

CREATING A WINDOWS EC2 INSTANCE AND CONNECT IT FROM A WINDOWS OPERATING SYSTEM



-Sooriya

Step 1: Create a Name for Your Instance

Naming your instance helps you easily identify and manage multiple instances within your AWS environment.

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name

[Add additional tags](#)

Step 2: Choose an AMI (Windows, Free Tier eligible)

An AMI (Amazon Machine Image) provides a pre-configured operating system and application software, simplifying deployment. I selected a Windows AMI that is eligible under the Free Tier.

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q

Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Li

SUS

Q

[Browse more AMIs](#)

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Microsoft Windows Server 2022 Base

ami-0888db1202897905c (64-bit (x86))

Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible ▼

Description

Microsoft Windows 2022 Datacenter edition. [English]

Architecture

AMI ID

Username

64-bit (x86)

ami-0888db1202897905c

root

Verified provider

Step 3: Choose an Instance Type (Free Tier eligible)

The instance type determines the resources (CPU, memory) allocated to your instance. I chose t2.micro, which is free-tier eligible and suitable for low-traffic applications.

▼ Instance type [Info](#) | [Get advice](#)

Instance type

t2.micro Free tier eligible

Family: t2 1 vCPU 1 GiB Memory Current generation: true

On-Demand Windows base pricing: 0.0162 USD per Hour

On-Demand SUSE base pricing: 0.0116 USD per Hour

On-Demand RHEL base pricing: 0.026 USD per Hour

On-Demand Linux base pricing: 0.0116 USD per Hour

☐ All generations

[Compare instance types](#)

Additional costs apply for AMIs with pre-installed software


▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

windows

↕

 [Create new key pair](#)

For Windows instances, you use a key pair to decrypt the administrator password. You then use the decrypted password to connect to your instance.

Step 4: Create a New Key Pair

A key pair ensures secure login using public-private encryption. The private key (.pem file) is required to access the instance securely using RDP (Remote Desktop Protocol).

Create key pair

Key pair name

Key pairs allow you to connect to your instance securely.

Enter key pair name

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type



☒ RSA
RSA encrypted private and public key pair

☐ ED25519
ED25519 encrypted private and public key pair (Not supported for Windows instances)

Private key file format

☒ .pem
For use with OpenSSH

☐ .ppk
For use with PuTTY

 When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** [Learn more](#) 

Cancel

Create key pair

Step 5: Set the Network Settings

Network settings configure how your instance connects to the internet and other resources. I used a Virtual Private Cloud (VPC) with a public IP to allow remote access.

▼ Network settings [Info](#)

VPC - *required* | [Info](#)

vpc-0289d6374d79fa1e8
172.31.0.0/16

(default) ▼

↻

Subnet | [Info](#)

No preference ▼

↻ [Create new subnet](#) [↗](#)

Auto-assign public IP | [Info](#)

Enable ▼

Additional charges apply when outside of free tier allowance

Step 6: Security Group

Security groups act as a virtual firewall for your instance. I used to RDP access via port 3389, ensuring secure connectivity while blocking unnecessary traffic.

Firewall (security groups) | [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

Security group name - *required*

windows

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-:/()#,@[]+=&;{}!\$*

Description - *required* | [Info](#)

launch-wizard-1 created 2024-10-08T03:01:50.782Z

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 3389, 0.0.0.0/0)

Remove

Type | [Info](#)

rdp ▼

Protocol | [Info](#)

TCP

Port range | [Info](#)

3389

Source type | [Info](#)

Anywhere ▼

Source | [Info](#)

🔍 Add CIDR, prefix list or security

0.0.0.0/0 ✕

Description - *optional* | [Info](#)

e.g. SSH for admin desktop

Add security group rule

Step 7: Configure Storage (30GB for Windows)

AWS uses Elastic Block Store (EBS) for storage. I configured 30GB, which is typical for Windows instances and falls within Free Tier limits.

▼ **Configure storage** [Info](#)

Advanced

1x GiB ▼ Root volume (Not encrypted)

