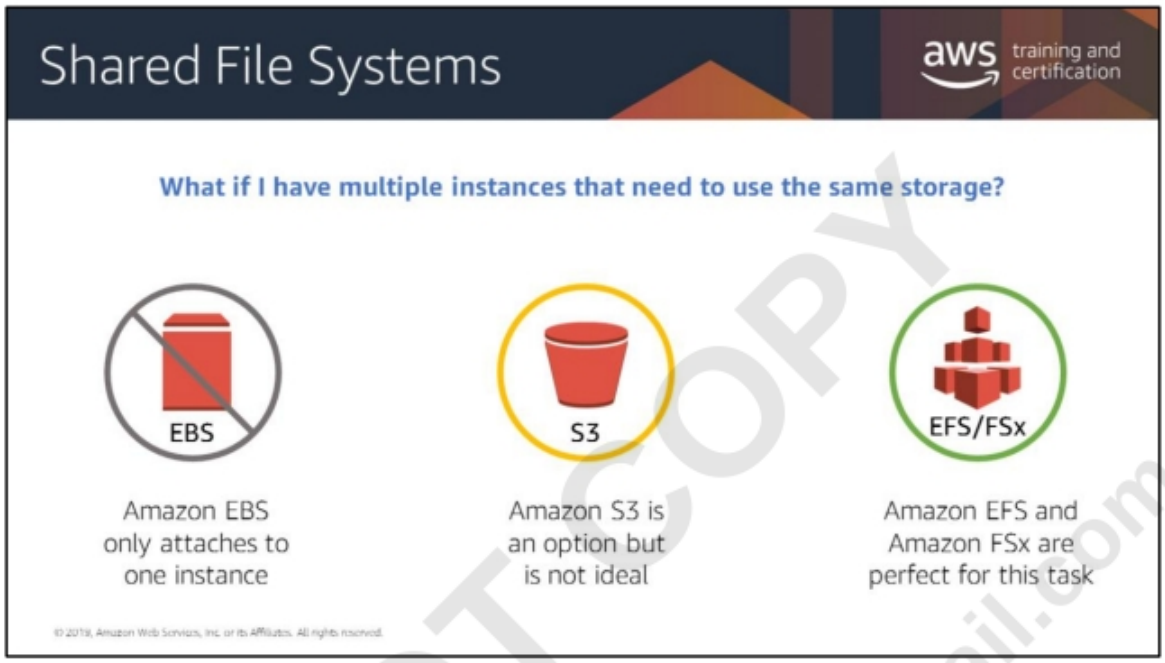


An instance optimized for Amazon EBS uses an optimized configuration stack and provides additional dedicated capacity for Amazon EBS I/O. This optimization provides the best performance for your EBS volumes by minimizing contention between Amazon EBS I/O and other traffic from your instance.
EBS-optimized instances deliver dedicated bandwidth to Amazon EBS, with options between 425 Mbps and 14,000 Mbps, depending on the instance type you use.

For more information, see
https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSOptimized.html

If your EC2 instance is EBS-backed, you can put it into EC2 Hibernation. This keeps the in-memory storage, private IP, and elastic IP to remain the same, and allows you to pick up when the instance left off. It currently can only be enabled on Linux1 EC2 instances, with Linux2 support coming soon. When the instance is in hibernation, you pay only for the EBS volume attached, and the Elastic IP in your account.
https://aws.amazon.com/blogs/aws/new-hibernate-your-ec2-instances/

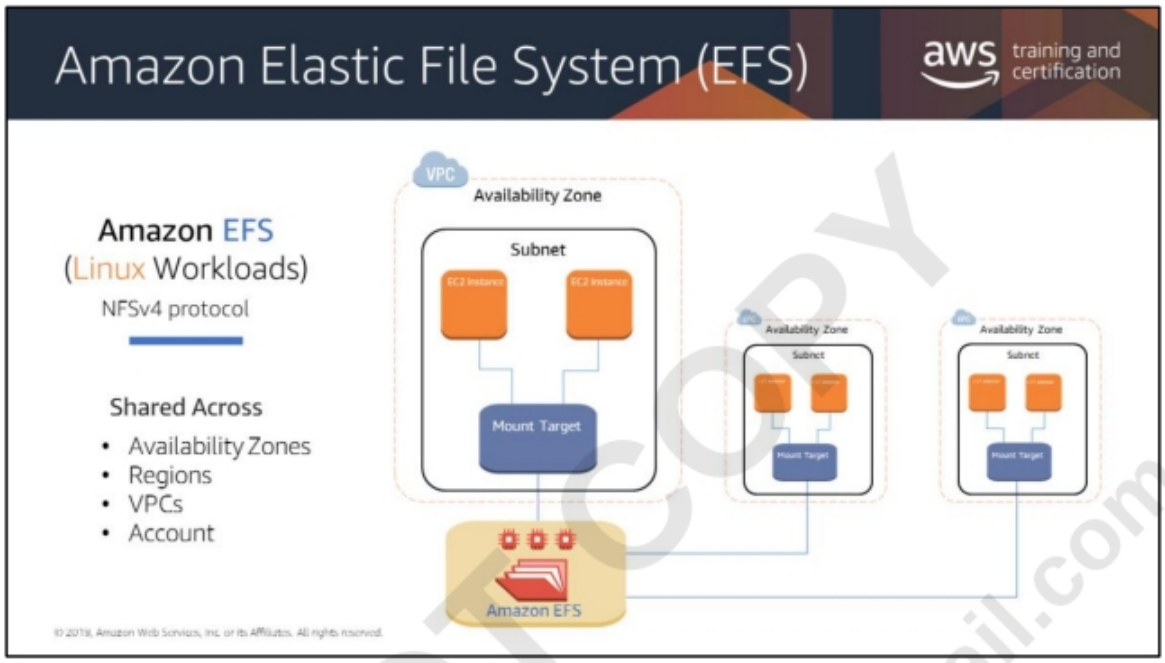
/



Problem: How do we handle an application running on multiple instances that needs to use the same file system? Amazon S3 is one option, but what if you need the performance and read-write consistency of a network file system? Amazon Elastic File System (Amazon EFS) may be your best option.

S3 is an object store system, not a block store, so changes overwrite entire files, not blocks of characters within files. For high throughput changes to files of varying sizes, a file system will be superior to an object store system for handling those changes.

/



Amazon Elastic File System (Amazon EFS) provides a simple, scalable, elastic file system for Linux-based workloads for use with AWS Cloud services and on-premises resources. You're able to access your file system across Availability Zones, AWS Regions, and VPCs while sharing files between thousands of EC2 instances and on-premises servers via AWS Direct Connect or AWS VPN. You can create a file system, mount the file system on an Amazon EC2 instance, and then read and write data to and from your file system. You can mount an Amazon EFS file system in your VPC, through the Network File System versions 4.0 and 4.1 (NFSv4) protocol.

The Amazon EFS file system can be accessed concurrently from Amazon EC2 instances in your Amazon VPC, so applications that scale beyond a single connection can access a file system. Amazon EC2 instances running in multiple Availability Zones within the same region can access the file system, so that many users can access and share a common data source.

For more information about access across accounts and VPCs, see Amazon EFS now Supports Access Across Accounts and VPCs
For more information about VPC peering, see Mounting EFS File Systems from Another Account or VPC

For a list of Amazon EC2 Linux Amazon Machine Images (AMIs) that support this protocol, see NFS Support. We recommend using a current generation Linux NFSv4.1 client, such as those found in Amazon Linux and Ubuntu AMIs. For some AMIs, you'll need to install an NFS client to mount your file system on your Amazon EC2 instance. For instructions, see Installing the NFS Client.

Note the following restrictions:

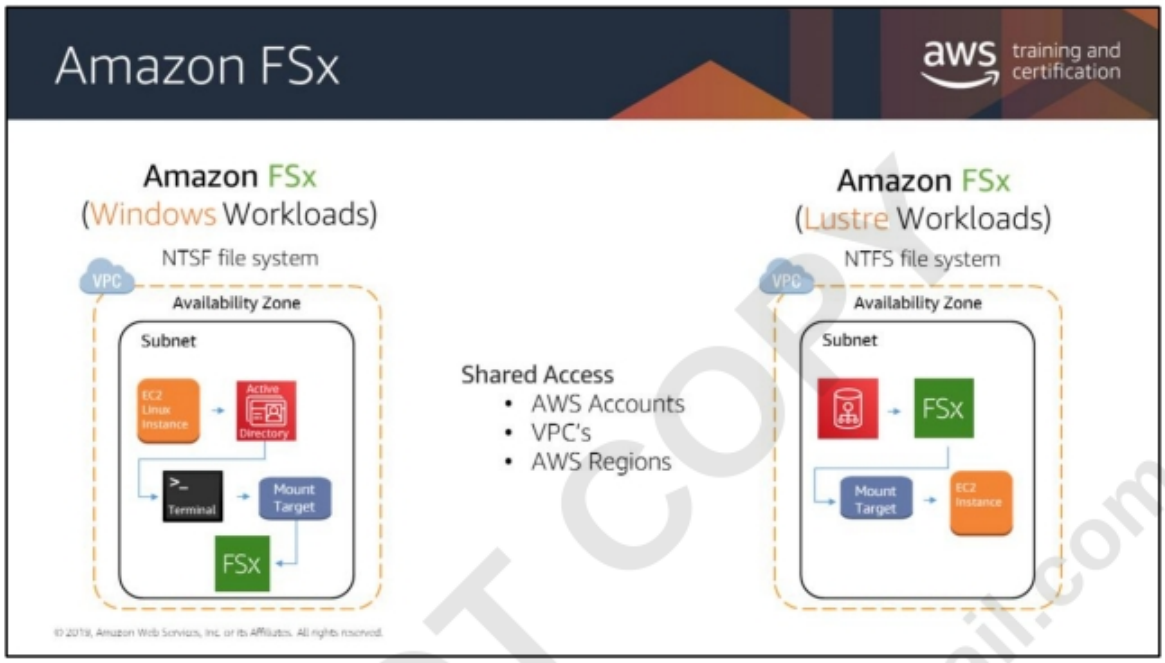
- You can mount an Amazon EFS file system on instances in only one VPC at a time.
- Both the file system and VPC must be in the same AWS Region.

How is File Storage Different?

Although object storage solutions enable storage of files as objects, accessing with existing applications requires new code and the use of APIs and direct knowledge of naming semantics. File storage solutions that support existing file system semantics and permissions models have a distinct advantage in that they do not require new code to be written to integrate with applications that are easily configured to work with shared file storage.

Block storage can be used as the underlying storage component of a self-managed file storage solution. However, the one-to-one relationship required between the host and volume makes it difficult to have the scalability, availability, and affordability of a fully managed file storage solution and would require additional budget and management resources to support. Using a fully managed cloud file storage solution removes complexities, reduces costs, and simplifies management.

/



Amazon FSx provides you with two file systems from which to choose: Amazon FSx for Windows File Server for Windows-based applications and Amazon FSx for Lustre for compute-intensive workloads.

Amazon FSx for Windows File Server was designed for Enterprise applications and is a completely managed service backed by a native windows file system. It is an easily lift and shift enterprise application to Amazon Web Services. Built on SSD storage, Amazon FSx for Windows File Server is ideal for supporting Windows workloads that require shared storage such as CRM, ERP, .NET applications, and user home directories. Thousands of compute instances can access a single Amazon FSx file system at the same time, which provides on-premise access via AWS Direct Connect or AWS VPN, and access from multiple VPCs, accounts, and regions using VPC Peering or AWS Transit Gateway. Amazon FSx for Windows File Server provides a shared file storage system for your Windows Amazon EC2 instances with high levels of throughput and sub-millisecond latency. Amazon FSx for Windows File Server supports:

- SMB Protocol
- Windows NTFS
- Active Directory (AD) Integration
- Distributed File System (DFS)

Amazon FSx for Windows File Server can also be mounted on an Amazon EC2 Linux instance. See Using Microsoft Windows File Shares for further instructions.

