

FACE DETECTION USING AWS

NAME: AVULA ROHITHA

PART-1

ARCHITECTURE:

Telegram bot will accept an image, and it transfers to service called EC2.

Once EC2 accepts an image, it puts an image in bucket (Storage medium) which is inside S3

After completing this activity by EC2, it asks for recognition for uploaded image

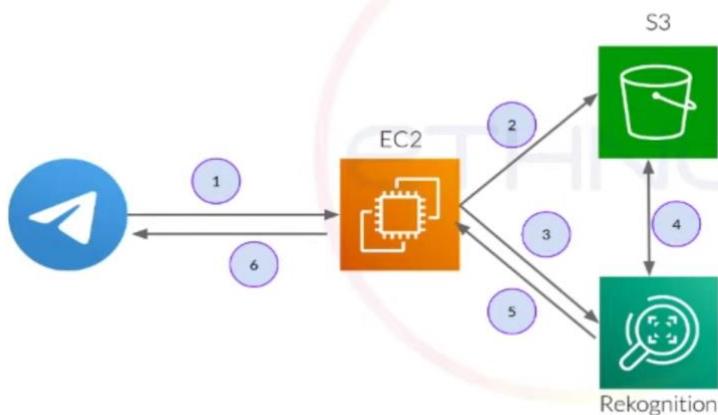
Rekognition is a service which understands the image (detecting features, how many faces are there, whether they are male or female, age etc)

Rekognition asks S3 for an image which EC2 has uploaded

S3 will give image Rekognition will detect features in image

EC2 will take response back given by rekognition

EC2 will send it back to telegram



LOGIN SCREEN:

The screenshot shows the AWS Management Console login page. The top navigation bar includes links for AWS Management Console, Billing Management Consoles, Services (selected), Resource Groups, and Support. The main content area displays the AWS Management Console logo and several service links:

- AWS services**: Includes Find Services (with a search bar), Recently visited services (EC2, S3, Lambda, Amazon Rekognition, IAM), and All services.
- Access resources on the go**: Provides a link to the AWS Console Mobile App.
- Explore AWS**: Offers Free Digital Training (with a link to 350+ courses), AWS IQ (with a link to connect with experts), Amazon SageMaker Studio (with a link to the first visual integrated development environment for machine learning), and Amazon CloudWatch.

The screenshot shows the AWS Billing Management Console interface. On the left, there's a sidebar with various navigation links like Home, Cost Management, Cost Explorer, Budgets, etc. The main content area has three sections: 'Account Settings' (with fields for Account Id, Name, and Password), 'Contact Information' (with detailed address information), and 'Alternate Contacts' (which is currently empty). A note at the bottom of the contact section states: 'Please note that updating your contact information on this page will not update the information displayed on your PDF Invoices. If you wish to update the billing address information associated with your Invoice, please edit it through the Payment Methods page, located [here](#).'

PART-1EC2:

EC2 Allows a user to rent virtual computer on which to run their own computer applications.

Computer Machine comprises of 3 Layers

- 1)Hardware: Keyboard, Mouse, HardDisk, RAM, Wi-Fi etc
- 2)Operating Systems: Windows 10, Ubuntu, Linux, Mac etc
- 3)Software: Word, Excel, Paint etc

STEPS INVOLVED TO CREATE EC2:

- 1)Select the operating system.
- 2)To take and configure the RAM and Processor
- 3)Configure Storage
- 4)Configure Security (How do I login the machine; Username, Password)

Screenshot of the AWS Services menu showing the "Compute" section selected. The menu includes a search bar, user information, and sorting options.

Compute	Blockchain	Analytics	End User Computing
EC2	Amazon Managed Blockchain	Athena	WorkSpaces
Lightsail		EMR	AppStream 2.0
Lambda		CloudSearch	WorkDocs
Batch	Satellite	Elasticsearch Service	WorkLink
Elastic Beanstalk	Ground Station	Kinesis	
Serverless Application Repository		QuickSight	
AWS Outposts		Data Pipeline	Internet Of Things
EC2 Image Builder	Amazon Braket	AWS Data Exchange	IoT Core
Storage	Management & Governance	AWS Glue	FreeRTOS
S3	AWS Organizations	AWS Lake Formation	IoT 1-Click
EFS	CloudWatch	MSK	IoT Analytics
FSx	AWS Auto Scaling		IoT Device Defender
S3 Glacier	CloudFormation		IoT Device Management
Storage Gateway	CloudTrail		IoT Events
AWS Backup	Config		IoT Greengrass
Database	OpsWorks		IoT SiteWise
	Service Catalog		IoT Things Graph

Select EC2 EC2 DASHBOARD:

Screenshot of the AWS EC2 Dashboard.

Left sidebar navigation:

- New EC2 Experience (radio button)
- Events New
- Tags
- Reports
- Limits
- INSTANCES**
 - Instances
 - Instance Types
 - Launch Templates New
 - Spot Requests
 - Savings Plans
 - Reserved Instances
 - Dedicated Hosts New
 - Capacity Reservations
- IMAGES**
 - AMIs
 - Bundle Tasks
- ELASTIC BLOCK STORE**

Main content area:

- Launch instance**: To get started, launch an Amazon EC2 Instance, which is a virtual server in the cloud. **Launch instance** button.
- Scheduled events**: US East (Ohio) - No scheduled events.
- Migrate a machine**: Use CloudEndure Migration to simplify, expedite, and automate large-scale migrations from physical, virtual, and cloud-based infrastructure to AWS. **Get started with CloudEndure Migration** link.

Click Launch instance

CHOOSING AN AMI:

The screenshot shows the AWS console interface for choosing an AMI. The top navigation bar includes 'Services' and 'Resource Groups'. Below the navigation is a progress bar with steps: 1. Choose AMI (highlighted), 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. A 'Cancel and Exit' button is on the right.

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search bar: Search for an AMI by entering a search term e.g. "Windows"

Quick Start sidebar: My AMIs, AWS Marketplace, Community AMIs, Free tier only (unchecked).

