

AWS Cloud and DevOps Training by Mr. Mahendran Selvakumar

Organized by KPR Institute of Engineering and Technology
Department of Computer Science and Engineering

Set Up and Access a Windows Web
Server on AWS EC2 with Elastic IP



Sooriya N
III - CSE

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

windows

Add additional tags

Step 1: Create a Name for Your Instance

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

▼ 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

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Microsoft Windows Server 2022 Base

Free tier eligible

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

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

Description

Microsoft Windows 2022 Datacenter edition. [English]

Architecture

AMI ID

Username

64-bit (x86)

ami-0888db1202897905c

root

Verified provider

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.

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

Instance type

t2.micro

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

Free tier eligible

☒ 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 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.

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 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).

▼ Network settings Info

VPC - required Info

vpc-0289d6374d79fa1e8172.31.0.0/16(default)↻

Subnet Info

No preference↻ Create new subnet ↗

Auto-assign public IP Info

Enable↕

Additional charges apply when outside of free tier allowance

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

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0)

Remove

Type Info

HTTP↕

Protocol Info

TCP

Port range Info

80

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

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

✕

Add security group rule

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.

Step 6: Security Group

RDP (Port 3389): Allows remote management of the Windows EC2 instance via Remote Desktop.

HTTP (Port 80): Enables public access to the web server for serving content.

Step 8: 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.

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

▼ Summary

Number of instances Info

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

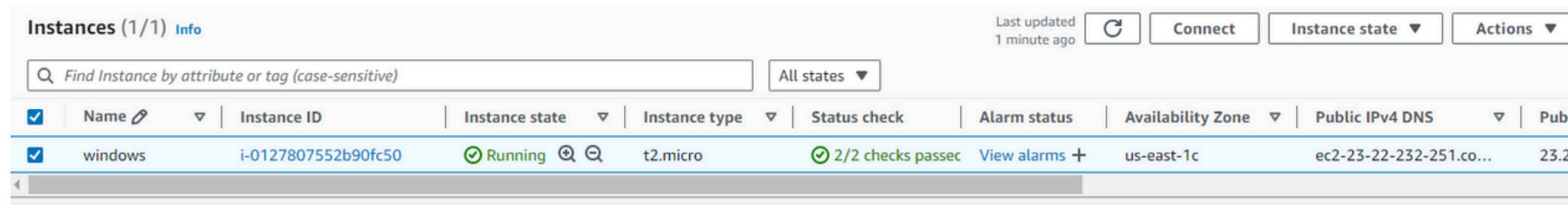
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1