Amazon FSx for Lustre

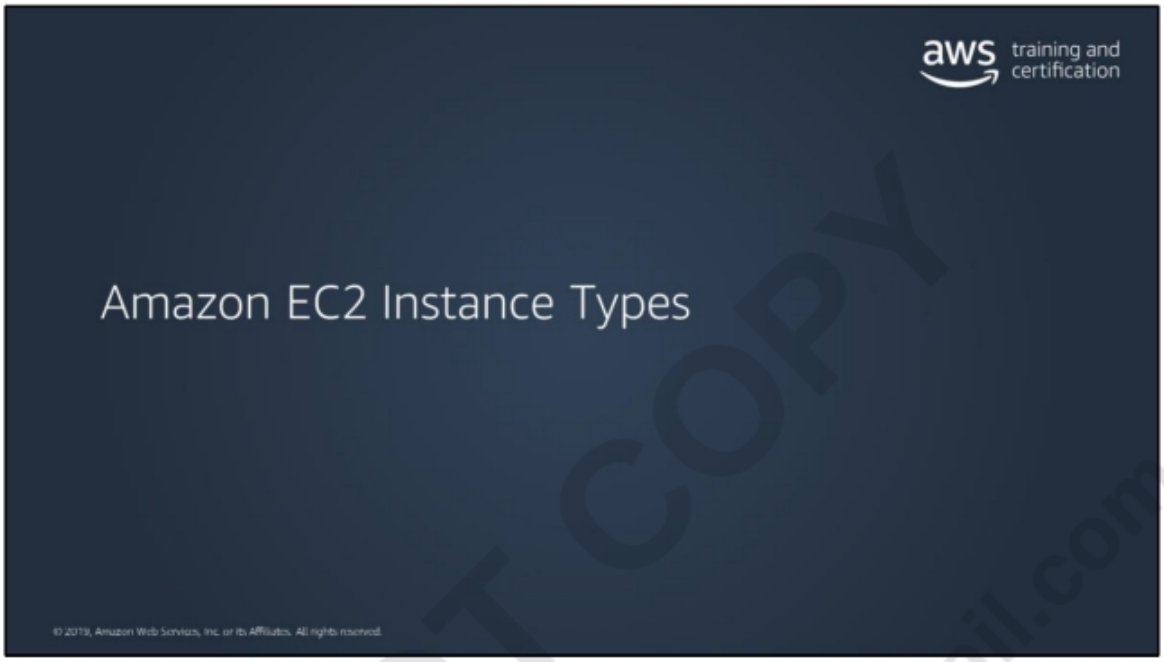
Amazon FSx for Lustre provides a similar, fully managed file system that is optimized for high performance computing (HPC), machine learning, and media processing workflows. A single Amazon FSx for Lustre file system can process massive data sets with hundreds of gigabytes (GB) per second of throughput at sub-millisecond latencies. Amazon FSx for Lustre can be integrated with Amazon S3, so you can join long-term data sets with a high performance file system. Data can be automatically copied to and from Amazon S3 from your Amazon FSx for Lustre file system.

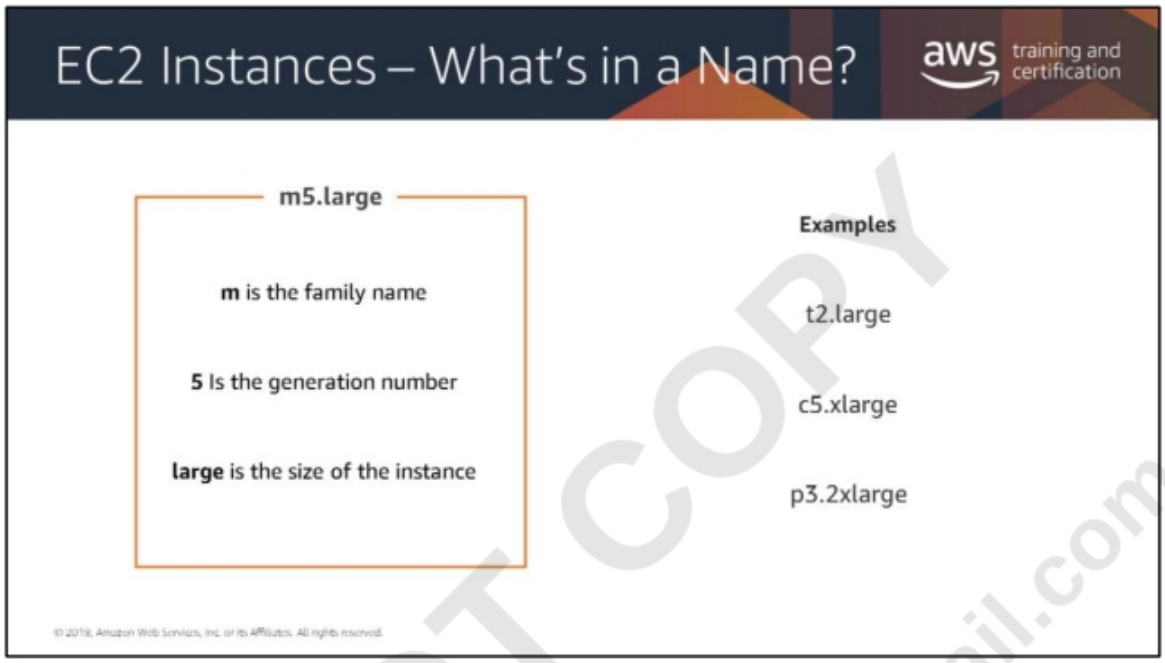
Amazon FSx for Lustre is POSIX-compliant, so you can use your current Linux-based applications without having to make any changes. FSx for Lustre provides a native file system interface and works as any file system does with your Linux operating system. It also provides read-after-write consistency and supports file locking. You can control access to your FSx for Lustre file systems with POSIX permissions and Amazon Virtual Private Cloud (VPC) permissions.

Amazon FSx for Lustre can also be mounted to an Amazon EC2 Instance. See Mounting from an Amazon EC2 Instance for more information.

Both Amazon FSx offerings support connection with on-premises workloads using AWS Direct Connect or a VPN connection. With both offerings, you pay only for the resources you use.

/

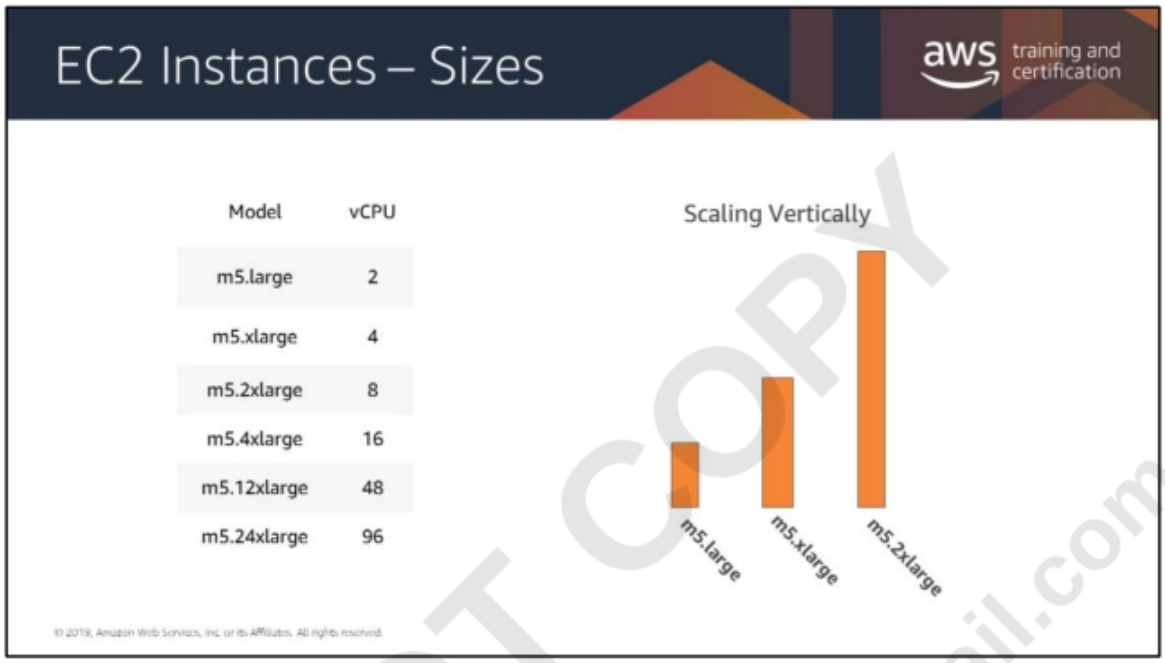




When looking at an instance type, you will see that the model has a few parts to its name—as an example, take the M type.

M is the family name, which is then followed up by a number. Here, that number is 5. The number is the generation number of that type. So, an M5 instance is the 5th generation of the M family. In general, instances of a higher generation are more powerful and provide a better value for the price.

/



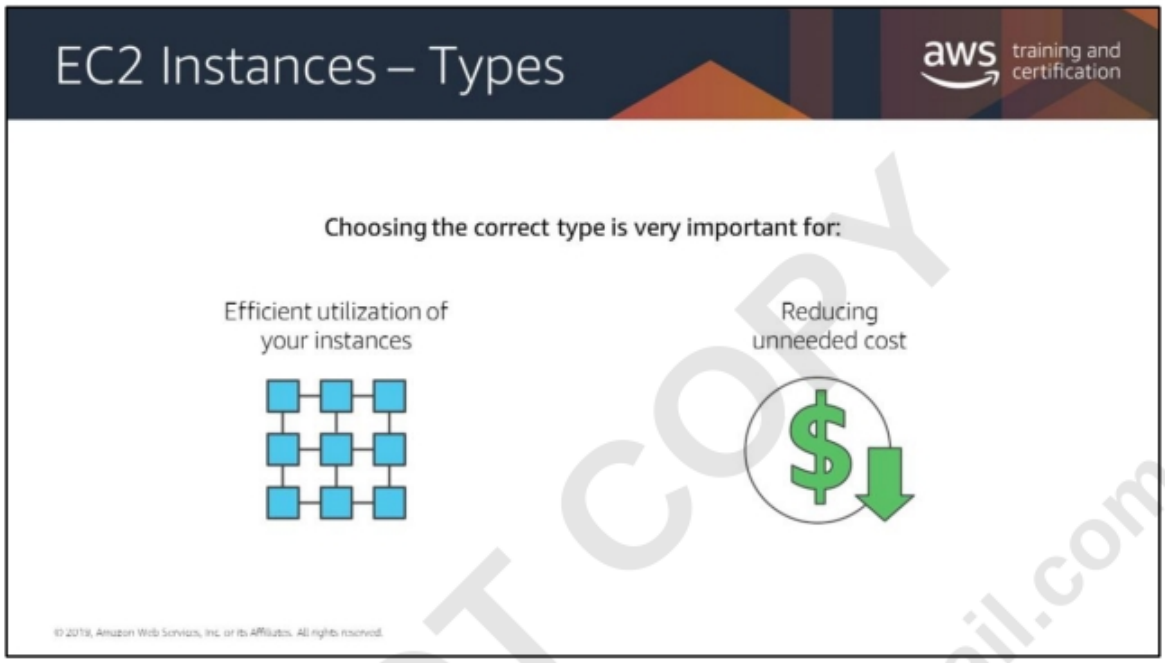
The next part of the name is the size portion of the instance. When comparing sizes, it's important to look at the coefficient portion of the size category.

For example a m5.2xlarge is twice as big as a m5.xlarge. This m5.xlarge is in turn twice as big as the m5.large.

You will notice later on in the chart there is a m5.12xlarge. This instance is 12 times as powerful as the m5.xlarge.

It is also important to note that network bandwidth is also tied to the size of your ec2 instance. If you are performing a task that is very network intensive you might be required to increase your instance specs in order to meet those needs.