Results table:

Image	Name	Description	Select	64-bit (x86)	64-bit (Arm)
Amazon Linux icon	Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0e01ce4ee18447327 (64-bit x86) / ami-03201f374ab66a26e (64-bit Arm)	Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras.	Select	<input checked="" type="radio"/>	<input type="radio"/>
Amazon Linux icon	Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-01b01bbd08f24c7a8	The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.	Select	64-bit (x86)	
Red Hat icon	Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-0520e698dd500b1d1 (64-bit x86) / ami-0099847d600887c9f (64-bit Arm)	Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-0520e698dd500b1d1 (64-bit x86) / ami-0099847d600887c9f (64-bit Arm)	Select	<input type="radio"/>	

Select Amazon Linux 2 (Free tier eligible) 64 Bit(X86)

CHOOSING AN INSTANCE TYPE:

The screenshot shows the AWS console interface for choosing an instance type. The top navigation bar includes 'Services' and 'Resource Groups'. Below the navigation is a progress bar with steps: 1. Choose AMI, 2. Choose Instance Type (highlighted), 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. A 'Cancel and Exit' button is on the right.

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance types, Current generation, Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes

Buttons: Cancel, Previous, Review and Launch, Next: Configure Instance Details

Choose General purpose t2 micro (Free Tier eligible) and then press Next: Configure instance details

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances	<input type="text" value="1"/>	Launch into Auto Scaling Group
Purchasing option	<input type="checkbox"/> Request Spot instances	
Network	<input type="text" value="vpc-427bb129 (default)"/>	<input type="button" value="Create new VPC"/>
Subnet	<input type="text" value="No preference (default subnet in any Availability Zone)"/>	<input type="button" value="Create new subnet"/>
Auto-assign Public IP	<input type="button" value="Use subnet setting (Enable)"/>	
Placement group	<input type="checkbox"/> Add instance to placement group	
Capacity Reservation	<input type="text" value="Open"/>	<input type="button" value="Create new Capacity Reservation"/>
IAM role	<input type="text" value="None"/>	
Shutdown behavior	<input type="text" value="Stop"/>	
Stop - Hibernate behavior	<input type="checkbox"/> Enable hibernation as an additional stop behavior	

Buttons: Cancel | Previous | **Review and Launch** | Next: Add Storage

Click add storage

ADDING STORAGE:

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-0f54692056aaa4c20	<input type="text" value="8"/>	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

Add New Volume

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

Buttons: Cancel | Previous | **Review and Launch** | Next: Add Tags

Click Next: Add Tags

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. A copy of a tag can be applied to volumes, instances or both. Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key (128 characters maximum) | Value (256 characters maximum) | Instances (i) | Volumes (i)

This resource currently has no tags

Choose the Add tag button or [click to add a Name tag](#). Make sure your [IAM policy](#) includes permissions to create tags.

Add Tag (Up to 50 tags maximum)

Cancel Previous Review and Launch Next: Configure Security Group

Click Next: Configure Security group CONFIGURE SECURITY GROUP:

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a new security group Select an existing security group

Security group name: launch-wizard-2

Description: launch-wizard-2 created 2020-03-26T20:23:34.349+05:30

Type (i)	Protocol (i)	Port Range (i)	Source (i)	Description (i)
SSH	TCP	22	Custom	0.0.0.0/0
e.g. SSH for Admin Desktop <input type="button" value="x"/>				

Add Rule

Warning
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.

Cancel Previous Review and Launch

Click Review and Launch

AWS Services Resource Groups

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 7: Review Instance Launch

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0e01ce4ee18447327

Free tier eligible

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras.

Root Device Type: ebs Virtualization type: hvm

Instance Type Edit instance type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Security Groups Edit security groups

Security group name: launch-wizard-2
Description: launch-wizard-2 created 2020-03-26T20:23:34.349+05:30

Type i	Protocol i	Port Range i	Source i	Description i
SSH	TCP	22	0.0.0.0/0	

Instance Details Edit instance details

Cancel Previous Launch

Click Launch

DOWNLOAD KEYPAIR:

Select an existing key pair or create a new key pair X

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Create a new key pair

Key pair name

Facedetection

Download Key Pair



You have to download the **private key file (*.pem file)** before you can continue.
Store it in a secure and accessible location. You will not be able to download the file again after it's created.

Cancel

Launch Instances

Choose create a new key pair
Name the key pair
Download key pair
It gets downloaded with .pem extension
After downloading keypair click launch instances



How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click [View Instances](#) to monitor your instances' status. Once your instances are in the **running** state, you can **connect** to them from the Instances screen. [Find out](#) how to connect to your instances.

Here are some helpful resources to get you started

- [How to connect to your Linux instance](#)
- [Amazon EC2: User Guide](#)
- [Learn about AWS Free Usage Tier](#)
- [Amazon EC2: Discussion Forum](#)

While your instances are launching you can also

[Create status check alarms](#) to be notified when these instances fail status checks. (Additional charges may apply)

Click on the following instance initiated

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public
i-0ad69d15508c44475	i-0ad69d15508c44475	t2.micro	us-east-2c	running	2/2 checks ...	None	ec2-18-217-97-62.us-east-2.compute.amazonaws.com

Instance: i-0ad69d15508c44475 Public DNS: ec2-18-217-97-62.us-east-2.compute.amazonaws.com

Description	Status Checks	Monitoring	Tags
Instance ID: i-0ad69d15508c44475			
Instance state: running			
Instance type: t2.micro			

Description		Status Checks	Monitoring	Tags
Instance ID	i-0ad69d15508c44475	Public DNS: ec2-18-217-97-62.us-east-2.compute.amazonaws.com		
Instance state	running	Public DNS (IPv4)	ec2-18-217-97-62.us-east-2.compute.amazonaws.com	
Instance type	t2.micro	IPv4 Public IP	18.217.97.62	
Finding	Opt-in to AWS Compute Optimizer for recommendations. Learn more	IPv6 IPs	-	
Private DNS	ip-172-31-40-136.us-east-2.compute.internal	Elastic IPs		
Private IPs	172.31.40.136	Availability zone	us-east-2c	
Secondary private IPs		Security groups	launch-wizard-2, view inbound rules, view outbound rules	
VPC ID	vpc-427bb129	Scheduled events	No scheduled events	
Subnet ID	subnet-a743d0eb	AMI ID	amzn2-ami-hvm-2.0.20200304.0-x86_64-gp2 (ami-0e01ce4ee18447327)	
Network interfaces	eth0	Platform details	-	
IAM role	-	Usage operation	-	
Key pair name	Facedetection	Source/dest. check	True	
Owner	481453507501	T2/T3 Unlimited	Disabled	
Launch time	March 26, 2020 at 8:32:22 PM UTC+5:30 (less than one hour)	EBS-optimized	False	
Termination protection	False	Root device type	ebs	
Lifecycle	normal	Root device	/dev/xvda	
Monitoring	basic	Block devices	/dev/xvda	
Alarm status	None	Elastic Graphics ID	-	
Kernel ID	-	Elastic Inference accelerator ID	-	
RAM disk ID	-	Capacity Reservation	-	
		Capacity Reservation Settings	Open	
		Outpost Arn	-	