Cancel Launch instance

Review commands

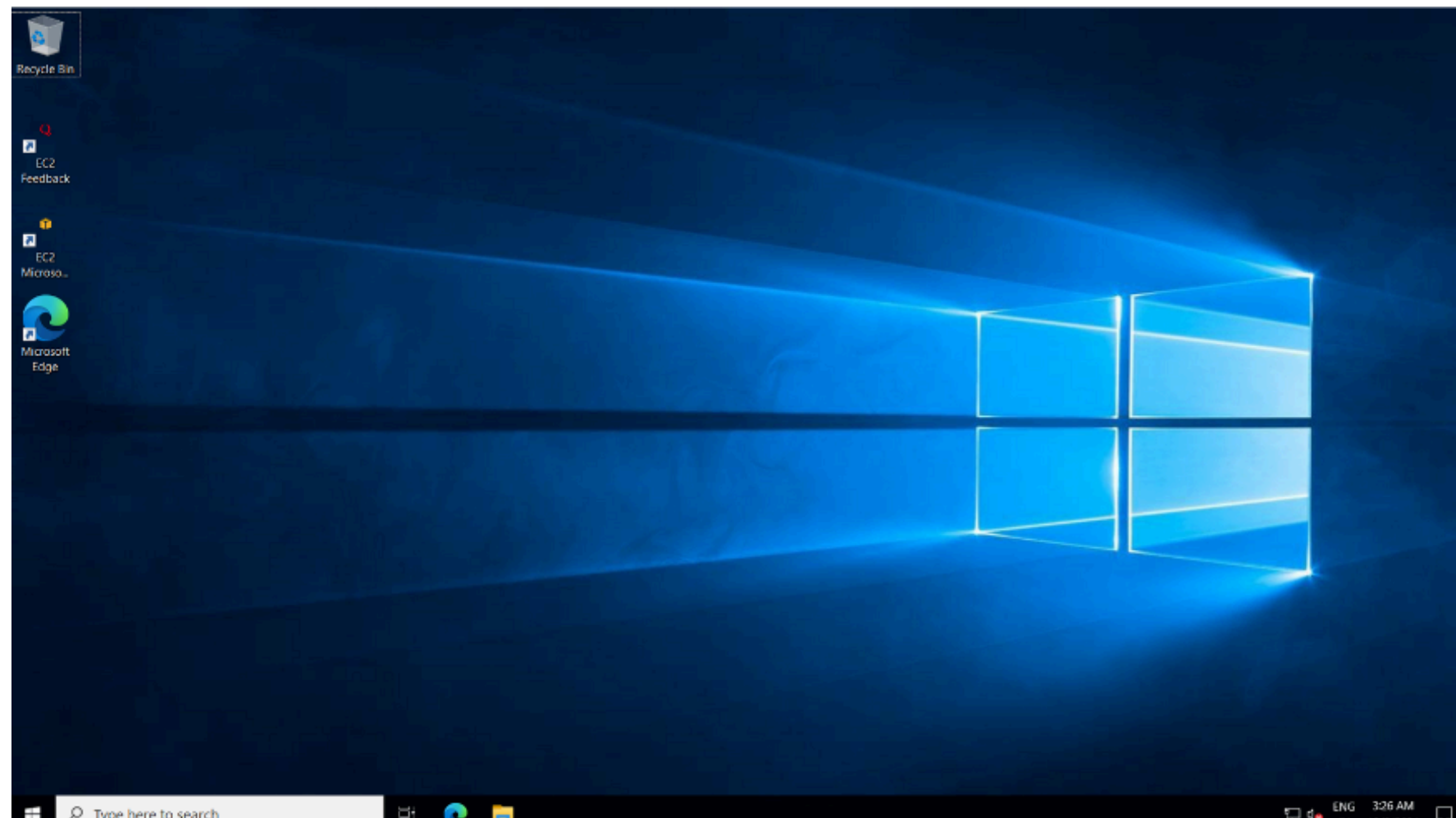
Step 9: Connect to Windows EC2 Instance

Set Up Windows EC2 Instance and Web Server: Launch a Windows EC2 instance, configure the required security groups for RDP and HTTP (ports 3389 and 80)



The screenshot shows the AWS Management Console 'Instances' page. At the top, there's a search bar and a filter for 'All states'. Below, a table lists the instances. One instance named 'windows' is shown, with ID 'i-0127807552b90fc50', in a 'Running' state, using the 't2.micro' instance type. It has passed 2/2 status checks and is located in the 'us-east-1c' availability zone. The public IPv4 DNS address is 'ec2-23-22-232-251.co...' and the public IP is '23.2'.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IP
windows	i-0127807552b90fc50	Running	t2.micro	2/2 checks passed	View alarms	us-east-1c	ec2-23-22-232-251.co...	23.2



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.

Step 10: Install IIS (Web Server) on the Windows Instance

Step 1: Use Remote Desktop (RDP) to connect to your Windows instance:

Open the RDP client on your local machine.

Use the Public IP of your instance (you can find it in the EC2 Dashboard).

Provide the username (usually Administrator) and the key pair to decrypt the password and log in.