/

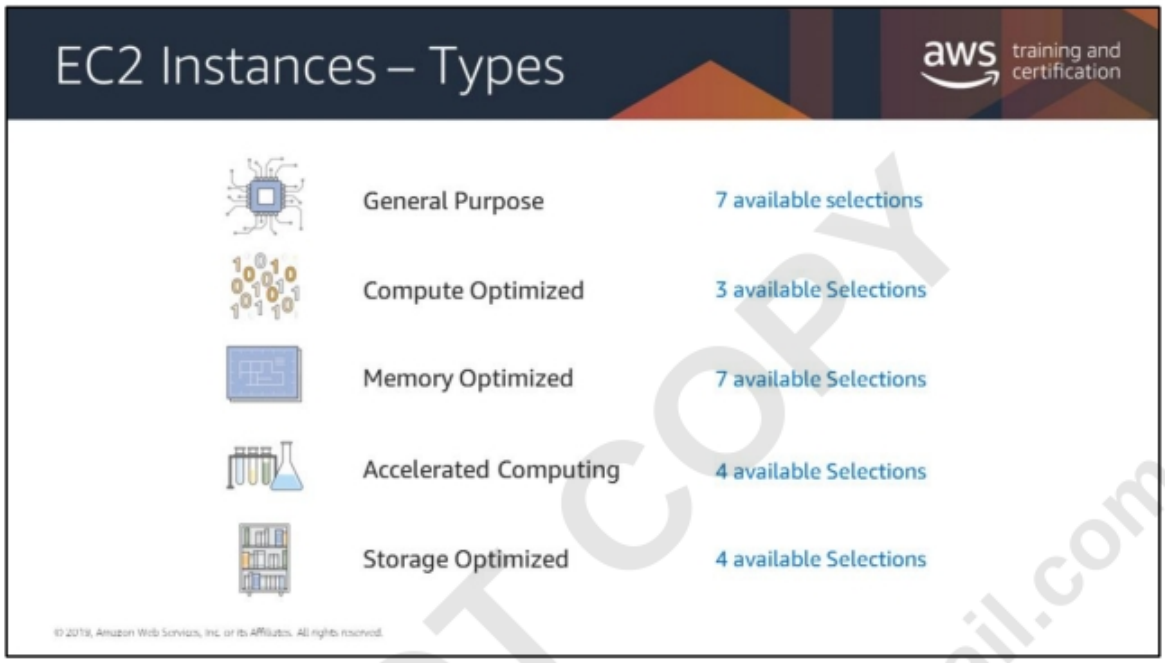


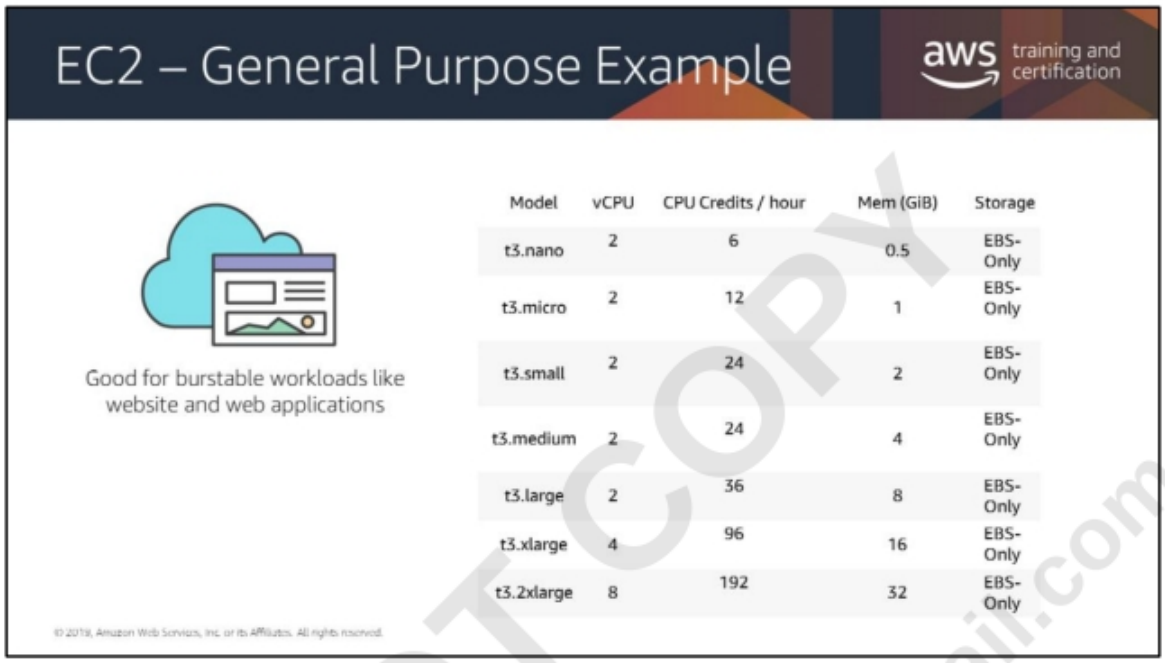
Choosing the correct instance type is very important for reducing unneeded cost and increasing utilization of an instance.

Each instance family has its own positives that need to be addressed when deciding how you are going to architect your solution.

Let's take a look at all of the instance families and see what their recommended workloads are.

/

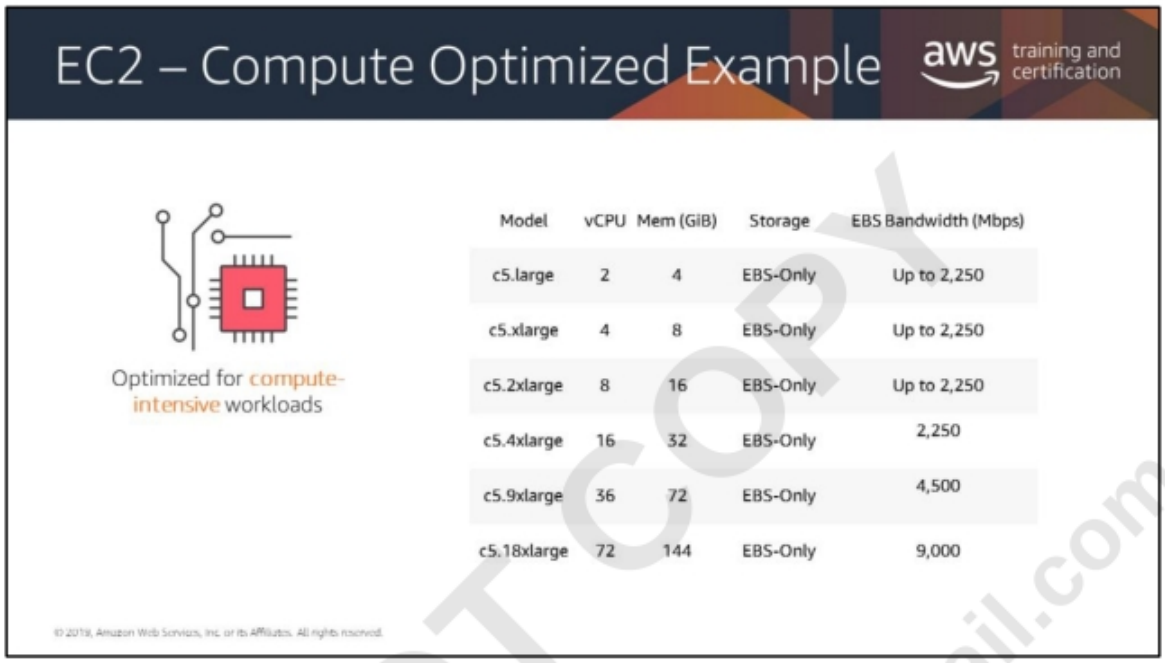




T2 instances are burstable performance instances that provide a baseline level of CPU performance with the ability to burst above the baseline.

Use cases for this type of instance include websites and web applications, development environments, build servers, code repositories, micro services, test and staging environments, and line of business applications.

/



C5 instances are optimized for compute-intensive workloads and deliver very cost-effective high performance at a low price per compute ratio.

Use cases include high-performance web servers, scientific modelling, batch processing, distributed analytics, high-performance computing (HPC), machine/deep learning inference, ad serving, highly scalable multiplayer gaming, and video encoding.

Consider using the Elastic Fabric Adapter for your HPC workloads: https://aws.amazon.com/about-aws/whats-new/2018/11/introducing-elastic-fabric-adapter/
Elastic Fabric Adapter, or EFA, is a network adapter for Amazon EC2 instances that delivers the performance of on-premises HPC clusters with the elasticity and scalability of AWS. You can run HPC applications that require high levels of inter-instance communications, such as computational fluid dynamics, weather modeling, and reservoir simulation. In addition, HPC applications use popular HPC technologies, such as Message Passing Interface (MPI), which can scale to thousands of CPU cores. EFA supports industry-standard libfabric APIs, so applications that use a supported MPI library can be migrated to AWS with little or no modification.

(Note: EFA is available as an optional Amazon EC2 networking feature that you can enable on C5n.9xl, C5n.18xl, and P3dn.24xl instances. Additional instance types will be supported in the coming months.)

/

EC2 – Memory Optimized Example

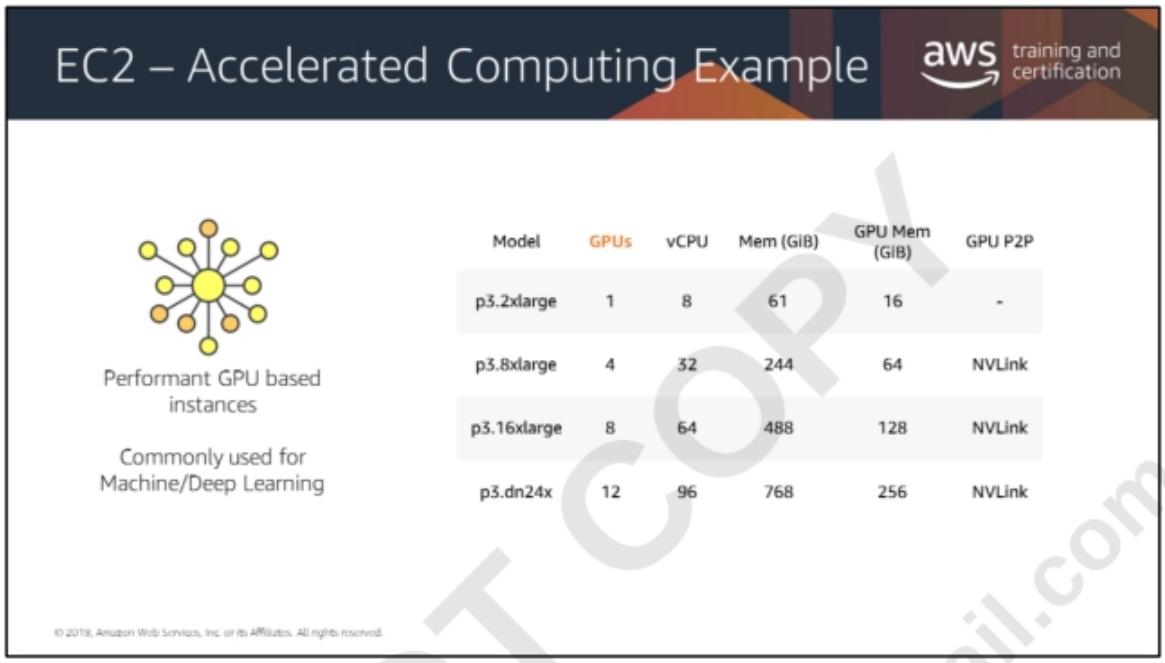
aws training and certification

Memory heavy applications or when you need more RAM than CPU

Model	vCPU	Mem (GiB)	Storage (GiB)	Dedicated EBS Bandwidth (Mbps)	Networking Performance (Gbps)
r5.large	2	16	EBS-Only	up to 3,500	Up to 10
r5.xlarge	4	32	EBS-Only	up to 3,500	Up to 10
r5.2xlarge	8	64	EBS-Only	up to 3,500	Up to 10
r5.4xlarge	16	128	EBS-Only	3,500	Up to 10
r5.12xlarge	48	384	EBS-Only	7,000	10
r5.24xlarge	96	768	EBS-Only	14,000	25

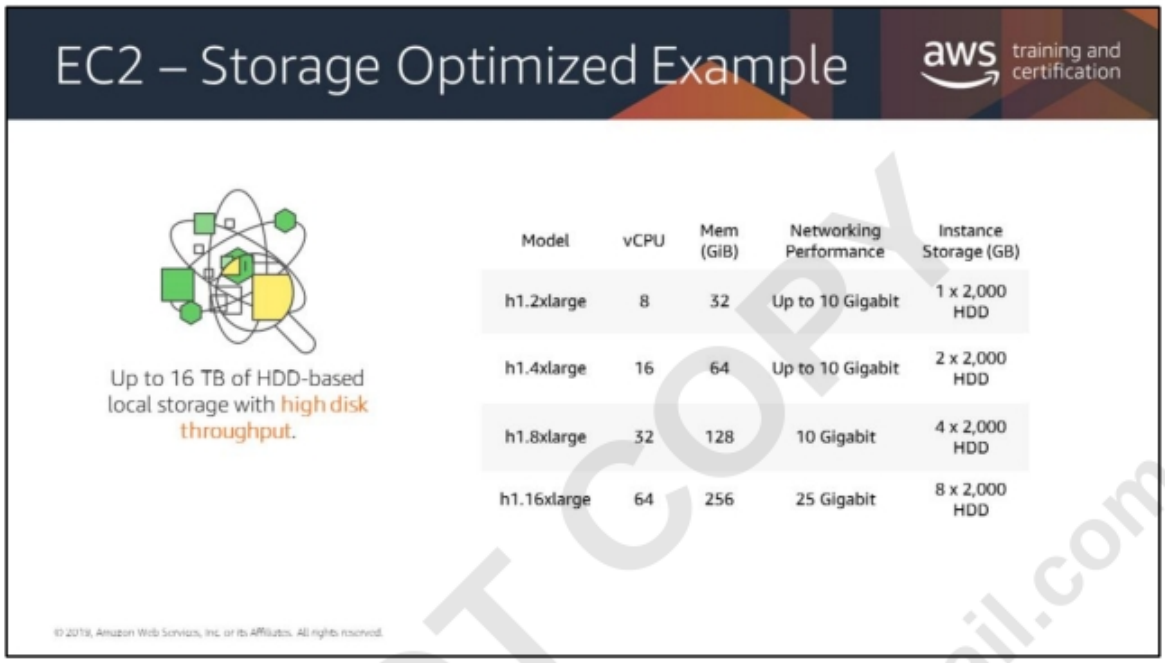
R4 instances are optimized for memory-intensive applications.

