

EC2 Instances

Instance:

An instance is a virtual server in the cloud. A cloud instance allows software developers to scale beyond traditional physical boundaries. There are two main benefits of cloud instances.

1. Scalability - Developers can horizontally scale cloud resources by increasing the CPU, memory, storage, and network resources to the particular instance.
2. Fault tolerance - Organizations create redundancy by using multiple duplicate instances for backup.

Its configuration at launch is a copy of the AMI that you specified when you launched the instance. You can launch different types of instances from a single AMI. An instance type essentially determines the hardware of the host computer used for your instance. Each instance type offers different compute and memory capabilities. Select an instance type based on the amount of memory and computing power that you need for the application or software that you plan to run on the instance.

Steps to launch an instance:

1. Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. In the navigation bar at the top of the screen, the current AWS Region is displayed (for example, US East (Ohio)). Select a Region in which to launch the instance. This choice is important because some Amazon EC2 resources can be shared between Regions, while others can't.
3. From the Amazon EC2 console dashboard, choose Launch instance.
4. (Optional) Under Name and tags, for Name, enter a descriptive name for your instance.
5. Under Application and OS Images (Amazon Machine Image), choose Quick Start, and then choose the operating system (OS) for your instance.
6. Under Key pair (login), for Key pair name, choose an existing key pair or create a new one.
7. In the Summary panel, choose Launch instance.

Security:

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that are built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The shared responsibility model describes this as security of the cloud and security in the cloud:

- Security of the cloud – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS Compliance Programs. To learn about the compliance programs that apply to Amazon EC2, see [AWS Services in Scope by Compliance Program](#).
- Security in the cloud – Your responsibility includes the following areas:
 - Controlling network access to your instances, for example, through configuring your VPC and security groups. For more information, see [Controlling network traffic](#).
 - Managing the credentials used to connect to your instances.
 - Managing the guest operating system and software deployed to the guest operating system, including updates and security patches. For more information, see [Update management in Amazon EC2](#).
 - Configuring the IAM roles that are attached to the instance and the permissions associated with those roles. For more information, see [IAM roles for Amazon EC2](#).

Monitoring:

Monitoring is an important part of maintaining the reliability, availability, and performance of your Amazon Elastic Compute Cloud (Amazon EC2) instances and your AWS solutions. You should collect monitoring data from all of the parts in your AWS solutions so that you can more easily debug a multi-point failure if one occurs.

System status checks – monitor the AWS systems required to use your instance to ensure that they are working properly. These checks detect problems with your instance that require AWS involvement to repair. When a system status check fails, you can choose to wait for AWS to fix the issue or you can resolve it yourself (for example, by stopping and restarting or terminating and replacing an instance). Examples of problems that cause system status checks to fail include:

- Loss of network connectivity
- Loss of system power
- Software issues on the physical host
- Hardware issues on the physical host that impact network reachability

Instance status checks – monitor the software and network configuration of your individual instance. These checks detect problems that require your involvement to repair. When an instance status check fails, typically you will need to address the problem yourself (for

example, by rebooting the instance or by making modifications in your operating system).

Examples of problems that may cause instance status checks to fail include:

- Failed system status checks
- Misconfigured networking or start-up configuration
- Exhausted memory
- Corrupted file system
- Incompatible kernel

Output:

1. Screenshot of Launched EC2 instance-

The screenshot displays the AWS Management Console interface for an EC2 instance. The top navigation bar includes the AWS logo, a search bar, and the user's location (N. Virginia) and name (SanikaMhatre). The left sidebar shows the EC2 Dashboard and various navigation options like EC2 Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, Images, AMIs, AMI Catalog, and Elastic Block Store. The main content area shows the 'Instance summary for i-0a892d45bcf19cd8e (expt1_33)'. The instance is in the 'Running' state. The summary includes details such as Instance ID, Public IPv4 address (54.89.153.104), Private IPv4 address (172.31.38.113), Instance state (Running), Hostname type (IP name: ip-172-31-38-113.ec2.internal), Private IP DNS name (ip-172-31-38-113.ec2.internal), Instance type (t2.micro), VPC ID (vpc-0aea0460abe9d525b), Subnet ID (subnet-080520df095b5638c), and IAM Role (Optional). The console also shows a warning for AWS Compute Optimizer finding: 'Opt-in to AWS Compute Optimizer for recommendations. | Learn more'.

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0a892d45bcf19cd8e (expt1_33)	54.89.153.104 open address	172.31.38.113

IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-54-89-153-104.compute-1.amazonaws.com open address

Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-172-31-38-113.ec2.internal	ip-172-31-38-113.ec2.internal	-

Answer private resource DNS name	Instance type	AWS Compute Optimizer finding
IPv4 (A)	t2.micro	Opt-in to AWS Compute Optimizer for recommendations. Learn more

Auto-assigned IP address	VPC ID	Auto Scaling Group name
54.89.153.104 [Public IP]	vpc-0aea0460abe9d525b	-

IAM Role	Subnet ID
Optional	subnet-080520df095b5638c

2.Screenshot of Security details of the launched EC2 instance.

The screenshot displays the AWS Management Console interface for an EC2 instance. The left sidebar shows the navigation menu with options like EC2 Dashboard, EC2 Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, Images, AMIs, AMI Catalog, and Elastic Block Store. The main content area is titled 'Security' and shows the following details:

- Security details:**
 - IAM Role: -
 - Owner ID: 371252621001
 - Launch time: Mon Aug 28 20:31:44 GMT+0530 (India Standard Time)
 - Security groups: sg-03c07798b4562dc91 (launch-wizard-4)
- Inbound rules:**

Name	Security group rule ID	Port range	Protocol	Source	Security groups
-	sgr-0d71f5ed8dfcbe4f	22	TCP	0.0.0.0/0	launch-wizard-4
- Outbound rules:**

Name	Security group rule ID	Port range	Protocol	Destination	Security groups
-	sgr-00339168e8ec315bd	All	All	0.0.0.0/0	launch-wizard-4

3.Screenshot of monitoring tab of launched instance-

The screenshot displays the AWS Management Console interface for an EC2 instance, specifically the 'Monitoring' tab. The left sidebar is the same as in the previous screenshot. The main content area shows various performance metrics for the instance, with a time range of 1h selected. The metrics are displayed in a grid of charts:

- CPU utilization (%):** A line chart showing CPU usage over time, with a peak of 0.333%.
- Status check failed (any):** A line chart showing the number of failed status checks, with a value of 0.5.
- Status check failed (instance):** A line chart showing the number of failed instance status checks, with a value of 0.5.
- Status check failed (system):** A line chart showing the number of failed system status checks, with a value of 0.5.
- Network in (bytes):** A line chart showing network input, with a peak of 844 bytes.
- Network out (bytes):** A line chart showing network output, with a peak of 818 bytes.
- Network packets in (count):** A line chart showing network input packets, with a peak of 11.
- Network packets out (count):** A line chart showing network output packets, with a peak of 13.
- Disk reads (bytes):** A line chart showing disk read operations, with a peak of 1.
- Disk read operations (count):** A line chart showing disk read operations, with a peak of 1.
- Disk writes (bytes):** A line chart showing disk write operations, with a peak of 1.
- Disk write operations (count):** A line chart showing disk write operations, with a peak of 1.

4. Screenshot of MobaXterm window – Connected the launched instance-