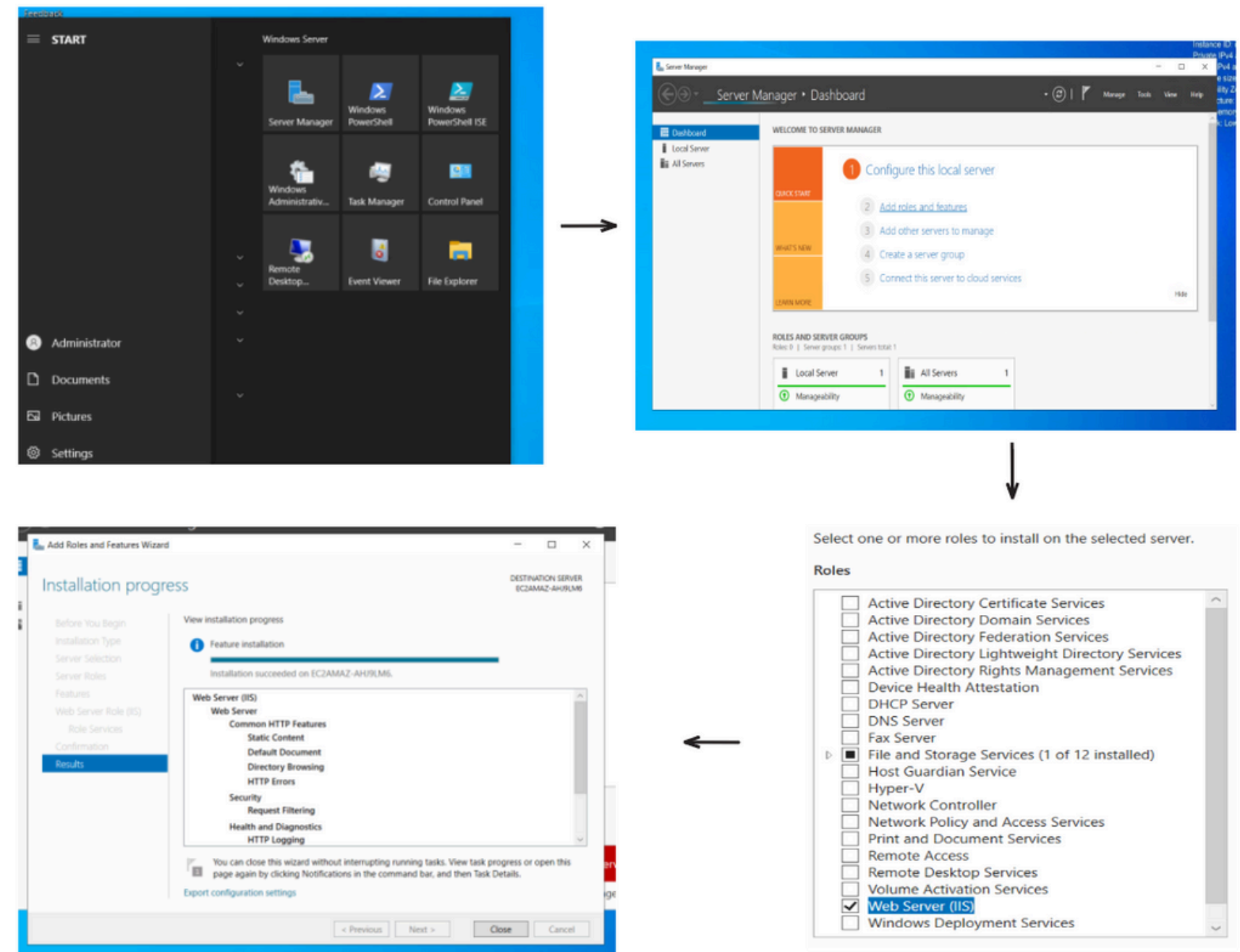
Step 2: Once logged in, open Server Manager.

Step 3: Click Manage > Add Roles and Features.

Step 4: In the Add Roles and Features Wizard, select the Web Server (IIS) role.

Step 5: Continue through the wizard, and click Install to install IIS.