This particular based instance is Linux based instance, for connecting to a linux based instance there is a software called Putty

18.217.97.62 - IP address

aws linux operating systems basically follows key pairs means there will be public key and private key (we need to have both in order to get connected)
 public keys will be stored on the server(instance)
 private keys- we need to upload while connecting to the server connecting to the serve using putty
 putty accepts private key in form of .ppk
 but we downloaded the file from aws inform of .pem
 so, we should .pem to .ppk

STEPS FOR GETTING CONNECTED:

- 1) Convert pem to ppk – puttygen

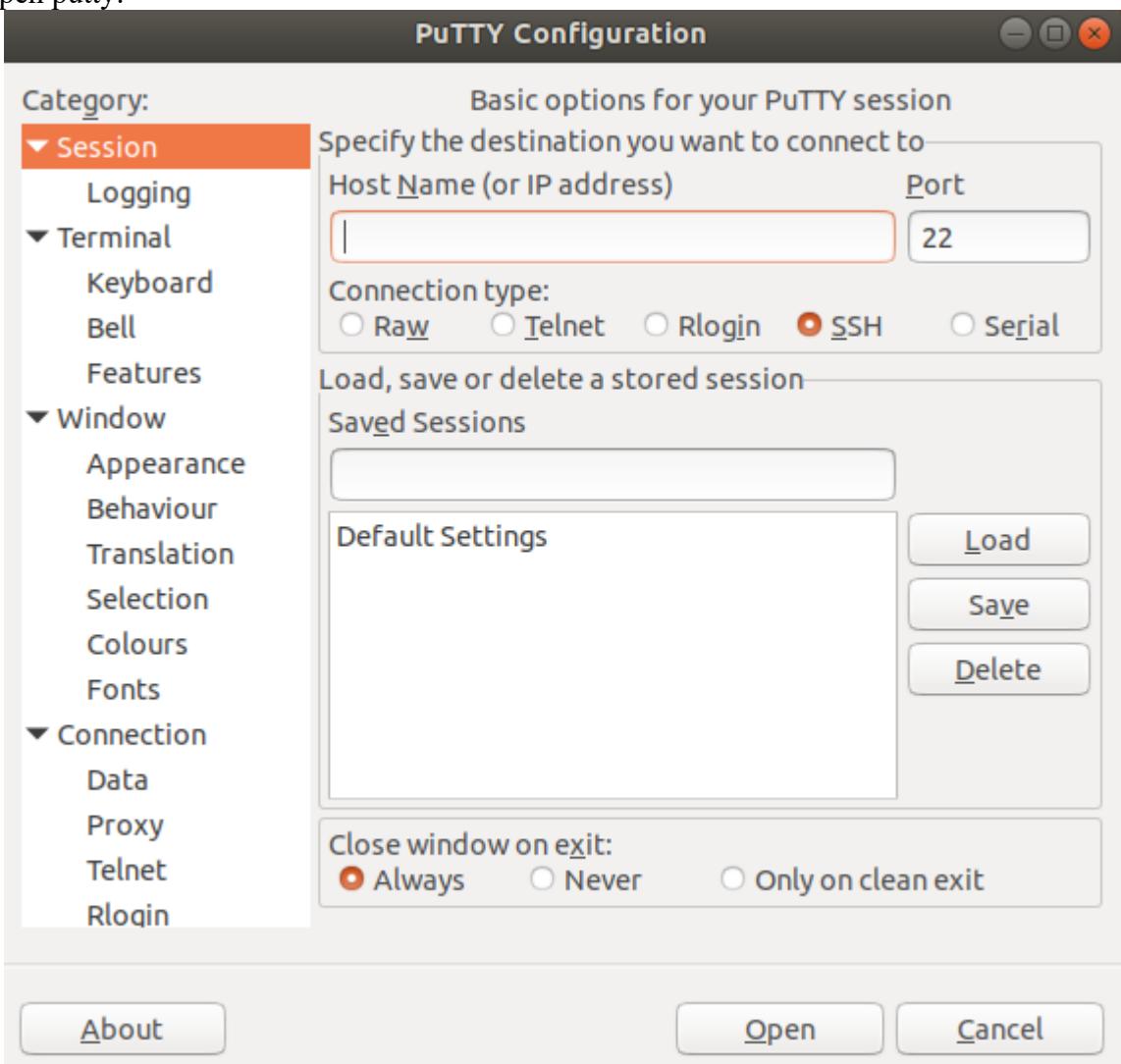
To install putty:

```
rohitha@rohitha-VirtualBox: ~$ sudo apt-get install putty -y
[sudo] password for rohitha:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  putty-tools
Suggested packages:
  putty-doc
The following NEW packages will be installed:
  putty putty-tools
0 upgraded, 2 newly installed, 0 to remove and 297 not upgraded.
Need to get 788 kB of archives.
After this operation, 3,362 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu bionic/universe amd64 putty-tools amd64 0.70-4 [386 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu bionic/universe amd64 putty amd64 0.70-4 [401 kB]
Fetched 788 kB in 5s (156 kB/s)
Selecting previously unselected package putty-tools.
(Reading database ... 163057 files and directories currently installed.)
Preparing to unpack .../putty-tools_0.70-4_amd64.deb ...
Unpacking putty-tools (0.70-4) ...
Selecting previously unselected package putty.
Preparing to unpack .../putty_0.70-4_amd64.deb ...
Unpacking putty (0.70-4) ...
Processing triggers for mime-support (3.60ubuntu1) ...
Processing triggers for desktop-file-utils (0.23-1ubuntu3.18.04.2) ...
Setting up putty-tools (0.70-4) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for gnome-menus (3.13.3-11ubuntu1.1) ...
Setting up putty (0.70-4) ...
rohitha@rohitha-VirtualBox: ~$
```

TO CONVERT .PEM FILE TO .PPK FILE

```
rohitha@rohitha-VirtualBox: ~$ cd Downloads
rohitha@rohitha-VirtualBox: ~/Downloads$ puttygen Facedetection.pem -O private -o
  Facedetection.ppk
rohitha@rohitha-VirtualBox: ~/Downloads$
```

