Day2 of learning AWS

Tuesday, January 28, 2025 8:37 PM

AWS EC2 Service

Amazon EC2 (Elastic Compute Cloud) is one of the most widely used services in AWS, providing scalable virtual servers on demand. In this post, we will explore the basics of EC2, its configuration options, and how it helps businesses manage computing resources efficiently.

What Is EC2?

EC2 allows you to rent virtual servers in the cloud to run your applications. You can select the operating system, storage, and compute power according to your requirements and access these servers remotely.

Key Features:

- **Flexible Configurations**: Choose the desired CPU, memory, storage, and network capacity.
- **Pay-as-You-Go**: Only pay for the compute resources you use.
- **Bootstrap Scripts**: Automate server setup using custom scripts executed during launch.

Why Use EC2?

1. **Scalability**: Easily scale resources up or down based on demand.
2. **Cost Efficiency**: No need to invest in on-premises hardware.
3. **Flexibility**: Supports a wide variety of operating systems and configurations.
4. **Global Availability**: Deploy instances in AWS Regions closest to your users.

Key Configurations in EC2

When launching an EC2 instance, you can configure the following:

1. **Operating System (OS)**:
   - Choose from popular Linux distributions (e.g., Ubuntu, Red Hat) or Windows Server versions.
2. **Instance Type**:
   - Select the compute power and memory based on your workload. Examples:
   - **t2.micro**: Suitable for small workloads with 1 CPU and 1 GB RAM.
   - **m5.large**: For medium workloads with 2 CPUs and 8 GB RAM.
3. **Network Settings**:
   - Configure inbound and outbound traffic rules using security groups.
   - Example: Allow SSH (port 22) for secure remote access.
4. **Storage**:
   - Attach storage volumes to your instance. Example:
   - **8 GB** root volume for OS installation.
5. **Key Pairs**:
   - Use public and private key pairs to securely access your instance.
6. **Bootstrap Scripts**:
   - Automate tasks like installing software or configuring DNS with startup scripts.

Steps to Launch an EC2 Instance

1. **Select Region**:
   - Choose an AWS Region (e.g., Asia Pacific - Mumbai) where the instance will be hosted.
2. **Launch Instance**:
   - Go to the EC2 dashboard and click **Launch Instance**.
3. **Choose AMI (Amazon Machine Image)**:
   - Select an OS image like Ubuntu LTS.
4. **Select Instance Type**:
   - Choose an instance type (e.g., t2.micro for free tier users).
5. **Configure Instance Details**:
   - Set up networking, storage, and other configurations.
6. **Add Storage**:
   - Specify storage size and type for your instance.
7. **Add Tags**:
   - Optionally, assign tags for easier management (e.g., Name: MyEC2Instance).
8. **Configure Security Groups**:
   - Define inbound rules, such as allowing SSH access.
9. **Review and Launch**:
   - Review configurations and launch the instance.
10. **Download Key Pair**:
    - Save the private key file to access your instance securely.

Example: Creating a Basic EC2 Instance

1. **Select Ubuntu LTS as the OS**.
2. **Choose t2.micro** instance type.
3. Configure a security group to allow **SSH (port 22)** access.
4. Attach an **8 GB storage volume**.
5. Launch the instance and download the private key file.

6. Use the key file to securely access the instance using SSH.

Use Cases of EC2

1. **Web Hosting**:
   - Deploy web servers to host websites and applications.
2. **Development and Testing**:
   - Create isolated environments for coding and testing.
3. **Batch Processing**:
   - Run scheduled tasks and data processing jobs.
4. **Gaming Servers**:
   - Host multiplayer gaming platforms with low latency.

AWS EC2 is a versatile service that enables businesses to deploy and manage virtual servers with ease. By customizing configurations and leveraging advanced features like bootstrap scripts, you can create scalable and efficient infrastructure tailored to your needs.

Accessing AWS EC2 Instances

In the previous post, we explored how to create an EC2 instance in AWS. Now, let's learn how to access it securely using different methods on Windows, Linux, and even directly from your browser.

Accessing EC2 from Windows 10 (and Above)

Windows 10 and later versions come with built-in SSH support, making it easy to access your EC2 instance. Follow these steps:

1. Navigate to the Location of Your Key Pair:
   - Open the folder where you downloaded the `.pem` file for your instance key pair.
   - Right-click within the folder and select Open in Terminal or type `cmd` in the folder path to open Command Prompt in that directory.
2. Run the SSH Command: Use the following command structure:
   `ssh -i <key_file_name>.pem ubuntu@<public_dns>`

   Example:
   `ssh -i youtube-key.pem ubuntu@ec2-18-223-45-12.compute-1.amazonaws.com`
   - Replace `<key_file_name>.pem` with your key file name.
   - Replace `<public_dns>` with the public DNS or IP of your EC2 instance.
3. Confirm the Connection:
   - When prompted, type `yes` to accept the host key fingerprint.
   - You are now logged into your EC2 instance.

Accessing EC2 from Older Versions of Windows

For Windows versions without built-in SSH:

1. Install Git Bash:
   - Download Git Bash from the official website.
   - Install and open Git Bash.
2. Navigate to the Key Pair Location:
   - Use the `cd` command to move to the folder containing the `.pem` file.
3. Run the SSH Command:
   - Use the same SSH command mentioned above.

Accessing EC2 from Linux

Linux systems have native SSH support. Here’s how to access your instance:

1. Open Terminal:
   - Use the `cd` command to navigate to the folder where the `.pem` file is stored.
2. Change File Permissions (If Required):
   `chmod 400 <key_file_name>.pem`
3. Run the SSH Command:
   `ssh -i <key_file_name>.pem ubuntu@<public_dns>`
4. Confirm the Connection:
   - Type `yes` if prompted to accept the host key fingerprint.

Browser-Based Access Using AWS Management Console

For users who prefer browser-based access:

1. Log in to the AWS Console.
2. Navigate to EC2 Instances:
   - Select your instance and click Connect.
3. Choose Browser-Based SSH:
   - Under the "Session Manager" or "EC2 Instance Connect" tab, click Connect.
4. Access Your Instance:
   - The browser-based terminal opens, allowing you to manage the instance without additional tools.

Verifying Your Instance Configuration

After accessing your EC2 instance, verify the setup by running the following commands:

1. Check the Operating System:
   `cat /etc/os-release`

   Example Output:

Ubuntu 22.04 LTS

2. Check Memory:
   `free -m`

   Example Output:
   Total: 1GB RAM
3. Check CPU:
   `lscpu`

   Example Output:
   CPUs: 1 Core
4. Check Network Configuration:
   `ip a`

   Example Output:

- Public and Private IP addresses.

Avoiding Overloaded Availability Zones

AWS manages availability zones intelligently to balance resources. However, users often unknowingly select the default zone, potentially leading to overcrowding. To ensure balanced distribution:

- Explicitly specify the availability zone when launching instances.
- AWS maps zones uniquely for each account, so what appears as `us-east-1a` for one user may differ for another.

Accessing your EC2 instance is straightforward with the right tools and configurations. Whether you use SSH from Windows, Linux, or a browser-based solution, AWS provides multiple options for seamless instance management. By understanding these methods, you can ensure secure and efficient access to your cloud resources.