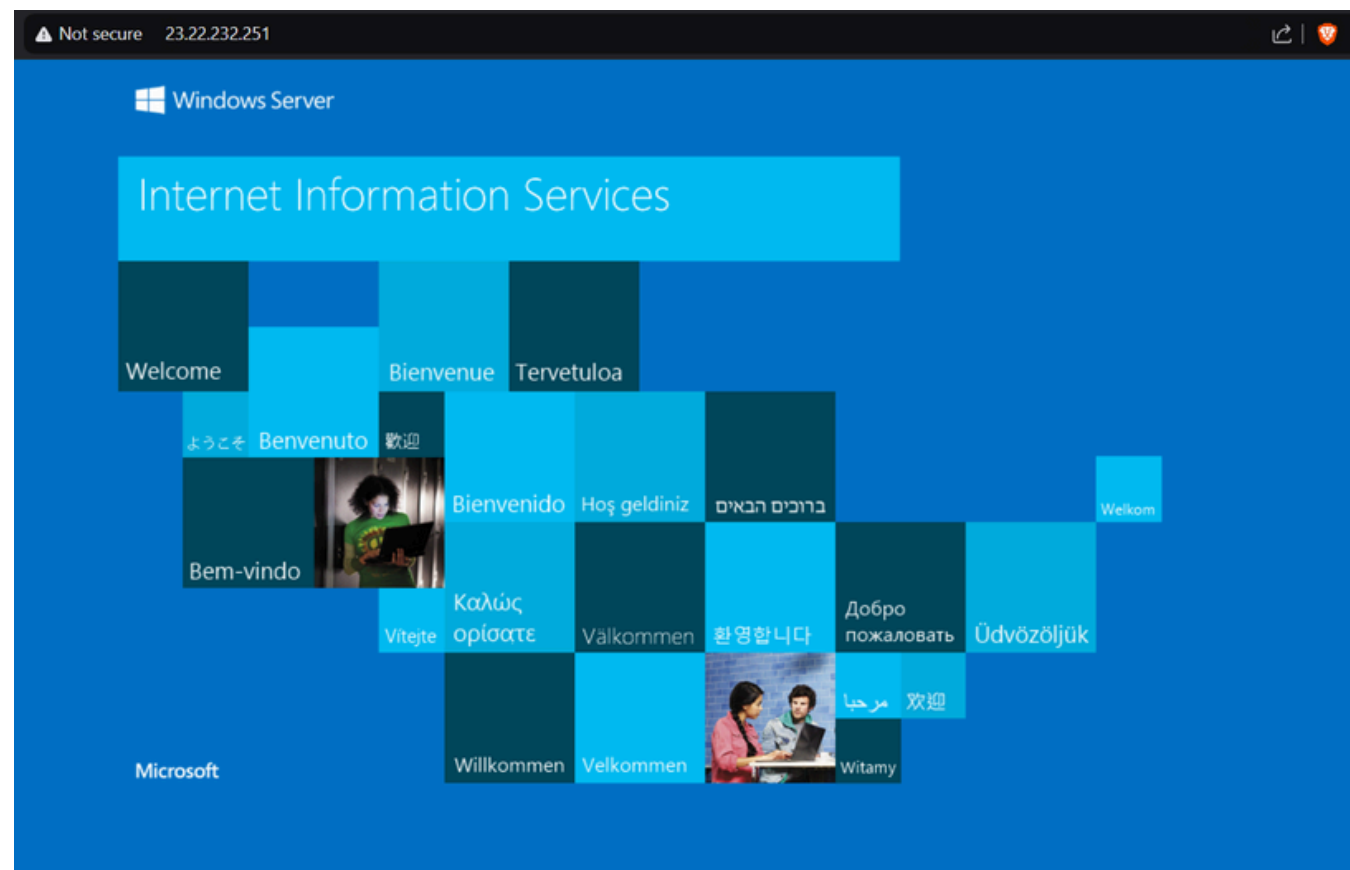
After installation, IIS should automatically start.



Step 11: Access the Web Server Using the Public IP

Step 1: Copy the Public IP of your EC2 instance from the EC2 Dashboard.