Use cases include high-performance databases, data mining and analysis, in-memory databases, distributed web scale in-memory caches, applications performing real-time processing of unstructured big data, Hadoop/Spark clusters, and other enterprise applications.



P3 instances are intended for general-purpose GPU compute applications.

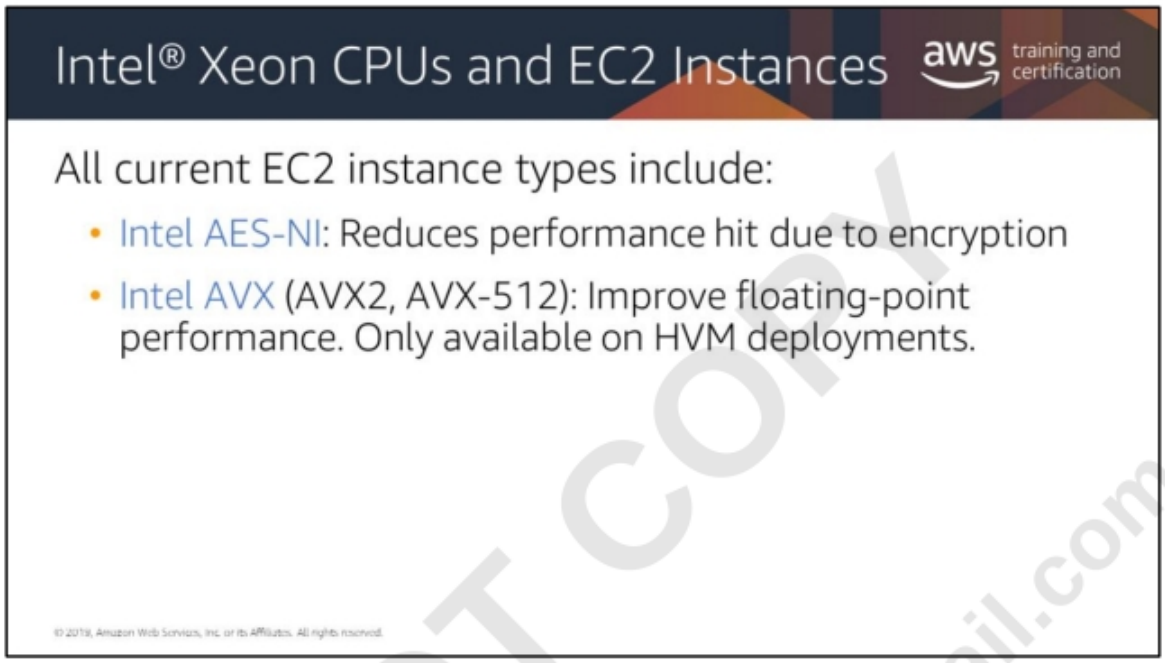
Use cases include machine learning, deep learning, high-performance computing, computational fluid dynamics, computational finance, seismic analysis, speech recognition, autonomous vehicles, and drug discovery.



H1 instances feature up to 16 TB of HDD-based local storage, deliver high disk throughput, and a balance of compute and memory.

Use cases include Amazon EMR-based workloads, distributed file systems such as HDFS and MapR-FS, network file systems, log or data processing applications such as Apache Kafka, and big data workload clusters.

/

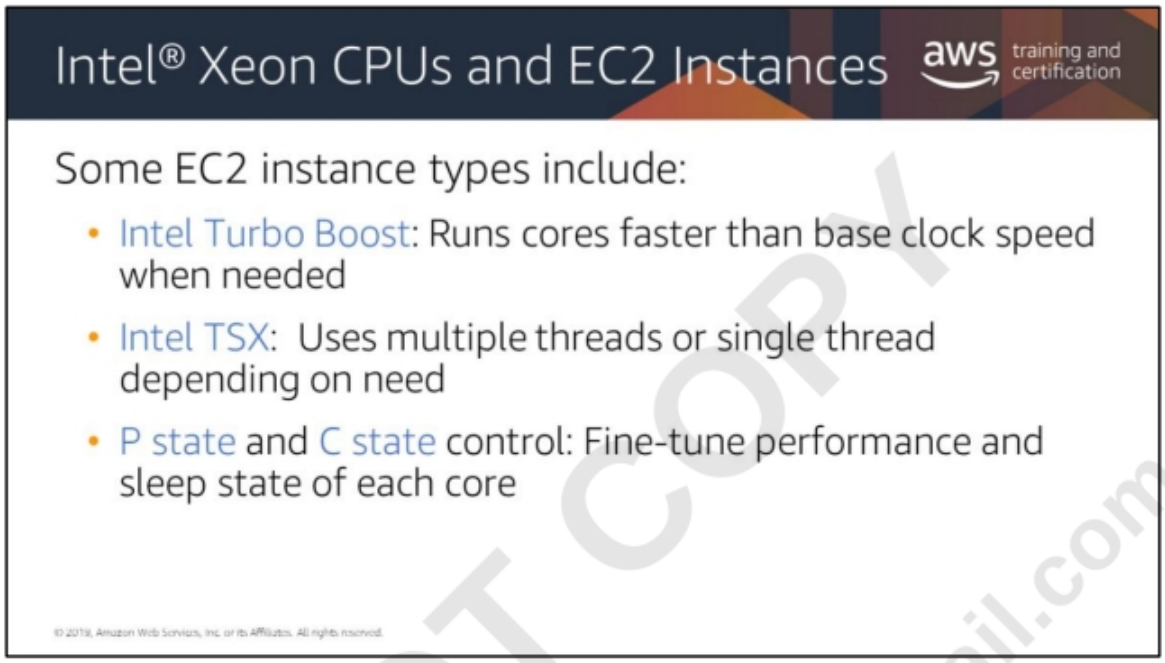


All current EC2 instance types that use Intel processors include Intel's Advanced Encryption Standard New Instructions (AES-NI), which reduces the performance hit your processor takes when you enable encryption.

All instance types also include some form of Intel Advanced Vector Extension (AVX), which is Intel's instructions custom-built for floating-point intensive workloads. AVX2 provides twice the floating point performance of AVX, and AVX-512, available only on the new Intel Xeon Scalable Processor family of CPUs, doubles the performance of AVX2.

Intel Transactional Synchronization Extensions (TSX): Provides workload optimized performance specific to the applications, multi-threaded when needed and single threaded when needed

/



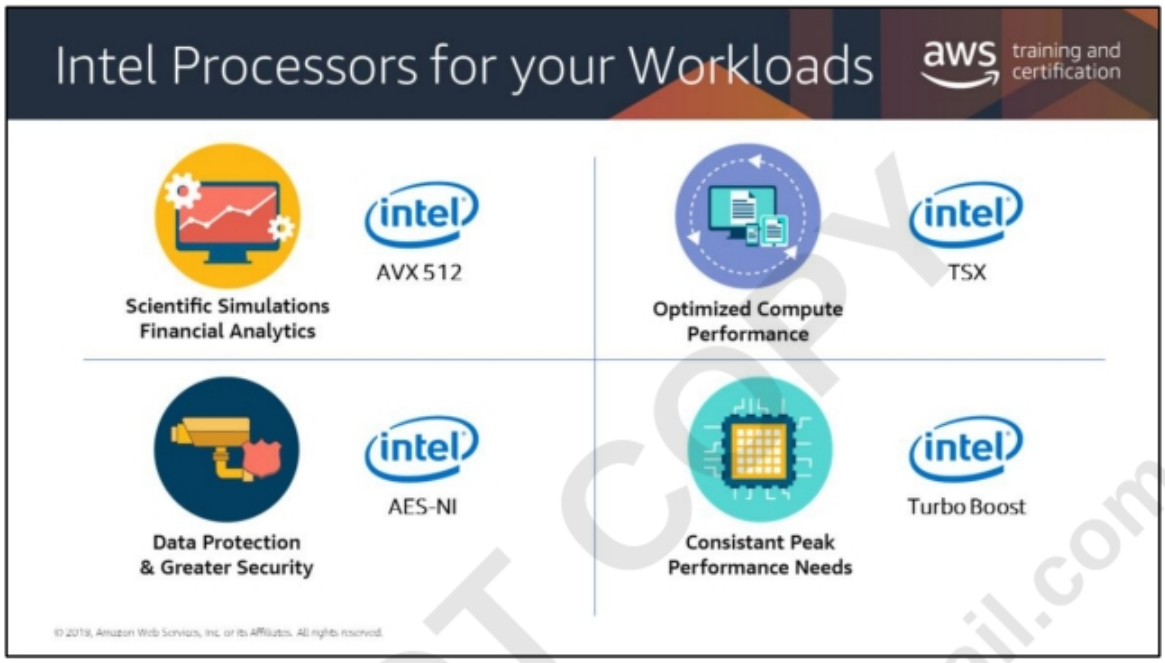
Some instance types also include Intel Turbo Boost, Intel TSX, and P State and C State control.

Intel Turbo Boost intelligently boosts the clock speed of cores based on need.

Intel Transactional Synchronization Extensions (TSX): Provides workload optimized performance specific to the applications, multi-threaded when needed and single threaded when needed.

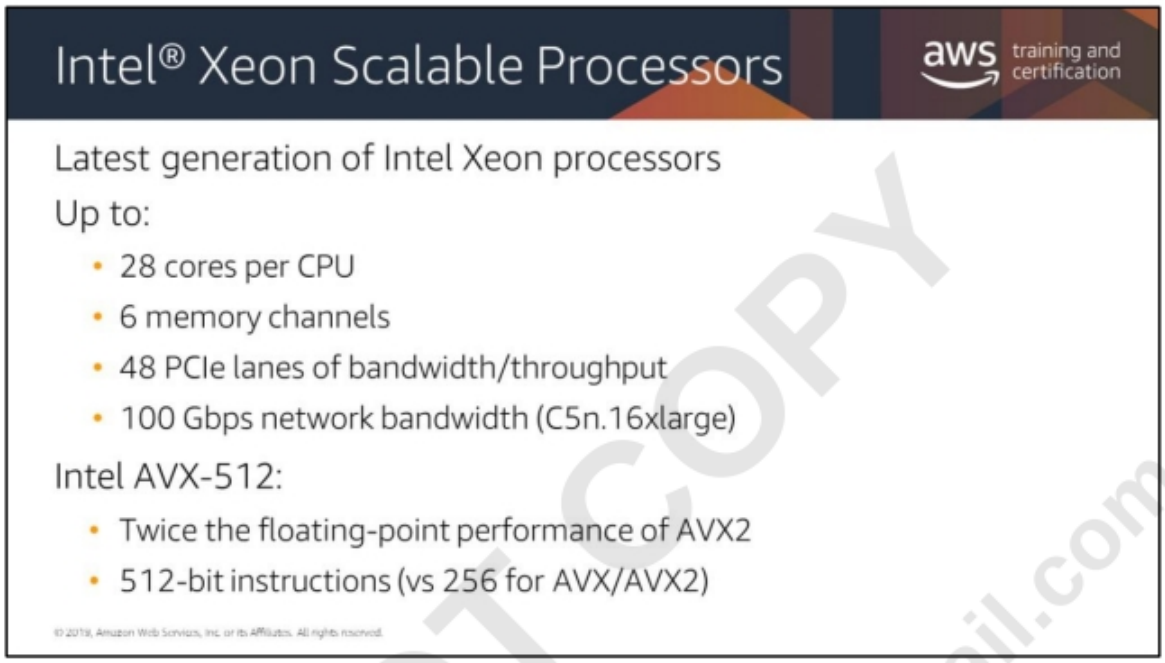
P state and C state control allows you to tune the performance and sleep state of each core to your own needs.

To find out which instance types currently support these options, see the AWS instance types page: https://aws.amazon.com/ec2/instance-types/



There are several different Intel processors to fit different workloads.

