

Amazon EC2 instance types

When you launch an instance, the *instance type* that you specify determines the hardware of the host computer used for your instance. Each instance type offers different compute, memory, and storage capabilities, and is grouped in an instance family based on these capabilities. Select an instance type based on the requirements of the application or software that you plan to run on your instance. For more information about features and use cases, see [Amazon EC2 Instance Types Details](#).

Amazon EC2 dedicates some resources of the host computer, such as CPU, memory, and instance storage, to a particular instance. Amazon EC2 shares other resources of the host computer, such as the network and the disk subsystem, among instances. If each instance on a host computer tries to use as much of one of these shared resources as possible, each receives an equal share of that resource. However, when a resource is underused, an instance can consume a higher share of that resource while it's available.

Each instance type provides higher or lower minimum performance from a shared resource. For example, instance types with high I/O performance have a larger allocation of shared resources. Allocating a larger share of shared resources also reduces the variance of I/O performance. For most applications, moderate I/O performance is more than enough. However, for applications that require greater or more consistent I/O performance, consider an instance type with higher I/O performance.

Contents

- Available instance types
- Hardware specifications
- Hypervisor type
- AMI virtualization types
- Processors

- [Find an Amazon EC2 instance type](#)
- [Get recommendations from EC2 instance type finder](#)
- [Get EC2 instance recommendations from Compute Optimizer](#)
- [Burstable performance instances](#)
- [Performance acceleration with GPU instances](#)
- [Amazon EC2 Mac instances](#)
- [Amazon EBS-optimized instance types](#)
- [CPU options for Amazon EC2 instances](#)
- [AMD SEV-SNP for Amazon EC2 instances](#)
- [Processor state control for Amazon EC2 Linux instances](#)

Available instance types

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instance types comprise varying combinations of CPU, memory, storage, and networking capacity and give you the flexibility to choose the appropriate mix of resources for your applications. Each instance type includes one or more instance sizes, allowing you to scale your resources to the requirements of your target workload.

Instance type naming conventions

Names are based on instance family, generation, processor family, capabilities, and size. For more information, see [Naming conventions](#) in the *Amazon EC2 Instance Types Guide*.

Find an instance type

To determine which instance types meet your requirements, such as supported Regions, compute resources, or storage resources, see [Find an Amazon EC2 instance type](#) and [Amazon EC2 instance type specifications](#) in the *Amazon EC2 Instance Types Guide*.

Hardware specifications

For detailed instance type specifications, see [Specifications](#) in the *Amazon EC2 Instance Types Guide*. For pricing information, see [Amazon EC2 On-Demand Pricing](#).

To determine which instance type best meets your needs, we recommend that you launch an instance and use your own benchmark application. Because you pay by the instance second, it's convenient and inexpensive to test multiple instance types before making a decision. If your needs change, even after you make a decision, you can change the instance type later. For more information, see [Amazon EC2 instance type changes](#).

Hypervisor type

Amazon EC2 supports the following hypervisors: Xen and Nitro.

Nitro-based instances

- General purpose: M5 | M5a | M5ad | M5d | M5dn | M5n | M5zn | M6a | M6g | M6gd | M6i | M6id | M6idn | M6in | M7a | M7g | M7gd | M7i | M7i-flex | M8g | T3 | T3a | T4g
- Compute optimized: C5 | C5a | C5ad | C5d | C5n | C6a | C6g | C6gd | C6gn | C6i | C6id | C6in | C7a | C7g | C7gd | C7gn | C7i | C7i-flex | C8g
- Memory optimized: R5 | R5a | R5ad | R5b | R5d | R5dn | R5n | R6a | R6g | R6gd | R6i | R6idn | R6in | R6id | R7a | R7g | R7gd | R7i | R7iz | R8g | U-3tb1 | U-6tb1 | U-9tb1 | U-12tb1 | U-18tb1 | U-24tb1 | U7i-6tb | U7i-8tb | U7i-12tb | U7in-16tb | U7in-24tb | U7in-32tb | U7inh-32tb | X2gd | X2idn | X2iedn | X2iezn | X8g | z1d
- Storage optimized: D3 | D3en | I3en | I4g | I4i | I7ie | I8g | Im4gn | Is4gen
- Accelerated computing: DL1 | DL2q | F2 | G4ad | G4dn | G5 | G5g | G6 | G6e | Gr6 | Inf1 | Inf2 | P3dn | P4d | P4de | P5 | P5e | P5en | Trn1 | Trn1n | Trn2 | Trn2u | VT1

- High-performance computing: Hpc6a | Hpc6id | Hpc7a | Hpc7g
- Previous generation: A1

For more information about the supported versions of Nitro hypervisor, see [Network feature support](#) in the *Amazon EC2 Instance Types Guide*.

Xen-based instances

- General purpose: M1 | M2 | M3 | M4 | T1 | T2
- Compute optimized: C1 | C3 | C4
- Memory optimized: R3 | R4 | X1 | X1e
- Storage optimized: D2 | H1 | I2 | I3
- Accelerated computing: F1 | G3 | P2 | P3

AMI virtualization types

The virtualization type of your instance is determined by the AMI that you use to launch it. Current generation instance types support hardware virtual machine (HVM) only. Some previous generation instance types support paravirtual (PV) and some AWS Regions support PV instances. For more information, see [Virtualization types](#).

