

(14)

## AWS EC2

### Elastic Cloud Compute

AWS EC2 (Amazon Elastic Compute Cloud) is a cloud service that provides resizable virtual servers, called instances, which you can use to run application.

- For hosting our application and managing need services which are → costly if buy physical → need maintenance
  - Instead we use the EC2 service where let you rent virtual servers in the cloud! These virtual servers are called instances that
- Note: → In EC2 no problem of buying & managing physical servers.

# You can configure several option #  
(Customize according to user)

- OS
- RAM
- Disk space
- CPU
- Network / Firewall  
And control, how to access the machine  
→ what permission need to be given and  
how much restriction need to apply.

(15)

# Terms (That is used when configure EC2)

1. Instance type :- Select the hardware capacity (e.g., C.P.U, memory).
2. AMI (Amazon machine Image) :- Choose the operating system & software (linux, mac, windows).
3. Storage : Configure the type and size of storage (eg. EBS volume).  
→ EBS :- (Elastic block store) is the hard disk (volume) attached to an EC2 instance.  
→ It's like a virtual SSD/HDD connected to your EC2 machine.
4. Security Groups :- Set up firewall rules to control inbound/outbound traffic  
→ Inbound → Traffic coming into your AWS resources (eg EC2)  
→ Outbound → Traffic leaving from your AWS resources to the outside.
5. key pair :- Create or use an existing key pair for SSH resources access.

(16)

Key pair is a set of a public key used and private key used to securely connect to your EC2 instance using SSH. You download the private key (.pem) and AWS keeps the public key.

→ You get the private key file only once → when you first create the key pair in AWS.

### # Steps to Create and download EC2 key pair (.pem)

→ Go to AWS EC2 dashboard

→ Open EC2 console

→ In the left menu, click on "key pairs"  
(under Network & Security)

→ Create a new key pair

→ Click on "Create key pair"

→ Set

- Name → eg, my-ec2-key

- Key pair type → RSA [RSA] Cryptographic keys

for secure access to EC2 instance] (Recommended).

- Private key format → .pem (for Linux/macOS) or .ppk (for Windows PuTTY)

- Click "Create key pair".

(17)

## SSH (Secure Shell)

SSH is a secure way to connect to remote servers (like EC2) over the internet using a terminal / command line.

→ It lets you control the EC2 servers from your computer.

Note: → VSCode + SSH  
using by Remote-SSH extension-VS.

6. Network Settings: Configure VPC (virtual network), subnet, and assign public or private IP address.

7. IAM Role: Attach an IAM role for permissions to access other AWS resources.

8. User Data: Add scripts to be executed when the instance starts. Eg. → Install node.js  
→ Clone a github repo  
→ Install dependencies  
→ Start the Node.js app (Express or any backend).

9. Elastic IP: Optionally associate a static IP address for consistent public access.

(B)

Eg: You're building a college admission system using node.js (as I make), and you:

1. Deploy backend to EC2
2. Assign Elastic IP
3. Setup a domain like admissionportal.in to point to it
4. Now it's permanently accessible until you delete the IP.

Note: → EC2 service is region specific  
show instance from which region you make

→ also global view also available for see all instances

### Steps to → Creating first EC2 instance

Step 1: Go to dashboard Search bar and Search.  
→ EC2 & click on it.

Step 2: Select the region on right top as  
for eg: Asia Pacific (Mumbai)  
Region code: ap-south-1

Step 3: click on Launch Instances & go to  
Launch an instance

Step 4: add name, set AMI (Amazon Machine Image) →  
Select OS, (for free tier or learning go with  
AWS Linux).

19

Vash Garg

Step-5 Select the instance type ALC to you.  
for CPU & RAM

Step-6  $\Rightarrow$  key-pair (login)

Click create key pair  $\rightarrow$  key pair name,  
 $\rightarrow$  key pair type  $\rightarrow$  RSA.  
 $\rightarrow$  private key file  $\rightarrow$  open  
 $\longrightarrow$  create key.

$\rightarrow$  private key download automatically.

Step-7  $\Rightarrow$  Set network setting.

$\rightarrow$  for web server  $\rightarrow$  allow HTTP traffic from internet.

Step-8  $\Rightarrow$  Customize EBS storage & make how many instances you want.

Step-9  $\Rightarrow$  User data  $\Rightarrow$  In Advance  $\rightarrow$  scroll down and see user data.

$\rightarrow$  Add script or upload.