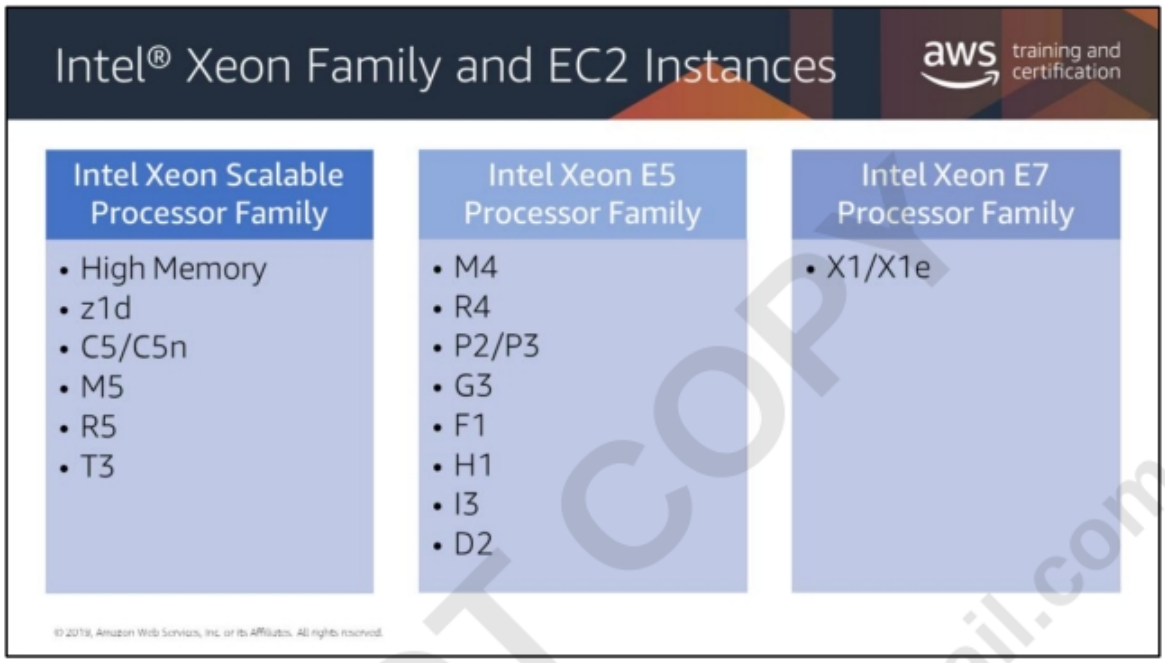
- **Intel® AVX 512:** Optimized for: scientific simulations, financial analytics, artificial intelligence (AI)/deep learning, 3D modeling and analysis, image and audio/video processing, cryptography and data compression.
- **Intel® AES-NI:** Intel® AES-NI provides faster data protection and greater security; making pervasive encryption feasible in areas where previously it was not.
- **Intel® TSX:** Intel® Transactional Synchronization Extensions (Intel® TSX) allows the processor to determine dynamically whether threads need to serialize through lock-protected critical sections, and to perform serialization only when required. Optimizing compute performance for business applications dynamically
- **Intel® Turbo Boost:** Intel® Turbo Boost Technology 2.0 accelerates processor and graphics performance for peak loads, automatically allowing processor cores to run faster than the rated operating frequency if they're operating below power, current, and temperature specification limits.



The latest generation of Intel Xeon processors is the Intel Xeon Scalable Processor Family. This group provides substantial performance improvement over the prior generation, with up to 28 cores delivering enhanced per core performance, and significant increases in memory bandwidth (6 memory channels) and I/O bandwidth and throughput (48 PCIe lanes), your most data-hungry, latency-sensitive applications such as in-memory databases and high-performance computing will see notable improvements enabled by denser compute and faster access to large data volumes.

This family also includes the latest version of Intel's AVX instructions, which double the floating point performance of processors using AVX2.

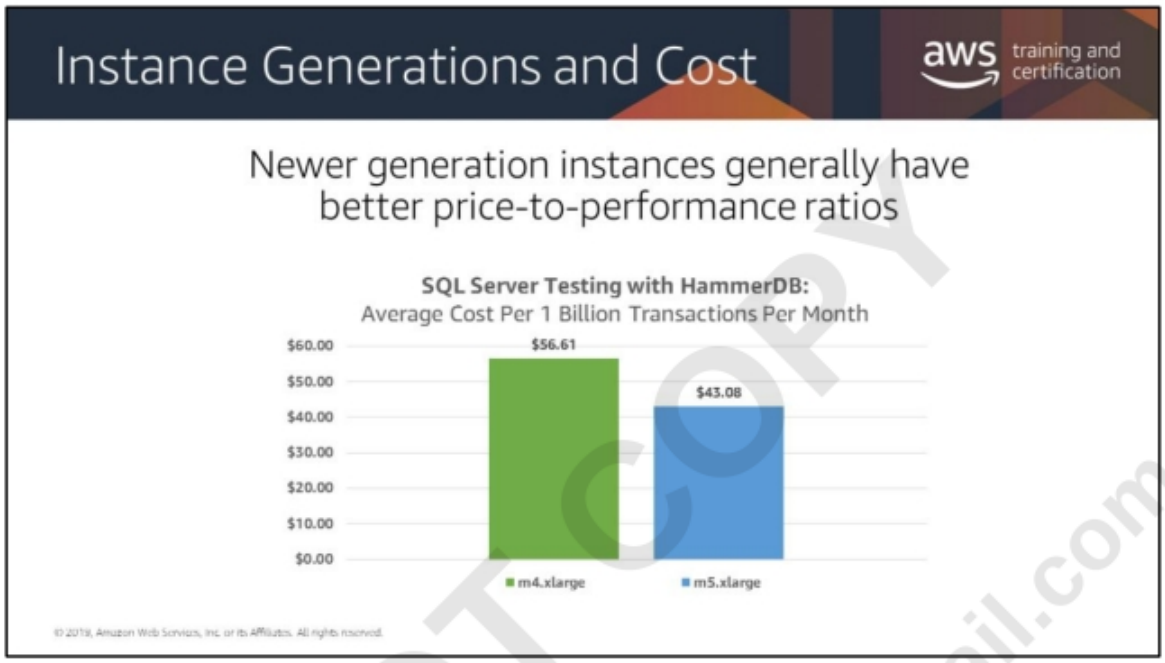
/



Here are the most current (as of March 2019) EC2 instance types and their associated processor families. For the latest list of EC2 instance types, see the AWS instance type information page:

https://aws.amazon.com/ec2/instance-types/

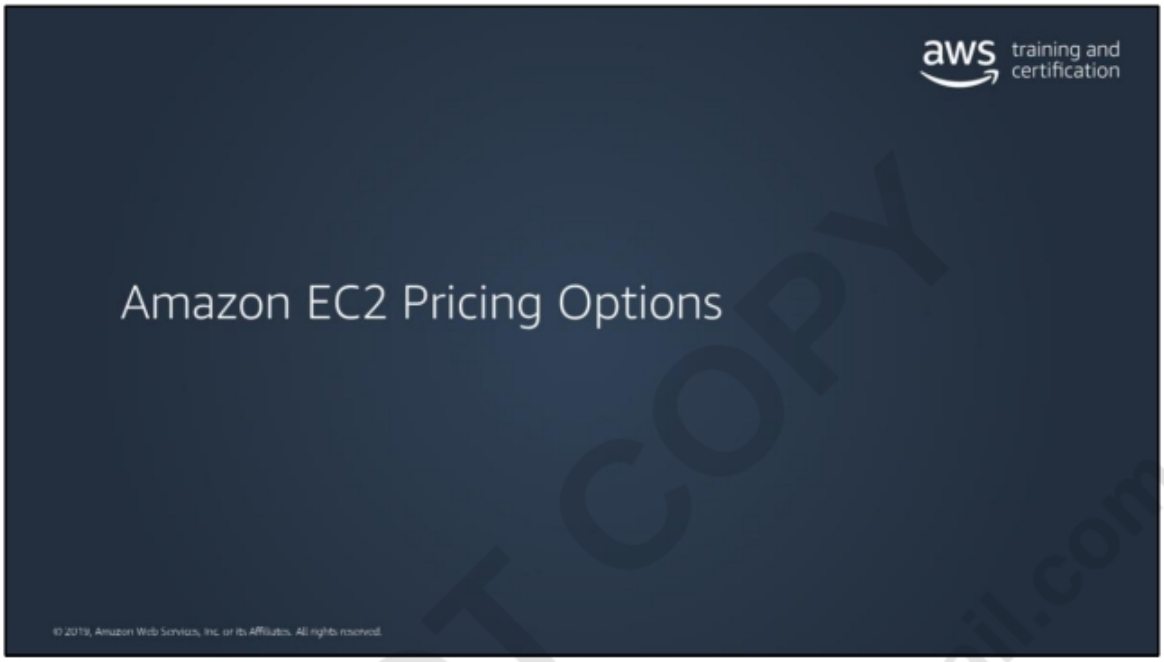
/



Even if newer generation instances cost more per hour/second than prior generation instances, the price to performance ratio of newer instances is typically higher than older generations. Here's one example of a test performed using HammerDB on several SQL Server deployments on m4.xlarge and m5.xlarge instances. In this test, they compared the number of transactions that could be performed per month by each instance based on a set number of users (from 3 to 233) with the monthly cost of operating the instances, and averaged all of the results for each instance type. You can read all the details in the link provided below.

Source: https://www.dbbest.com/blog/validating-aws-ec2-sql-server-deployments-using-benchmark-tools/

/

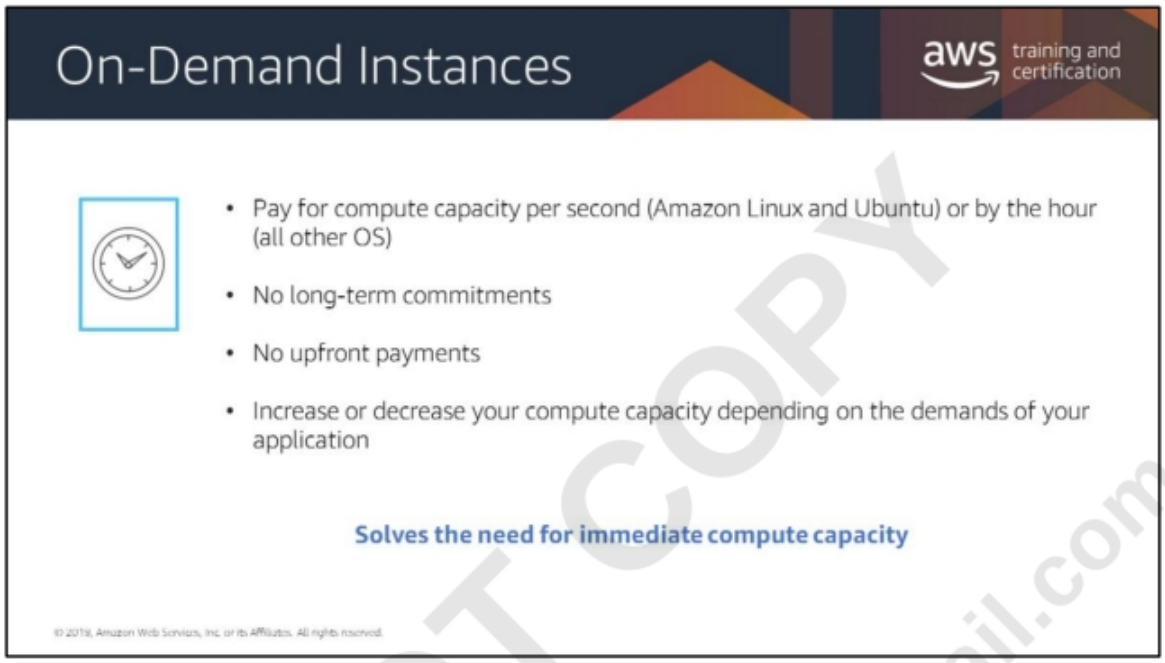


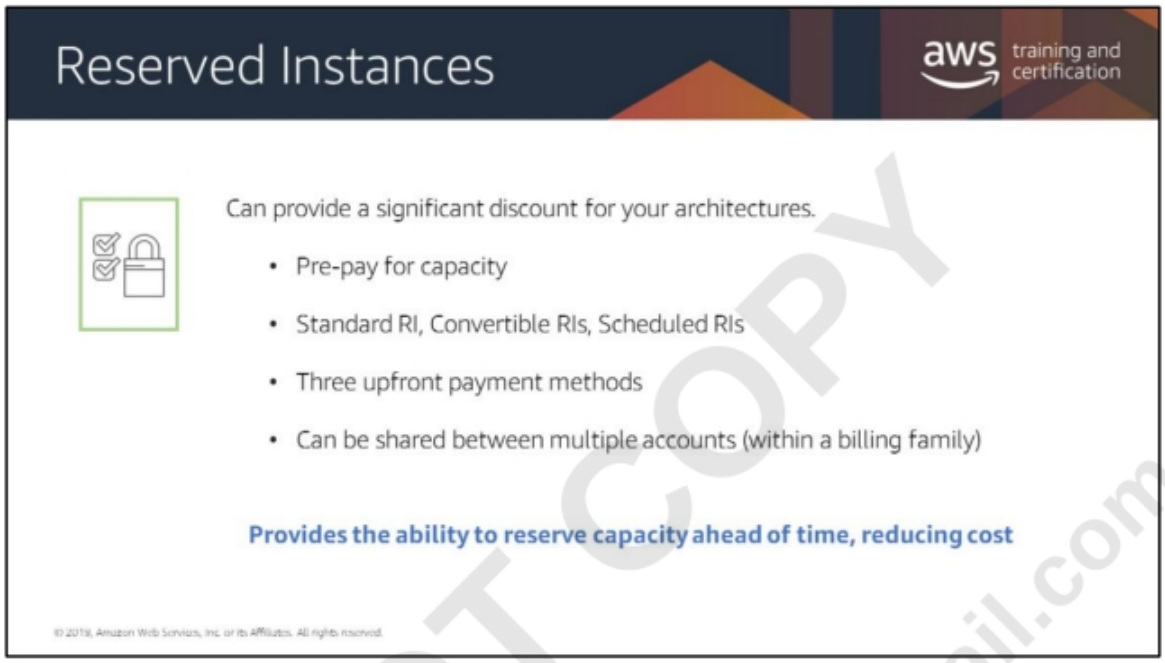
As part of the Free Tier from AWS, new AWS customers can get started with Amazon EC2 t2.micro instances, S3 bucket capacity, and many other AWS service offerings for free for up to one year after sign-up. What's available in the free tier varies from service to service. Please visit https://aws.amazon.com/free/ for details.