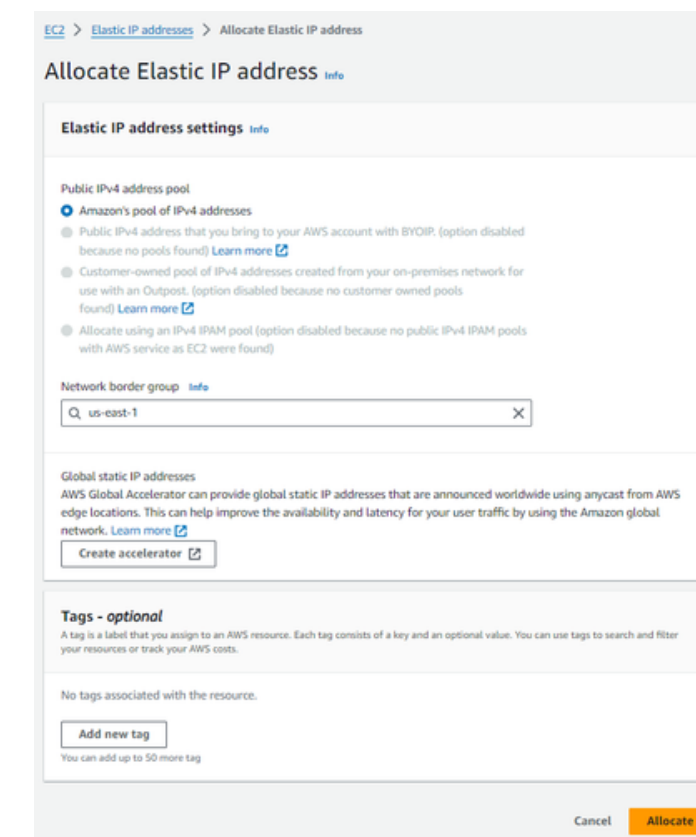
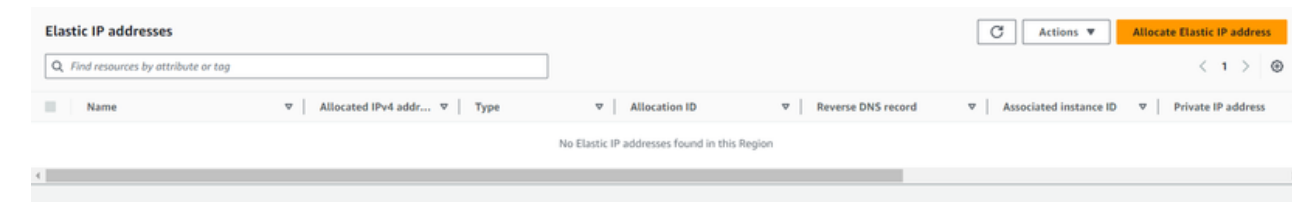
Step 2: Open a browser and enter `http://<Public_IP>` to verify the IIS default page is accessible, confirming the web server is running.



Step 12: Create an Elastic IP Address (EIP)

Step 1: Click on Elastic IPs under the "Network & Security" section.