ⓘ

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

×

Add new volume

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

🕒

Click refresh to view backup information

The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

↻

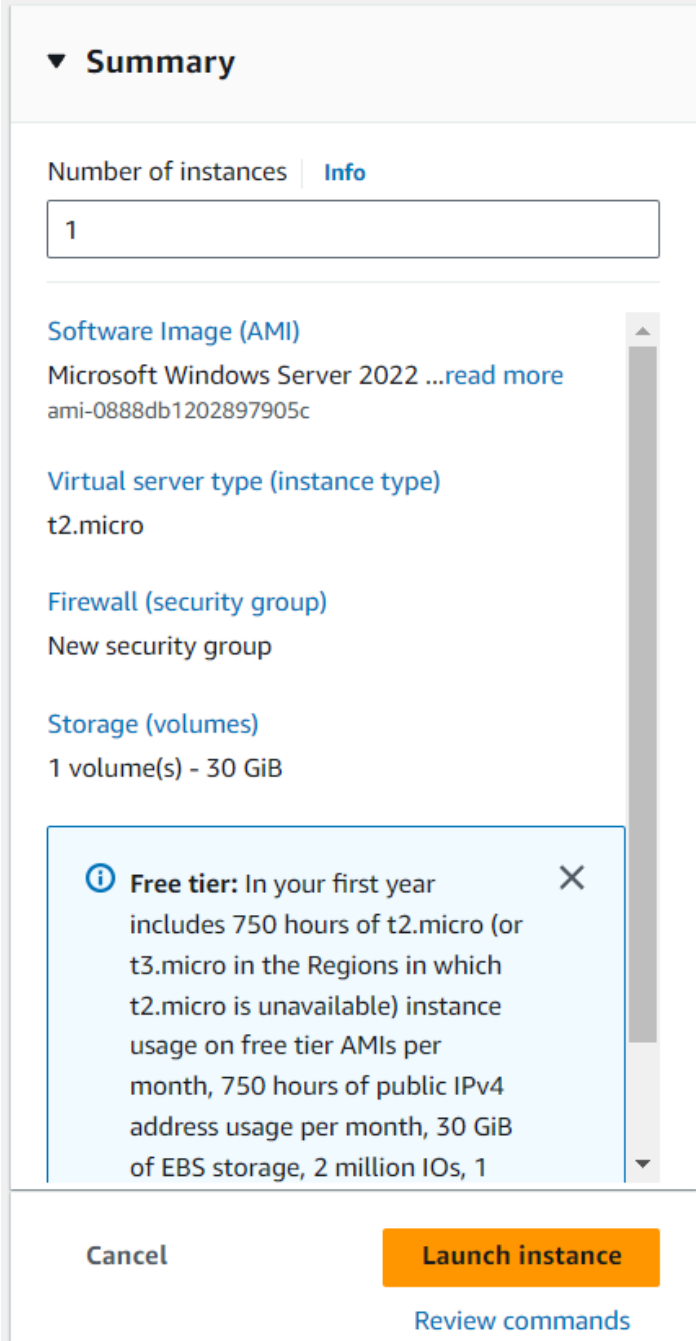
0 x File systems

Edit

Step 8: Launch Instance and Connect (RDP)

1. Launch Instance

After configuring your AMI, instance type, key pair, network, security group, and storage, the next step is to launch your instance. Clicking "Launch" will initiate the creation of your instance based on the specified configurations.



The screenshot shows the 'Summary' tab of the AWS 'Launch Instance' wizard. It displays the following configuration details:

- Number of instances:** 1
- Software Image (AMI):** Microsoft Windows Server 2022 ...[read more](#) ami-0888db1202897905c
- Virtual server type (instance type):** t2.micro
- Firewall (security group):** New security group
- Storage (volumes):** 1 volume(s) - 30 GiB

A blue information box at the bottom states: **Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1

At the bottom right, there are two buttons: 'Cancel' and 'Launch instance' (highlighted in orange). Below the 'Launch instance' button is a link that says 'Review commands'.

2. Connect to Instance

Once your instance is in the running state, you can connect to it. In the EC2 dashboard, select your instance and click on the “Connect” button. This will provide you with the details on how to connect to your instance, including using RDP for Windows instances.

Instances (1/1) [Info](#)

Last updated
2 minutes ago

Connect

Instance state ▼

Actions ▼

Launch instances ▼

Find Instance by attribute or tag (case-sensitive)

All states ▼

Instance ID = i-091213229e03de687

Clear filters

< 1 >

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state ▼	Instance type ▼	Status check	Alarm status	Availability Zone ▼	Public IPv4 DNS ▼	Public
<input checked="" type="checkbox"/>	windows	i-091213229e03de687	Running	t2.micro	Initializing	View alarms +	us-east-1d	ec2-52-86-238-196.co...	52.86


3. Get Windows Password


For a Windows instance, AWS generates a random password for the Administrator account. You will need to retrieve this password to connect:

- Click on "Get Windows Password" in the EC2 dashboard.
- Upload your private key file (the .pem file generated when you created the key pair).
- AWS will decrypt and display the Administrator password. Copy this password for use in the RDP connection.


Get Windows password [Info](#)


Use your private key to retrieve and decrypt the initial Windows administrator password for this instance.

Instance ID
 i-091213229e03de687 (windows)

Key pair associated with this instance
 windows

Private key
Either upload your private key file or copy and paste its contents into the field below.

 Upload private key file

 Windows.pem
1.674KB

Private key contents - *optional*

```
9gfgDJK+QinCa83D7BKHKYDgWrNkTqzfnqDzz/VgN2+x123yIRnXAW45ypBau31g
iAjlh0X0sLU+gcjhUOvUtwKBgQCCygQ/wLx/IRsBJHl62wEZ+D9QxRJ7MJAec7Hi
tOQf0KUhpIszO/EK26lVdCdbmBC/M5b1Xd9GXo7U3tg3cEyM9jggf+vcCoLgVTw0
rB67W0oE+kA8ZgRQ+4b7gv4QgS/0SRjDVGftGtyRzu3i/EjihkMfXf51032itdQo
uWG0EQKBgDvFzJjxP85hNfgJKpi8Ouo9X3rXCSbZ5/uDUf1JbfukjO86EvF2YSdl
fy8AAJG+txjUR+COuaqNwdxU0+hxn6EfzfO3eSU2lORTINZQ+1xX48949lYtrpZG
eKAvtYAmkRusFeIOd6ZrVs41YlGI2iAcdWf/U6WzXS1r1UvN8iJ8
-----END RSA PRIVATE KEY-----
```

Cancel Decrypt password

4. Connect (RDP)

You can now connect to the instance using Remote Desktop Protocol (RDP):

- Open the Remote Desktop Connection app on your local Windows machine.
- Enter the public IP address of your EC2 instance (visible in the EC2 console under instance details).
- When prompted, enter the username (Administrator) and the decrypted password obtained from the previous step.
- You will be logged into the Windows instance running in the AWS cloud.