Amazon EC2 usage of Amazon Linux- and Ubuntu-based instances that are launched in On-Demand, Reserved and Spot form will be billed on one-second increments, with a minimum of 60 seconds. All other operating systems are billed in one-hour increments, and are billed hour forward, that is, billed at the start of the hour whether you use the full hour or not. Note that Reserved Instances are launched as, and indistinguishable from, On-Demand Instances until the bill is processed.

For more information about how AWS pricing works, see https://d0.awsstatic.com/whitepapers/aws_pricing_overview.pdf

/





Reserve Instances (RI) are a great tool to help reduce cost in your architecture. If you know what the baseline level of usage is going to be for your EC2 instances, an RI can provide significant discounts.

You can set up an RI in multiple ways:

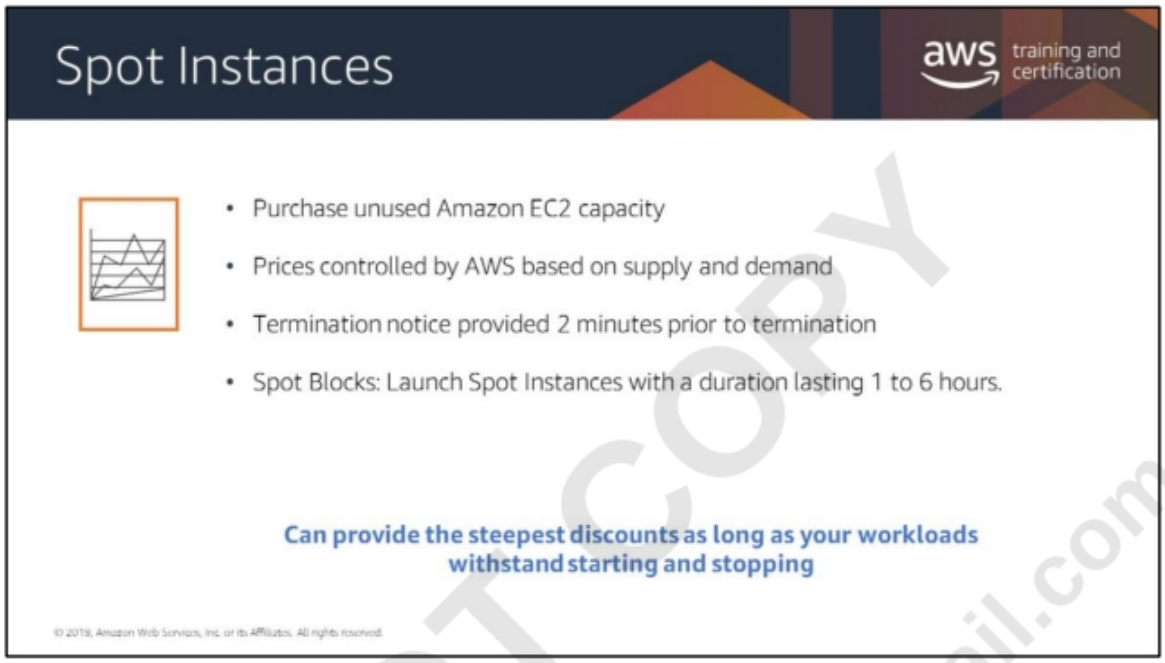
- Standard RIs: Provide the most significant discount (up to 75% off the On-Demand price) and are best suited for ready state usage
- Convertible RIs: Provide a discount (up to 54% off On-Demand price) and are able to change the attributes of the RI as long as the change results in the creation of Ris of equal or greater value
- Schedule RIs: These RIs launch in the time window of your choice, allowing you to match your capacity needs.

Term: AWS offers Standard RIs for 1-year or 3-year terms. Reserved Instance Marketplace sellers also offer RIs with shorter terms. AWS offers Convertible RIs for 1-year or 3-year terms.

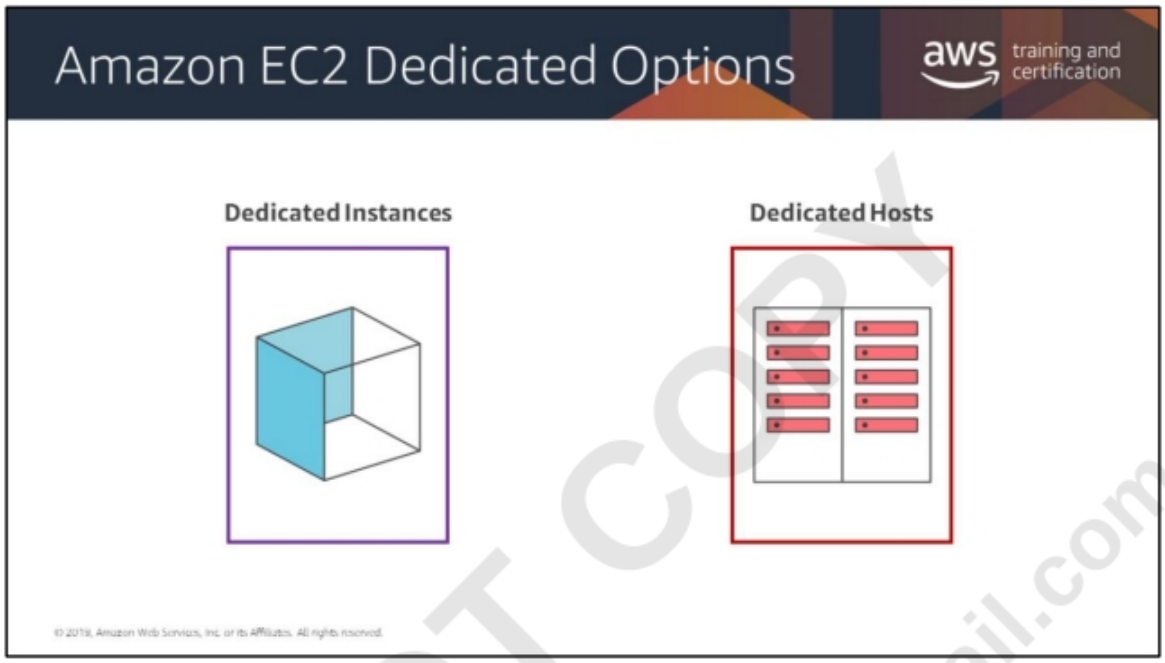
Payment option: You can choose between three payment options: All Upfront, Partial Upfront, and No Upfront. If you choose the Partial or No Upfront payment option, the remaining balance will be due in monthly increments over the term.

For more information, see https://docs.aws.amazon.com/aws-technical-content/latest/cost-optimization-reservation-models/introduction.html

/



With Amazon EC2 Spot Instances, you don't have to bid for Spot Instances in the new pricing model, and you just pay the Spot price that's in effect for the current hour for the instances that you launch. You can request Spot capacity just like you would request On-Demand capacity, without having to spend time analyzing market prices or setting a maximum bid price.

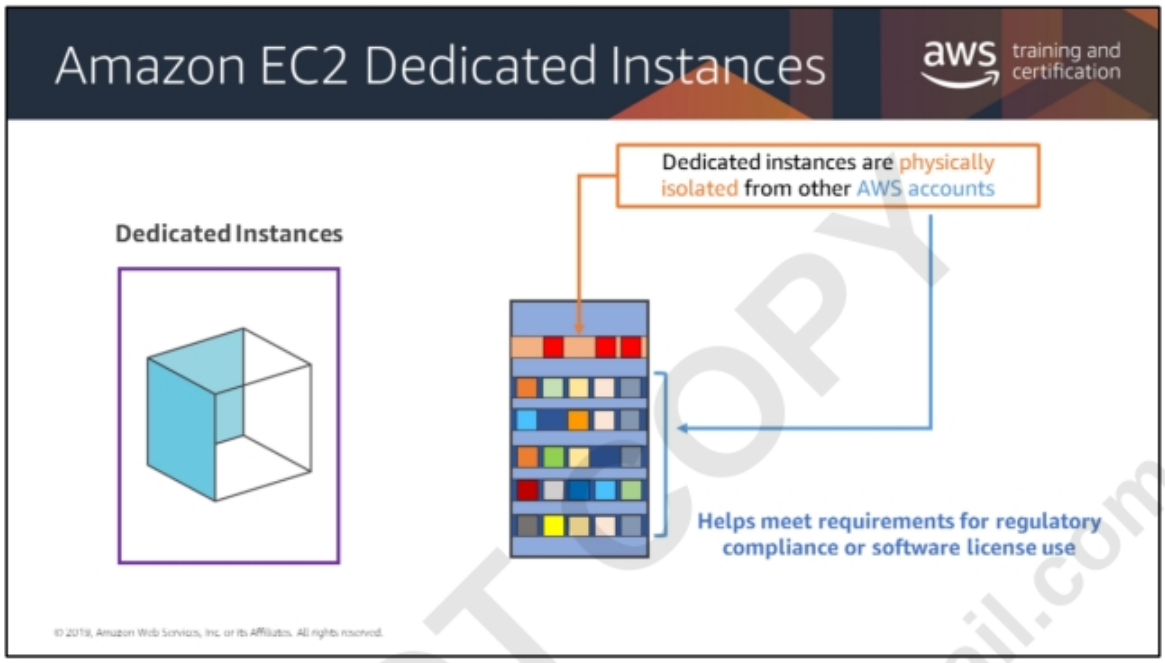


In addition to these dedicated options, you might want to consider AWS License Manger for your license requirements:

AWS License Manager makes it easier for users to manage licenses in AWS and on-premises servers from various different software vendors (Microsoft, SAP, Oracle, etc...) It will let admins create customized licensing rules when an EC2 instance gets launched, and can use these rules to limit licensing violations such as using more licenses than an agreement allows or being able to reassign licenses to different servers on a short-term basis.

Admins gain control and visibility of all their licenses with the AWS License Manager dashboard.
https://aws.amazon.com/about-aws/whats-new/2018/11/announcing-aws-license-manager/

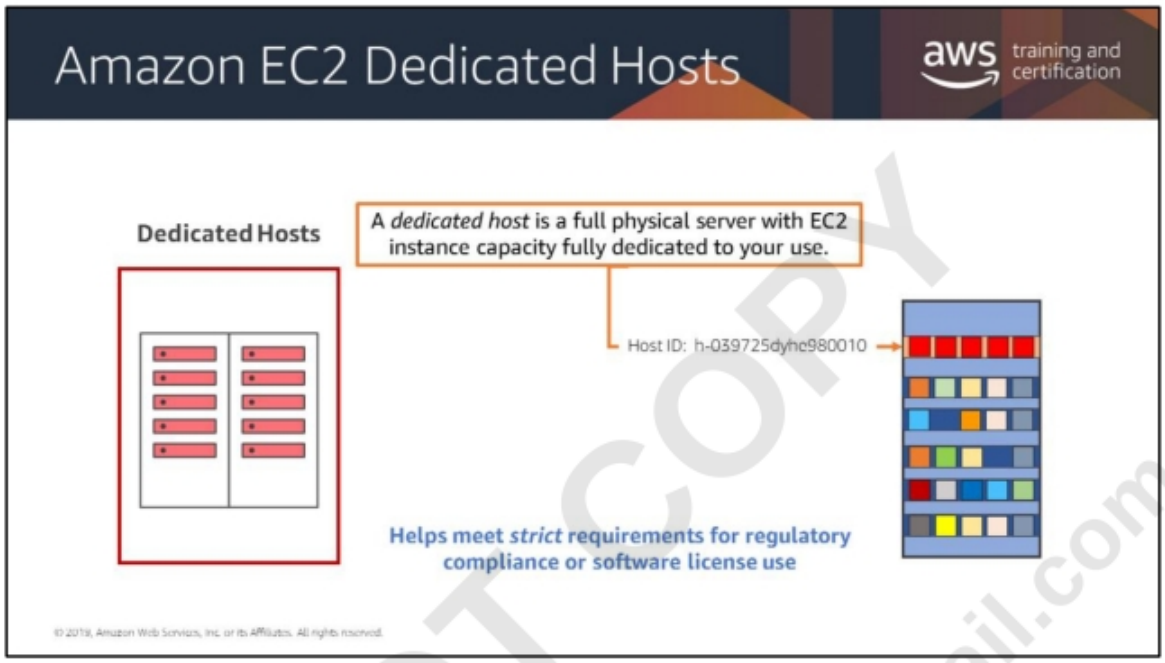
/



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 Instance pricing has two components:

- An hourly per instance usage fee
- A dedicated per-region fee (note that you pay this once per hour, regardless of how many Dedicated Instances you're running)

/

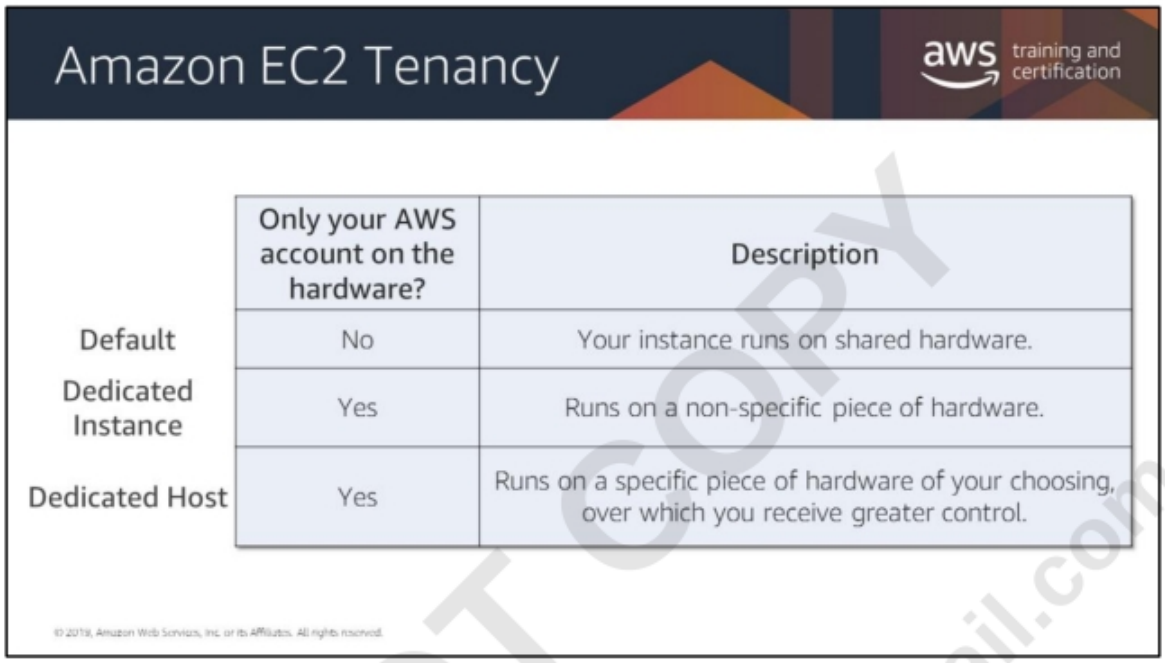


A Dedicated Host is a physical EC2 server with instance capacity fully dedicated for your use. Dedicated Hosts can help you reduce costs by allowing you to use your existing server-bound software licenses, including Windows Server, SQL Server, and SUSE Linux Enterprise Server (subject to your license terms), and can also help you meet compliance requirements. Dedicated Hosts can be purchased On-Demand (hourly). Reservations can provide up to a 70% discount compared to the On-Demand price.

Dedicated Host benefits:

- **Save money on licensing costs**: Dedicated Hosts can enable you to save money by using your own per-socket or per-core software licenses in Amazon EC2.
- **Help meet compliance and regulatory requirements**: Dedicated Hosts allow you to place your instances in a VPC on a specific, physical server. This enables you to deploy instances using configurations that help address corporate compliance and regulatory requirements

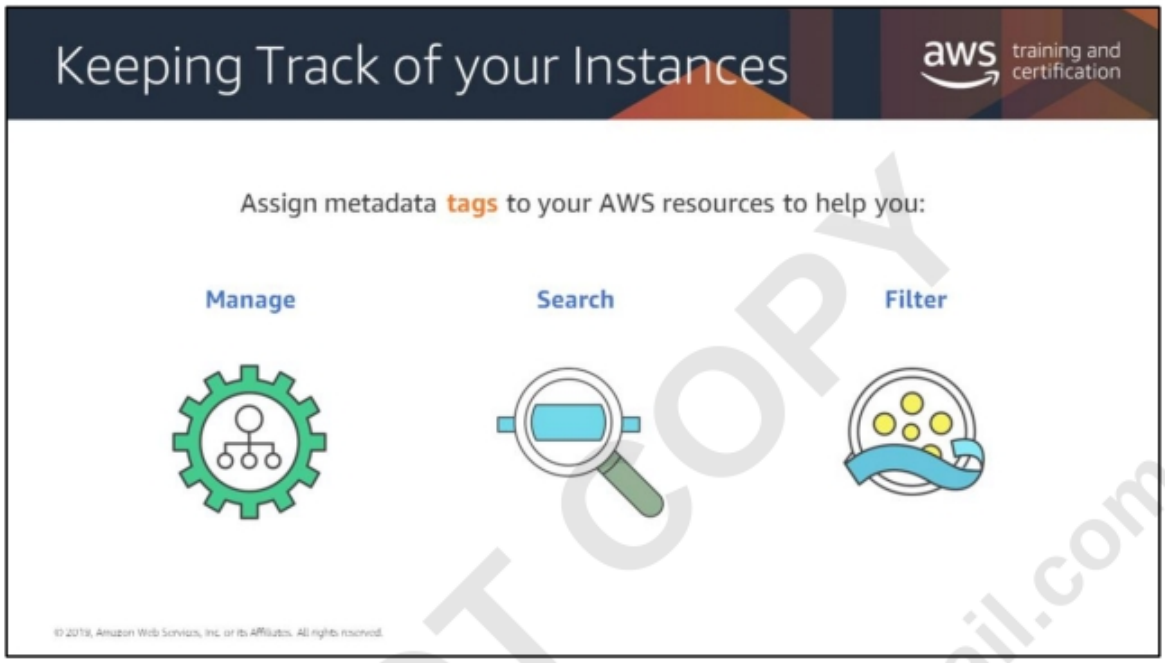
For more information about Dedicated Hosts, see https://aws.amazon.com/ec2/dedicated-hosts/



After you launch an instance, there are some limitations to changing its tenancy.

- You cannot change the tenancy of an instance from default to dedicated or host after you've launched it.
- You cannot change the tenancy of an instance from dedicated or host to default after you've launched it.
- You *can* change the tenancy of an instance from dedicated to host, or from host to dedicated, after you've launched it.

For more information, see Changing the Tenancy of an Instance.

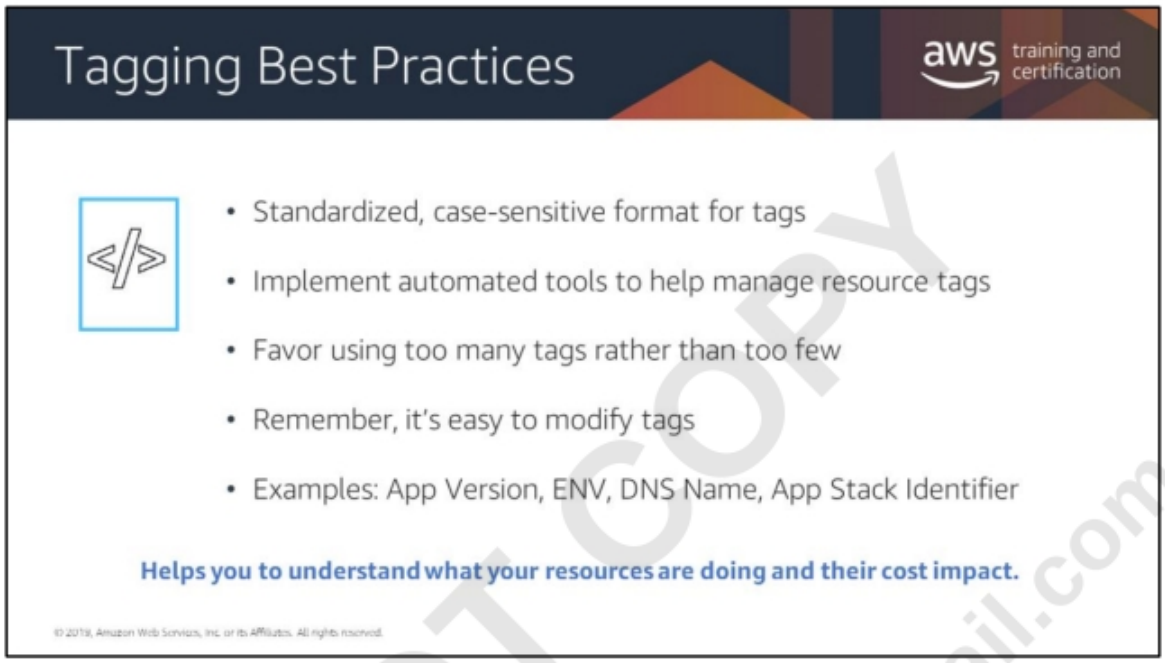


AWS allows customers to assign metadata to their AWS resources in the form of *tags*. Each tag is a simple label consisting of a customer-defined key and an optional value that can make it easier to manage, search for, and filter resources.

Although there are no inherent types of tags, they enable customers to categorize resources by purpose, owner, environment, or other criteria. This webpage describes commonly used tagging categories and strategies to help AWS customers implement a consistent and effective tagging strategy. The following sections assume basic knowledge of AWS resources, tagging, detailed billing, and IAM.

For more information about AWS tagging strategies, see https://aws.amazon.com/answers/account-management/aws-tagging-strategies/.