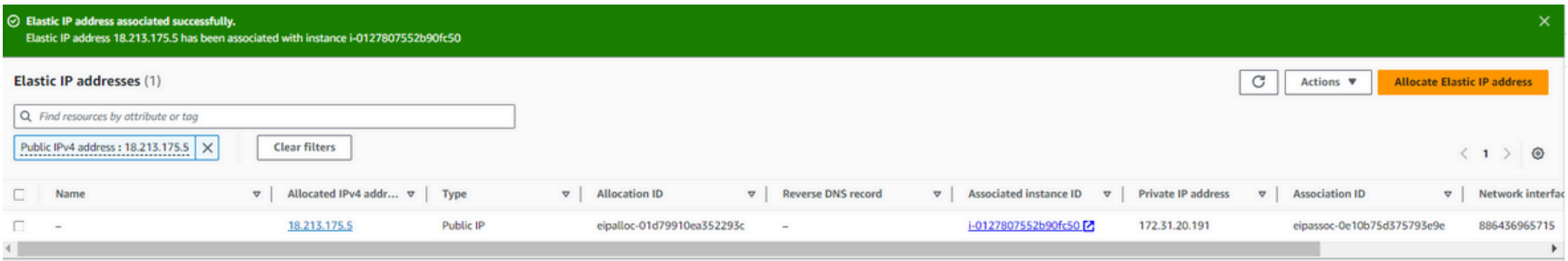
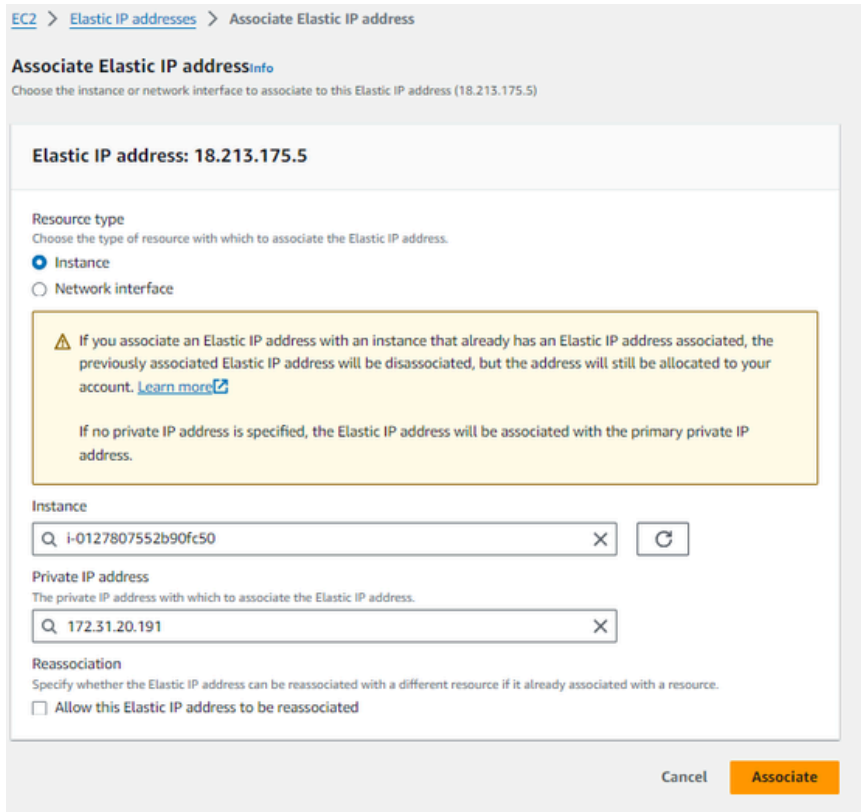
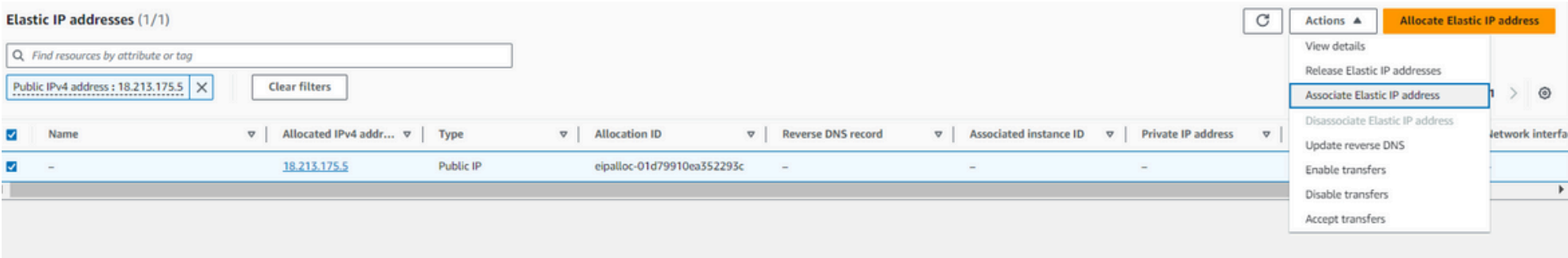
Step 2: Click on Allocate Elastic IP address.



Step 13: Assign Elastic IP to a Windows Web Server

- Step 1: On the Elastic IPs page, select the newly created Elastic IP.
- Step 2: Click on Actions > Associate Elastic IP address.
- Step 3: In the Associate Elastic IP address window, choose the Instance option and select your Windows web server instance.
- Step 4: Click on Associate to assign the Elastic IP to the Windows instance.