Eg #!/bin/bash  
yum update -y  
curl -sL https://rpm.nodesource.com/setup-18.x | bash  
yum install -y nodejs

# create server file  
cat << EOF >> /home/ec2-user/server.js  
const http = require('http');  
const port = 3000;  
http.createServer((req, res) => {

(20)

res.end("Hello from node.js EC2");

3). listen(PORT, () => console.log(`Server running on PORT 3000`));  
EOF

# Run the Server

node /home/EC2-user/server.js &

Step-10 finally launce instance

AWS EC2 + EBS + Elastic IP - Cost & Management

(Real Cost for usage for medium projects - AIC to our requirement customize everything)

200 gbl month

• EC2 Instance :- Can Stop anytime to save cost

- No charges when stopped
- Data remains on EBS-

<u>Cost</u> instances	VCPU	RAM	USD/hr	INR
t3.micro	2	1GB	\$0.0104	₹0.86/hr
t3.medium	2	1GB	\$ 0.0416	₹ 3.44/hr

• EBS Volume (Storage) Price → \$ 0.08 / GB-month

• Stores all websites data / server data.

• charged per gblmonth, even if EC2 is stopped

• Deleted EBS = permanent data loss

• TIP: Create snapshot before deleting.

(21)

### 3. EBS Snapshots (Backup) ( $\approx \$4.15 / \text{GB-month}$ )

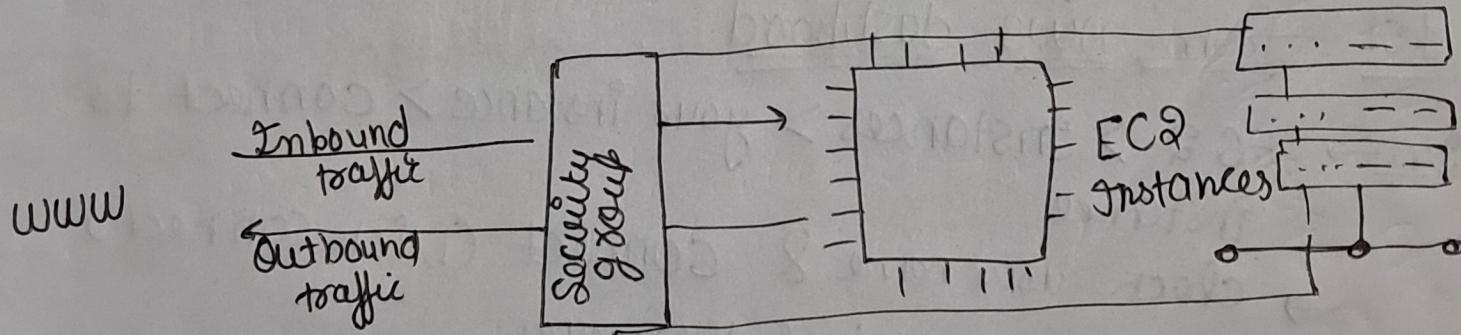
price:  $\$0.05 / \text{GB-month}$  standard rate  
 = incremental: only charged block bill mode.

⇒ Elastic IP → free only when attached to running EC2.

= charged when not attached or EC2 is stopped ( $\approx \$0.45 / \text{hours}$ ).

Tip ⇒ Release when not needed, or resign later.

Security Groups ⇒ Network control inbound & outbound traffic for instances.



#### Inbound rules to add

Type	Port	Source	Purpose
SSH	22	my IP	Connect via VS Code
HTTP	80	Anywhere (0.0.0.0)	serve frontend / backend
HTTPS	443	Anywhere	for secure frontend
CUSTOM TCP	3000	Anywhere or my IP	for Node.js / Express dev server

(22)

- # Never open SSH to anywhere for security reason
- # Outbound rule:
  - = Leave default: all traffic allowed.
  - = Security groups are region specific
  - = By default inbound traffic blocked & outbound allowed.
  - = You define rules for specific:
  - = Protocols
  - = Port no.
  - = IP address & range

### Accessing our EC2 instances

→ How to SSH into EC2 instances?

SSH allows you to control / access a remote machine

1st from AWS dashboard

In → EC2 > Instances > your instance > Connect to instance

→ Check user name & connect (click connect)

→ And go to terminal & run.

2nd SSH EC2 from windows

To using PUTTY

→ download putty.

→ Open putty gen → Load → go where private key

downloaded → Load it → save private key

→ PPK format save it → PPK private key format

form.

(23)

- Now open putty.
- Enter user name and public key from EC2 instance
- Tip: make same.
- Load that PPK private key.
- Once done. Login as - user name.

for MAC users → given in Connect to instance →  
SSH client → all steps given.

### (ii) VS Code (Graphical via SSH)

Steps: install remote-SSH extension in VSCode

- Add .pem key in SSH config
- connect to EC2 from sidebar.

### EC2 Termination

EC2 → EC2 Instances → select that you want  
to terminate.

- Click on Action → Instance State → do Stop,  
Reboot & Terminate (Delete) permanently.

## Some Instance Types

### General Usage

≡ Case 1 Small website or blog.

Suitable → t3.micro or t3.small

≡ Case 2 E-Commerce Application.

Suitable m5.large or m5.xlarge

≡ Case 3 Real time video Rendering & streaming.

→ Suitable g5.12xlarge or g5.24xlarge.

≡ Case 4 In-memory Database for Real time Analytics  
(memory optimized).

≡ r5g.16xlarge or x2idn.32xlarge.