For best performance, we recommend that you use an HVM AMI. In addition, HVM AMIs are required to take advantage of enhanced networking. HVM virtualization uses hardware-assist technology provided by the AWS platform. With HVM virtualization, the guest VM runs as if it were on a native hardware platform, except that it still uses PV network and storage drivers for improved performance.

Processors

EC2 instances support a variety of processors.

Processors

- Intel processors
- AMD processors
- AWS Graviton processors
- AWS Trainium
- AWS Inferentia

Intel processors

Amazon EC2 instances that run on Intel processors might include the following processor features. Not all instances that run on Intel processors support all of these processor features. For information about which features are available for each instance type, see [Amazon EC2 Instance types](#).

- **Intel AES New Instructions (AES-NI)** — Intel AES-NI encryption instruction set improves upon the original Advanced Encryption Standard (AES) algorithm to provide faster data protection and greater security. All current generation EC2 instances support this processor feature.
- **Intel Advanced Vector Extensions (Intel AVX, Intel AVX2, and Intel AVX-512)** — Intel AVX and Intel AVX2 are 256-bit, and Intel AVX-512 is a 512-bit instruction set extension designed for applications that are Floating Point (FP) intensive. Intel AVX instructions improve performance for applications like image and audio/video processing, scientific simulations, financial analytics, and 3D modeling and analysis. These features are only available on instances launched with HVM AMIs.
- **Intel Turbo Boost Technology** — Intel Turbo Boost Technology processors automatically run cores faster than the base operating

frequency.

- **Intel Deep Learning Boost (Intel DL Boost)** — Accelerates AI deep learning use cases. The 2nd Gen Intel Xeon Scalable processors extend Intel AVX-512 with a new Vector Neural Network Instruction (VNNI/INT8) that significantly increases deep learning inference performance over previous generation Intel Xeon Scalable processors (with FP32) for image recognition/segmentation, object detection, speech recognition, language translation, recommendation systems, reinforcement learning, and more. VNNI may not be compatible with all Linux distributions.

The following instances support VNNI: M5n, R5n, M5dn, M5zn, R5b, R5dn, D3, D3en, and C6i. C5 and C5d instances support VNNI for only 12xlarge, 24xlarge, and metal instances.

Confusion can result from industry naming conventions for 64-bit CPUs. Chip manufacturer Advanced Micro Devices (AMD) introduced the first commercially successful 64-bit architecture based on the Intel x86 instruction set. Consequently, the architecture is widely referred to as AMD64 regardless of the chip manufacturer. Windows and several Linux distributions follow this practice. This explains why the internal system information on an instance running Ubuntu or Windows displays the CPU architecture as AMD64 even though the instances are running on Intel hardware.

AMD processors

Amazon EC2 instances that run on [AMD EPYC](#) processors can help you optimize both cost and performance for your workloads. These instances might support the following processor features. Not all instances that run on AMD processors support all of these processor features. For information about which features are available for each instance type, see [Amazon EC2](#)

[Instance types.](#)

- AMD Secure Memory Encryption (SME)
- AMD Transparent Single Key Memory Encryption (TSME)
- AMD Advanced Vector Extensions (AVX)
- AMD Secure Encrypted Virtualization-Secure Nested Paging ([SEV-SNP](#))
- Vector Neural Network Instructions (VNNI)
- BFloat16

AWS Graviton processors

[AWS Graviton](#) is a family of processors designed to deliver the best price performance for your workloads running on Amazon EC2 instances.

For more information, see [Getting started with Graviton](#).

AWS Trainium

Instances powered by [AWS Trainium](#) are purpose built for high-performance, cost-effective deep learning training. You can use these instances to train natural language processing, computer vision, and recommender models used across a broad set of applications, such as speech recognition, recommendation, fraud detection, and image and video classification. Use your existing workflows in popular ML frameworks, such as PyTorch and TensorFlow.

AWS Inferentia

Instances powered by [AWS Inferentia](#) are designed to accelerate machine learning. They provide high performance and low latency machine learning

inference. These instances are optimized for deploying deep learning (DL) models for applications, such as natural language processing, object detection and classification, content personalization and filtering, and speech recognition.

There are a variety of ways that you can get started:

- Use SageMaker AI, a fully-managed service that is the easiest way to get started with machine learning models. For more information, see [Get Started with SageMaker AI](#) in the *Amazon SageMaker AI Developer Guide*.
- Launch an Inf1 or Inf2 instance using the Deep Learning AMI. For more information, see [AWS Inferentia with DLAMI](#) in the *AWS Deep Learning AMIs Developer Guide*.
- Launch an Inf1 or Inf2 instance using your own AMI and install the [AWS Neuron SDK](#), which enables you to compile, run, and profile deep learning models for AWS Inferentia.
- Launch a container instance using an Inf1 or Inf2 instance and an Amazon ECS-optimized AMI. For more information, see [Amazon Linux 2 \(Inferentia\) AMIs](#) in the *Amazon Elastic Container Service Developer Guide*.
- Create an Amazon EKS cluster with nodes running Inf1 instances. For more information, see [Inferentia support](#) in the *Amazon EKS User Guide*.