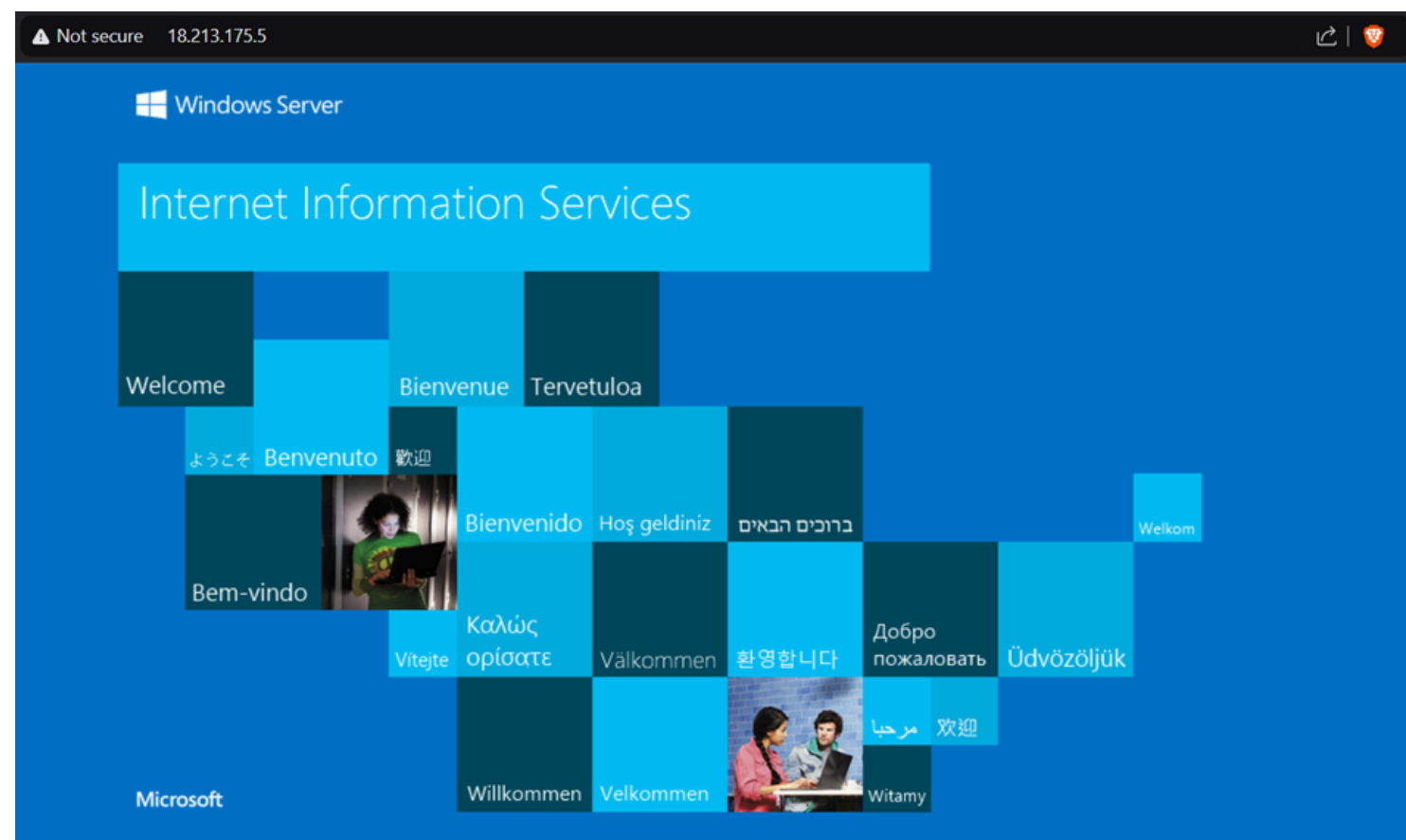
The importance of checking the private IP address when associating an Elastic IP is to ensure that the Elastic IP is correctly mapped to the right network interface on the EC2 instance.



Step 14: Access the Windows Web Server using the Elastic IP

Step 1: After associating the Elastic IP with your instance, note down the public Elastic IP.

Step 2: Now, you should be able to access the website hosted on your Windows web server by navigating to the Elastic IP address in a browser (e.g., `http://<Elastic_IP>`).



Key advantages of using an Elastic IP in AWS:

Static IP Address: Unlike regular public IPs that can change when an instance is stopped or restarted, Elastic IP remains static. This ensures consistent access to your EC2 instance or services.

Reliability: If you need to move your application or service to another instance (e.g., for scaling or maintenance), you can easily reassign the Elastic IP without any downtime.

Easy Management: Elastic IP allows better control over your instance's connectivity, making it easier to manage public-facing resources in the cloud.