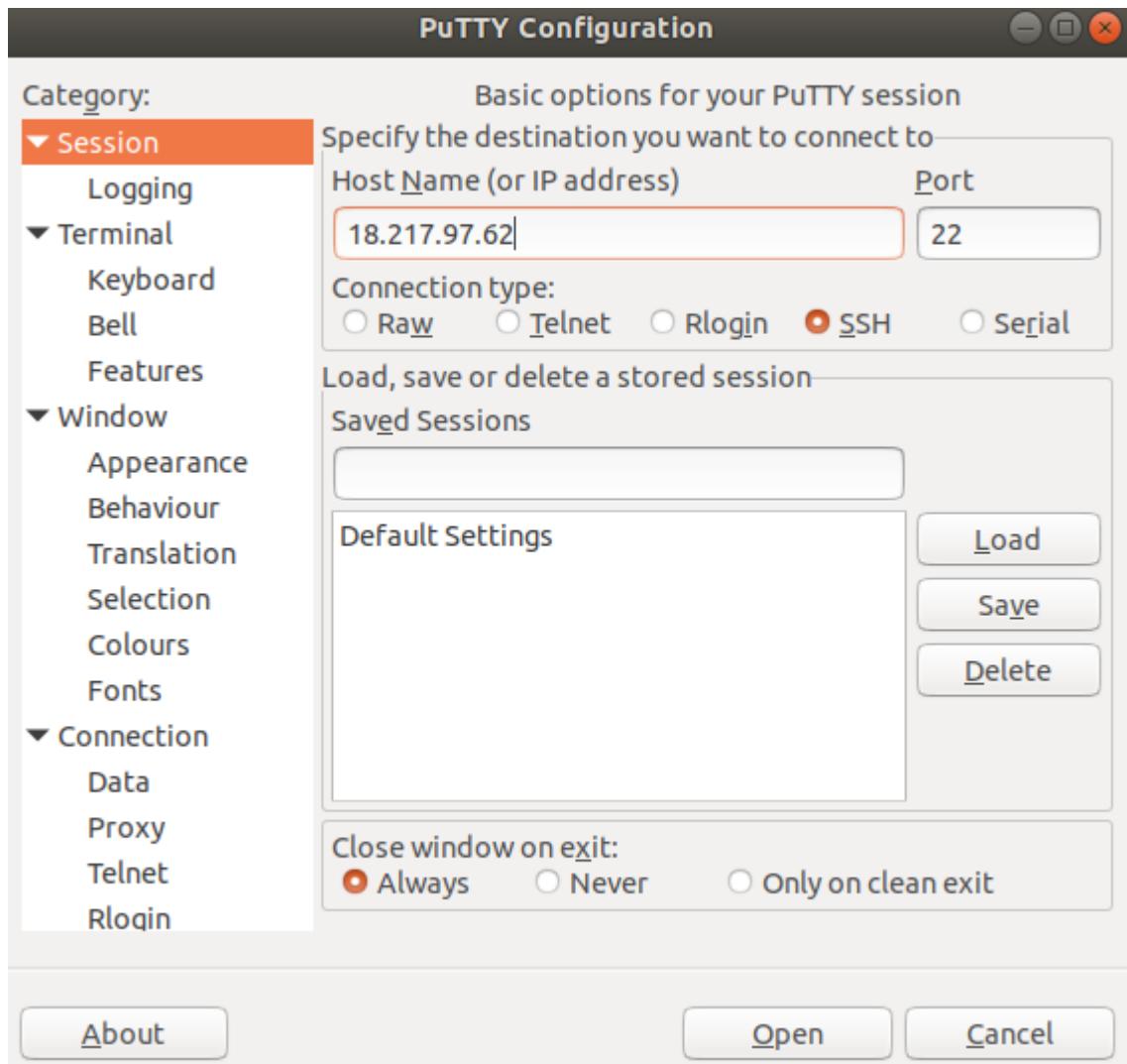
Open putty:



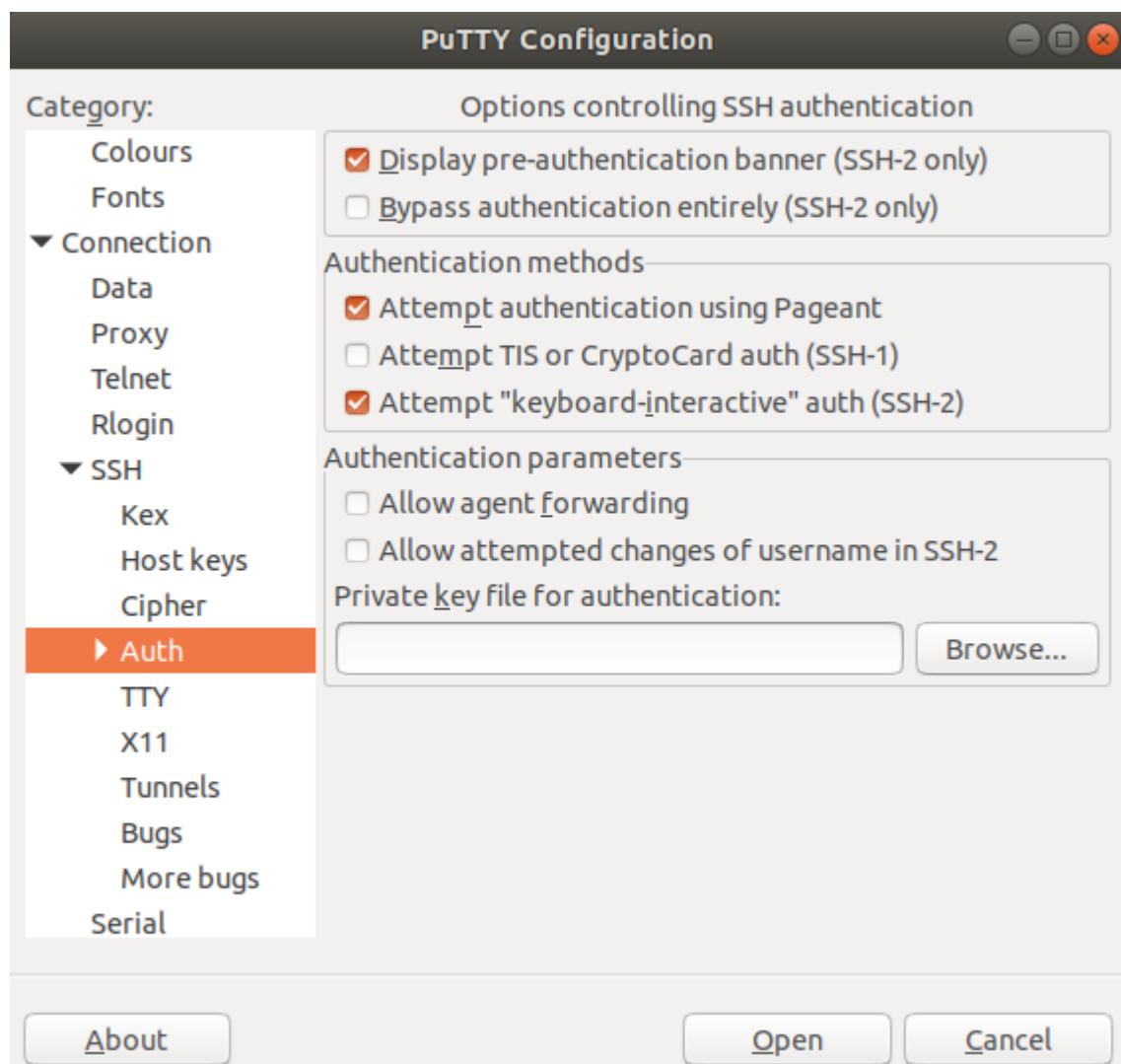
Make sure Port :22

Connection type: SSH

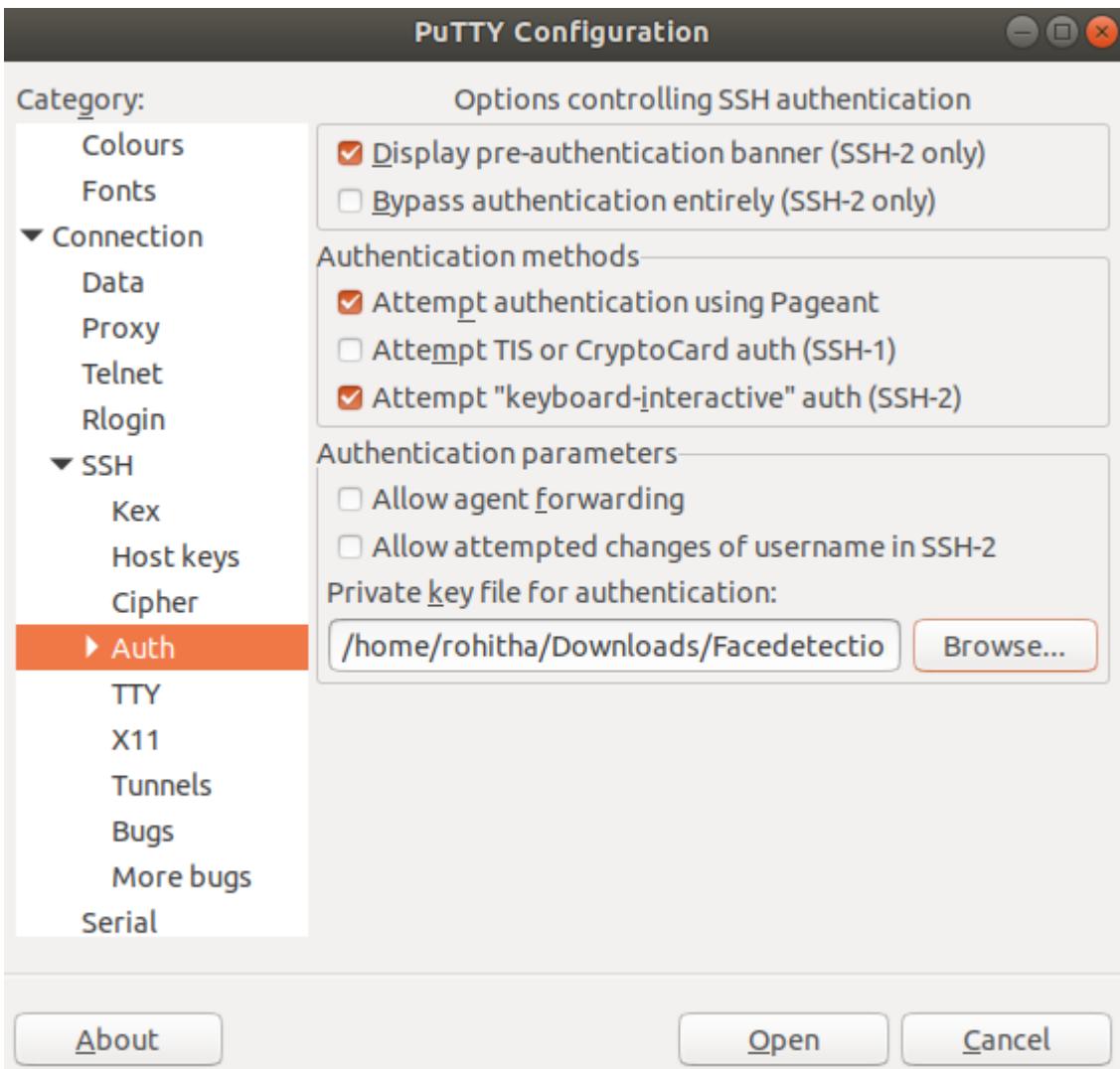
Enter IP address:



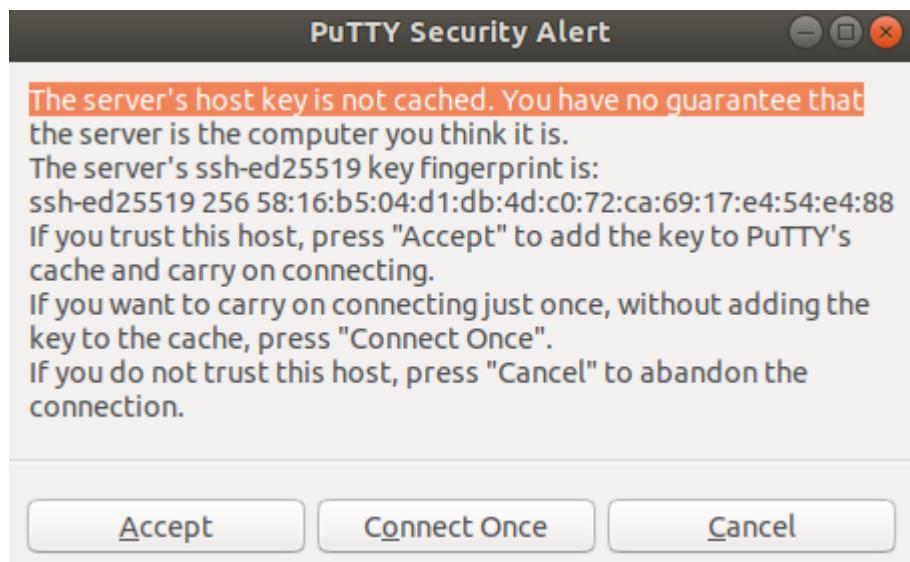
click Connection -> SSH -> auth



click browse -> downloads and select ppk file

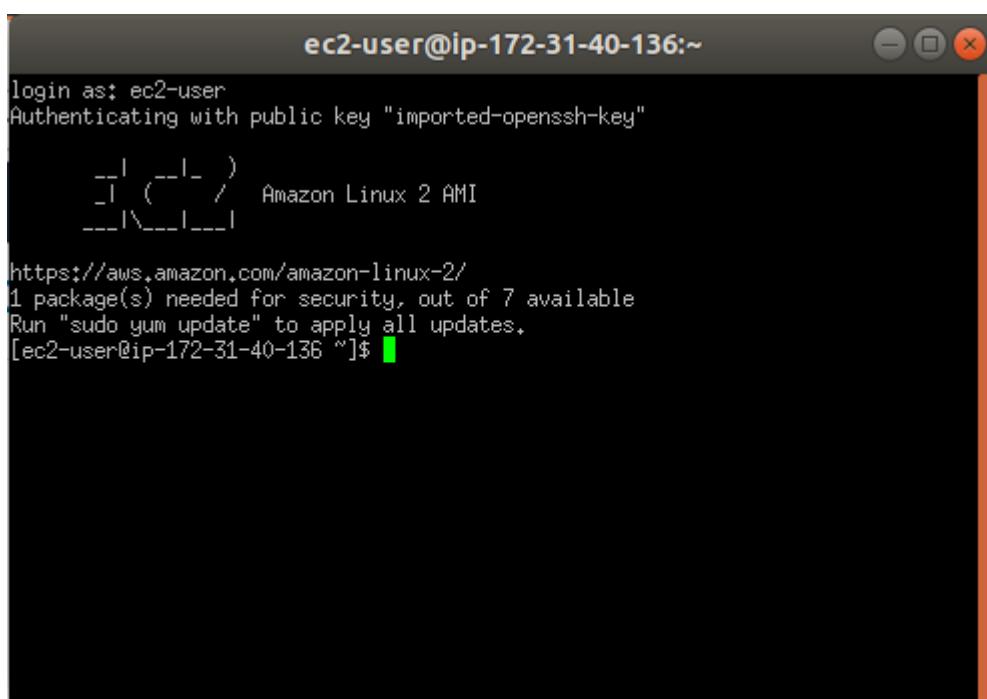


click open



click accept

LOGGED INTO EC2 :



The screenshot shows a terminal window titled "ec2-user@ip-172-31-40-136:~". The session starts with a message: "login as: ec2-user" followed by "Authenticating with public key "imported-openssh-key"". Below this, the Amazon Linux logo is displayed with the text "Amazon Linux 2 AMI". A URL "https://aws.amazon.com/amazon-linux-2/" is shown, along with a note about package updates: "1 package(s) needed for security, out of 7 available" and "Run "sudo yum update" to apply all updates.". The command "[ec2-user@ip-172-31-40-136 ~]\$" is visible at the bottom of the terminal.

Installing server:

will be installing httpd server that will be on virtual machine

1) sudo yum install httpd

```
ec2-user@ip-172-31-40-136:~  
Run "sudo yum update" to apply all updates.  
[ec2-user@ip-172-31-40-136 ~]$ sudo yum install httpd  
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd  
amzn2-core | 2.4 kB 00:00  
Resolving Dependencies  
--> Running transaction check  
--> Package httpd.x86_64 0:2.4.41-1.amzn2.0.1 will be installed  
--> Processing Dependency: httpd-tools = 2.4.41-1.amzn2.0.1 for package: httpd-2  
.4.41-1.amzn2.0.1.x86_64  
--> Processing Dependency: httpd-filesystem = 2.4.41-1.amzn2.0.1 for package: ht  
tpd-2.4.41-1.amzn2.0.1.x86_64  
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.41-1.amzn2.  
0.1.x86_64  
--> Processing Dependency: mod_http2 for package: httpd-2.4.41-1.amzn2.0.1.x86_6  
4  
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.41-1.amzn2.0.  
1.x86_64  
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.41-1.amzn2.0.1  
.x86_64  
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.41-  
1.amzn2.0.1.x86_64  
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.41-1.am  
zn2.0.1.x86_64  
--> Running transaction check  
--> Package apr.x86_64 0:1.6.3-5.amzn2.0.2 will be installed  
--> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed  
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package:  
apr-util-1.6.1-5.amzn2.0.2.x86_64  
--> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed  
--> Package httpd-filesystem.noarch 0:2.4.41-1.amzn2.0.1 will be installed  
--> Package httpd-tools.x86_64 0:2.4.41-1.amzn2.0.1 will be installed  
--> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed  
--> Package mod_http2.x86_64 0:1.15.3-2.amzn2 will be installed  
--> Running transaction check  
--> Package apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 will be installed  
--> Finished Dependency Resolution  
  
Dependencies Resolved  
  
=====  
Package           Arch    Version        Repository      Size  
=====  
Installing:  
httpd            x86_64  2.4.41-1.amzn2.0.1  amzn2-core   1.3 M  
Installing for dependencies:  
apr              x86_64  1.6.3-5.amzn2.0.2  amzn2-core   118 k  
apr-util         x86_64  1.6.1-5.amzn2.0.2  amzn2-core   99 k  
apr-util-bdb    x86_64  1.6.1-5.amzn2.0.2  amzn2-core   19 k  
generic-logos-httpd noarch 18.0.0-4.amzn2  amzn2-core   19 k  
httpd-filesystem noarch 2.4.41-1.amzn2.0.1  amzn2-core   23 k  
httpd-tools      x86_64  2.4.41-1.amzn2.0.1  amzn2-core   87 k  
mailcap          noarch  2.1.41-2.amzn2     amzn2-core   31 k  
mod_http2        x86_64  1.15.3-2.amzn2    amzn2-core  146 k  
  
Transaction Summary  
=====  
Install 1 Package (+8 Dependent packages)  
  
Total download size: 1.8 M  
Installed size: 5.1 M  
Is this ok [y/d/N]: 
```