/

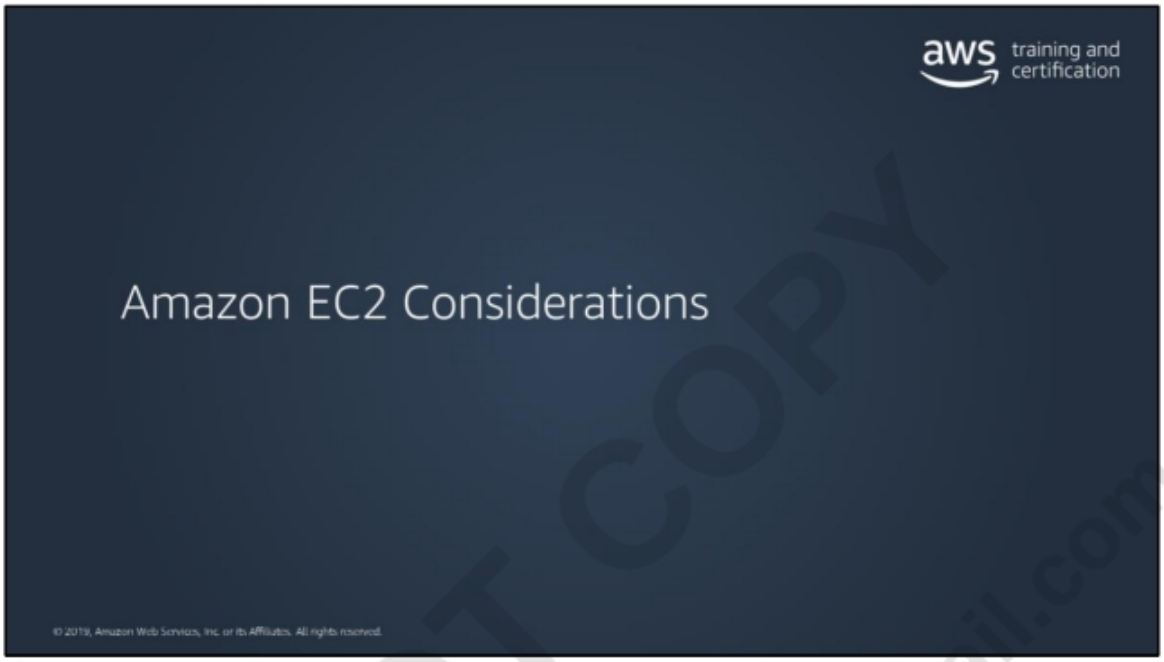


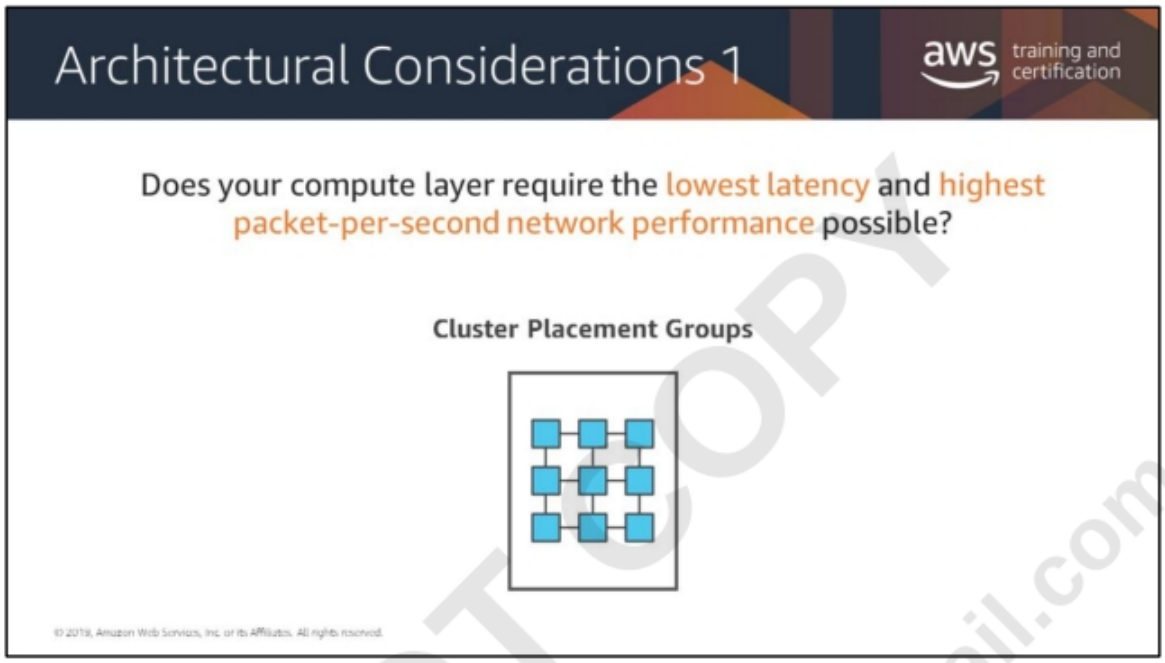
Always use a standardized, case-sensitive format for tags, and implement it consistently across all resource types.
Consider tag dimensions that support the ability to manage resource access control, cost tracking, automation, and organization.

Implement automated tools to help manage resource tags. The Resource Groups Tagging API enables programmatic control of tags, making it easier to automatically manage, search, and filter tags and resources. It also simplifies backups of tag data across all supported services with a single API call per AWS Region.
Err on the side of using too many tags rather than too few tags.

Remember that it is easy to modify tags to accommodate changing business requirements, but make sure to consider the ramifications of future changes, especially in relation to tag-based access control, automation, or upstream billing reports.

/

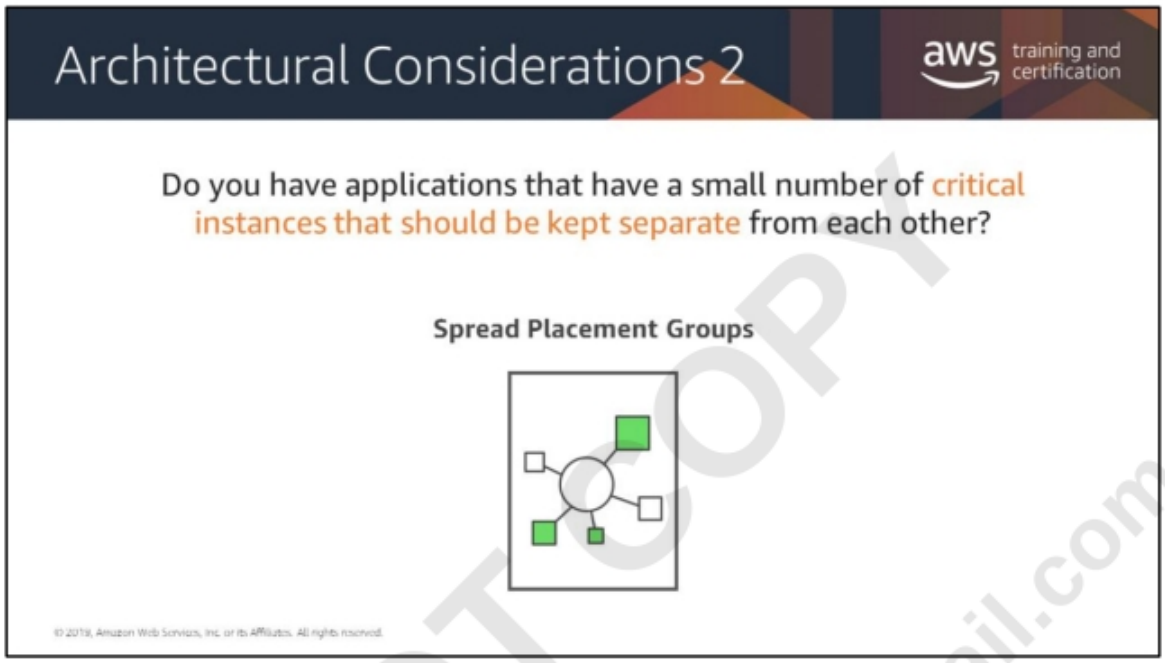




The cluster placement group is a logical grouping of instances within a single Availability Zone. This grouping provides the lowest latency and highest packet per second network performance possible.

We recommend that you launch all the instances you will need in this grouping *at one time*. If you try to add more instances into the group later, you will increase your chance of receiving an insufficient capacity error.

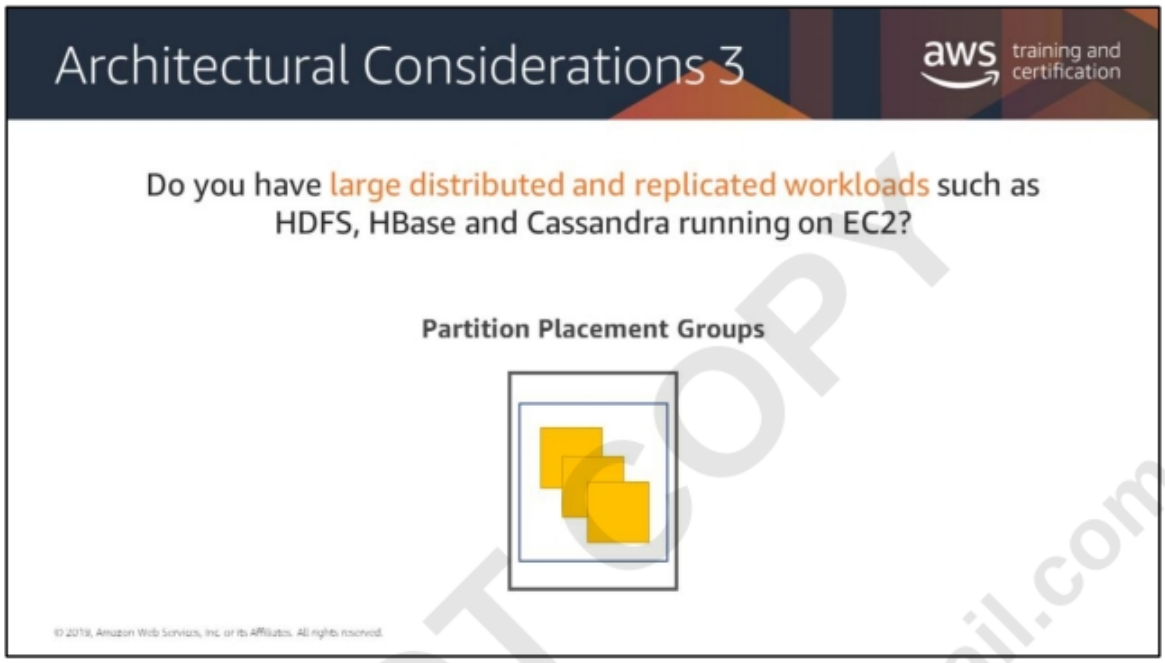
/



A *spread placement group* is a grouping of instances that are purposely positioned on distinct underlying hardware. This grouping reduces the risk of simultaneous failures that could occur if instances were sharing underlying hardware.

This type of group can span multiple Availability Zones, up to a maximum of seven instances per Availability Zone per group.

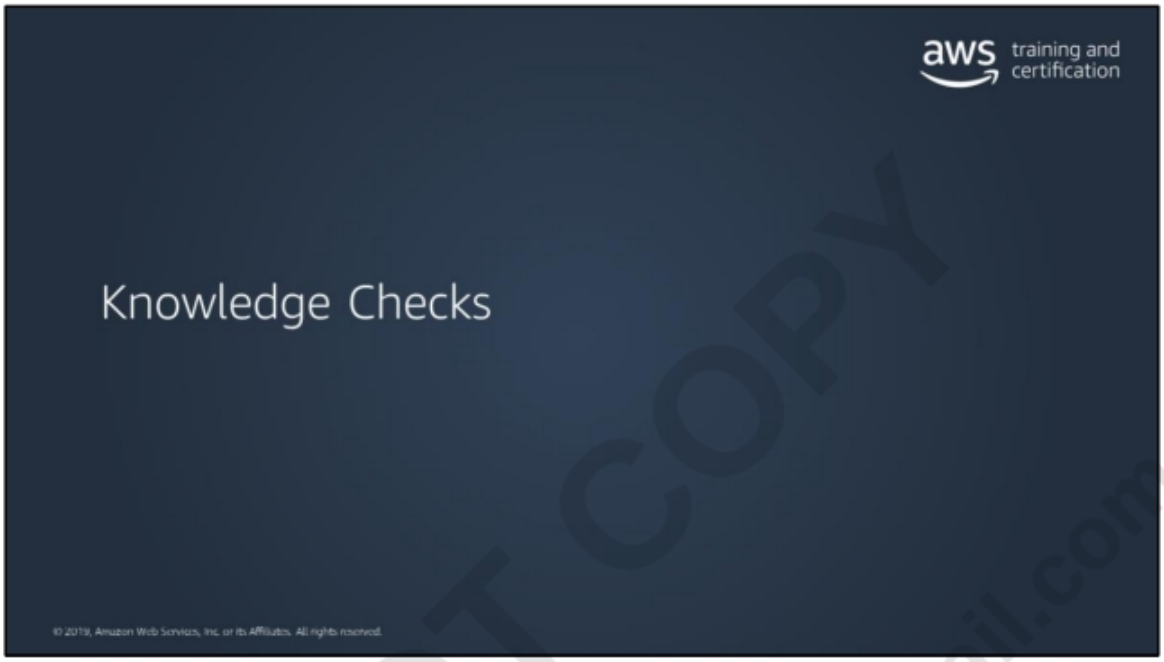
/



Partition placement groups spread EC2 instances across logical partitions and ensure that instances in different partitions do not share the same underlying hardware, thus containing the impact of hardware failure to a single partition.

In addition, partition placement groups offer visibility into the partitions and allow topology aware applications to use this information to make intelligent data replication decisions, increasing data availability and durability.

/



Knowledge Check 4

aws training and certification

What is an AMI?

1. An AMI is an object that stores data about the instance such as Local Hostname, Instance ID, or Public IP address.
2. It provides block-level storage that will disappear on instance shutdown.
3. AMIs are used to create new EC2 instances and contain a template for the root volume.
4. A type of storage bucket for Amazon S3.

/

Knowledge Check 4: Answer

aws training and certification

What is an AMI?

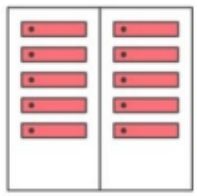
1. An AMI is an object that stores data about the instance such as Local Hostname, Instance ID, or Public IP address.
2. It provides block-level storage that will disappear on instance shutdown.
3. AMIs are used to create new EC2 instances and contain a template for the root volume.
4. A type of storage bucket for Amazon S3.

/

Knowledge Check 5

aws training and certification

If you wanted to select the host on which an instance would run, which option should you use?

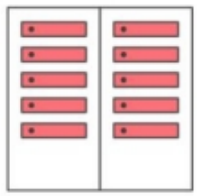


1. Default
2. Dedicated instance
3. Dedicated Host

Knowledge Check 5 : Answer

aws training and certification

If you wanted to select the host on which an instance would run, which option should you use?



1. Default
2. Dedicated instance
3. Dedicated Host

Knowledge Check 6

aws training and certification

What is Amazon EBS?

1. Object storage solution that can scale to incredible sizes to meet demand and storage requirements
2. Block storage device that can connect to multiple instances at the same time.
3. File storage system that can connect to multiple instances at the same time.
4. Block storage device that connects to one instance at a time. Can be backed up to Amazon S3.

Knowledge Check 6 : Answer

aws training and certification

What is Amazon EBS?

1. Object storage solution that can scale to incredible sizes to meet demand and storage requirements
2. Block storage device that can connect to multiple instances at the same time.
3. File storage system that can connect to multiple instances at the same time.
4. Block storage device that connects to one instance at a time. Can be backed up to Amazon S3.