Click y

```

Is this ok [y/d/N]: y
Downloading packages:
(1/9): apr-util-1.6.1-5.amzn2.0.2.x86_64.rpm | 99 kB 00:00:00
(2/9): apr-1.6.3-5.amzn2.0.2.x86_64.rpm | 118 kB 00:00:00
(3/9): apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64.rpm | 19 kB 00:00:00
(4/9): generic-logos-htpd-18.0.0-4.amzn2.noarch.rpm | 19 kB 00:00:00
(5/9): httpd-filesystem-2.4.41-1.amzn2.0.1.noarch.rpm | 23 kB 00:00:00
(6/9): httpd-2.4.41-1.amzn2.0.1.x86_64.rpm | 1.3 MB 00:00:00
(7/9): httpd-tools-2.4.41-1.amzn2.0.1.x86_64.rpm | 87 kB 00:00:00
(8/9): mailcap-2.1.41-2.amzn2.noarch.rpm | 31 kB 00:00:00
(9/9): mod_http2-1.15.3-2.amzn2.x86_64.rpm | 146 kB 00:00:00
-----
Total                                         7.1 MB/s | 1.8 MB 00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : apr-1.6.3-5.amzn2.0.2.x86_64          1/9
  Installing : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64    2/9
  Installing : apr-util-1.6.1-5.amzn2.0.2.x86_64          3/9
  Installing : httpd-tools-2.4.41-1.amzn2.0.1.x86_64      4/9
  Installing : generic-logos-htpd-18.0.0-4.amzn2.noarch    5/9
  Installing : mailcap-2.1.41-2.amzn2.noarch            6/9
  Installing : httpd-filesystem-2.4.41-1.amzn2.0.1.noarch   7/9
  Installing : mod_http2-1.15.3-2.amzn2.x86_64           8/9
  Installing : httpd-2.4.41-1.amzn2.0.1.x86_64           9/9
  Verifying   : apr-util-1.6.1-5.amzn2.0.2.x86_64          1/9
  Verifying   : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64      2/9
  Verifying   : httpd-2.4.41-1.amzn2.0.1.x86_64           3/9
  Verifying   : httpd-filesystem-2.4.41-1.amzn2.0.1.noarch   4/9
  Verifying   : mod_http2-1.15.3-2.amzn2.x86_64           5/9
  Verifying   : apr-1.6.3-5.amzn2.0.2.x86_64             6/9
  Verifying   : mailcap-2.1.41-2.amzn2.noarch            7/9
  Verifying   : generic-logos-htpd-18.0.0-4.amzn2.noarch    8/9
  Verifying   : httpd-tools-2.4.41-1.amzn2.0.1.x86_64       9/9
-----
Installed:
  httpd.x86_64 0:2.4.41-1.amzn2.0.1

Dependency Installed:
  apr.x86_64 0:1.6.3-5.amzn2.0.2
  apr-util.x86_64 0:1.6.1-5.amzn2.0.2
  apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2
  generic-logos-htpd.noarch 0:18.0.0-4.amzn2
  httpd-filesystem.noarch 0:2.4.41-1.amzn2.0.1
  httpd-tools.x86_64 0:2.4.41-1.amzn2.0.1
  mailcap.noarch 0:2.1.41-2.amzn2
  mod_http2.x86_64 0:1.15.3-2.amzn2

Complete!
[ec2-user@ip-172-31-40-136 ~]$ 
```

Now, start the service

To start the service command:

`sudo service httpd start`

```
[ec2-user@ip-172-31-40-136 ~]$ sudo service httpd start
Redirecting to /bin/systemctl start httpd.service
[ec2-user@ip-172-31-40-136 ~]$ 
```

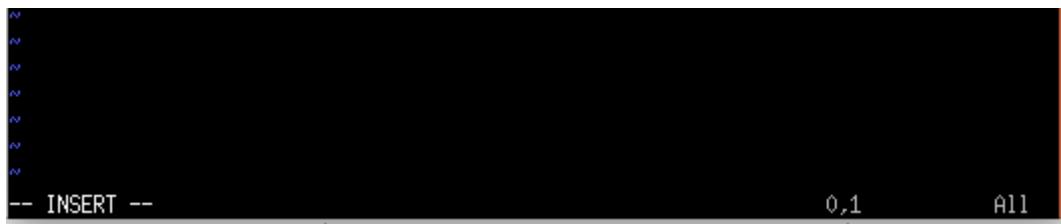
Now type command, sudo service httpd status
this is to make sure that whether the service got started or not

```
[ec2-user@ip-172-31-40-136 ~]$ sudo service httpd status
Redirecting to /bin/systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Thu 2020-03-26 16:15:16 UTC; 2min 26s ago
     Docs: man:httpd.service(8)
   Main PID: 3831 (httpd)
      Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
    CGroup: /system.slice/httpd.service
            ├─3831 /usr/sbin/httpd -DFOREGROUND
            ├─3832 /usr/sbin/httpd -DFOREGROUND
            ├─3833 /usr/sbin/httpd -DFOREGROUND
            ├─3834 /usr/sbin/httpd -DFOREGROUND
            ├─3835 /usr/sbin/httpd -DFOREGROUND
            └─3836 /usr/sbin/httpd -DFOREGROUND

Mar 26 16:15:16 ip-172-31-40-136.us-east-2.compute.internal systemd[1]: Starting Th...
Mar 26 16:15:16 ip-172-31-40-136.us-east-2.compute.internal systemd[1]: Started The...
Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-172-31-40-136 ~]$
```

now type command, sudo vim /var/www/html/index.html

Press i



Enter the text u wanted it to be printed on website

```
ec2-user@ip-172-31-40-136:~  
Hello I am Rohitha, here creating Face detection app.  
-- INSERT --
```

After competing ,
press Esc, then insert vanishes
press :wq
Hit enter, then will be back to previous window

```
ec2-user@ip-172-31-40-136:~
```

Total 7.1 MB/s | 1.8 MB 00:00

Running transaction check

Running transaction test

Transaction test succeeded

Running transaction

Action	Package	Status
Installing :	apr-1.6.3-5.amzn2.0.2.x86_64	1/9
Installing :	apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64	2/9
Installing :	apr-util-1.6.1-5.amzn2.0.2.x86_64	3/9
Installing :	httpd-tools-2.4.41-1.amzn2.0.1.x86_64	4/9
Installing :	generic-logos-httpd-18.0.0-4.amzn2.noarch	5/9
Installing :	mailcap-2.1.41-2.amzn2.noarch	6/9
Installing :	httpd-filesystem-2.4.41-1.amzn2.0.1.noarch	7/9
Installing :	mod_http2-1.15.3-2.amzn2.x86_64	8/9
Installing :	httpd-2.4.41-1.amzn2.0.1.x86_64	9/9
Verifying :	apr-util-1.6.1-5.amzn2.0.2.x86_64	1/9
Verifying :	apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64	2/9
Verifying :	httpd-2.4.41-1.amzn2.0.1.x86_64	3/9
Verifying :	httpd-filesystem-2.4.41-1.amzn2.0.1.noarch	4/9
Verifying :	mod_http2-1.15.3-2.amzn2.x86_64	5/9
Verifying :	apr-1.6.3-5.amzn2.0.2.x86_64	6/9
Verifying :	mailcap-2.1.41-2.amzn2.noarch	7/9
Verifying :	generic-logos-httpd-18.0.0-4.amzn2.noarch	8/9
Verifying :	httpd-tools-2.4.41-1.amzn2.0.1.x86_64	9/9

Installed:

httpd.x86_64 0:2.4.41-1.amzn2.0.1

Dependency Installed:

apr.x86_64 0:1.6.3-5.amzn2.0.2
apr-util.x86_64 0:1.6.1-5.amzn2.0.2
apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2
generic-logos-httpd.noarch 0:18.0.0-4.amzn2
httpd-filesystem.noarch 0:2.4.41-1.amzn2.0.1
httpd-tools.x86_64 0:2.4.41-1.amzn2.0.1
mailcap.noarch 0:2.1.41-2.amzn2
mod_http2.x86_64 0:1.15.3-2.amzn2

Complete!

```
[ec2-user@ip-172-31-40-136 ~]$ sudo service httpd start
Redirecting to /bin/systemctl start httpd.service
[ec2-user@ip-172-31-40-136 ~]$ sudo service httpd status
Redirecting to /bin/systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
     Active: active (running) since Thu 2020-03-26 16:15:16 UTC; 2min 26s ago
       Docs: man:httpd.service(8)
     Main PID: 3831 (httpd)
        Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
    CGroup: /system.slice/httpd.service
            └─3831 /usr/sbin/httpd -DFOREGROUND
              ├─3832 /usr/sbin/httpd -DFOREGROUND
              ├─3833 /usr/sbin/httpd -DFOREGROUND
              ├─3834 /usr/sbin/httpd -DFOREGROUND
              ├─3835 /usr/sbin/httpd -DFOREGROUND
              ├─3836 /usr/sbin/httpd -DFOREGROUND

Mar 26 16:15:16 ip-172-31-40-136.us-east-2.compute.internal systemd[1]: Starting Th...
Mar 26 16:15:16 ip-172-31-40-136.us-east-2.compute.internal systemd[1]: Started The...
Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-172-31-40-136 ~]$ sudo vim /var/www/html/index.html
[ec2-user@ip-172-31-40-136 ~]$
```

Now, in aws copy IP address and put it on the browser

we create port no. 22 so, only that particular traffic will be entering into our machine but traffic which we are sending to machine is http traffic, so we should allow http now, go to security groups-> inbound rules

A screenshot of the AWS CloudWatch Metrics interface. At the top, it shows 'Private IPs' as 172.31.40.136 and 'Secondary private IPs' as vpc-427bb129. Below that, 'Subnet ID' is listed as subnet-a743d0eb. On the right, there's a callout box titled 'Security Groups associated with i-0ad69d15508c44475' containing a table with one row:

Ports	Protocol	Source	Action
22	tcp	0.0.0.0/0	✓

port 22 is allowed but http is not allowed

port number of http is 80

so, to add http click security groups – launchwizard2

A screenshot of the AWS EC2 Security Groups page. The left sidebar shows navigation options like EC2 Dashboard, Instances, Images, and Elastic Block Store. The main area shows a table for 'Security Groups (1/1)'. One row is listed:

Security group ID	Security group name	VPC ID	Description
sg-04e508fea388873d9	launch-wizard-2	vpc-427bb129	launch-wizard-2 create..

Below the table, a note says 'sg-04e508fea388873d9 - launch-wizard-2'. At the bottom, there are tabs for Details, Inbound rules, Outbound rules, and Tags.

select security group

EC Management Console Problem loading page Problem loading page + https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#Security Services Resource Groups sg-04e508fea388873d9 - launch-wizard-2 Delete security group Copy to new security group

New EC2 Experience Tell us what you think

EC2 Dashboard New

- Events New
- Tags
- Reports
- Limits

INSTANCES

- Instances
- Instance Types
- Launch Templates New
- Spot Requests
- Savings Plans
- Reserved Instances
- Dedicated Hosts New
- Capacity Reservations

IMAGES

- AMIs
- Bundle Tasks

ELASTIC BLOCK STORE

Feedback English (US) © 2008 - 2020, Amazon Internet Services Private Ltd, or its affiliates. All rights reserved. Privacy Policy Terms of Use

click edit inbound rules

aws Services Resource Groups sg-04e508fea388873d9 - launch-wizard-2 Edit inbound rules

Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Type	Protocol	Port range	Source	Description - optional
SSH	TCP	22	Custom 0.0.0.0/0	

Add rule

⚠ NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Preview changes Save rules

click add rule

Edit inbound rules [Info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules [Info](#)

Type [Info](#) Protocol [Info](#) Port range [Info](#) Source [Info](#) Description - optional [Info](#)

SSH	TCP	22	Custom ▾	<input type="text"/> 0.0.0.0/0 X	Delete
HTTP	TCP	80	Anywhere ▾	<input type="text"/> 0.0.0.0/0 X	Delete

[Add rule](#)

click save rules

Details

Security group name launch-wizard-2	Security group ID sg-04e508fea388873d9	Description launch-wizard-2 created 2020-03-26T20:23:34.349+0 5:30 Edit	VPC ID vpc-427bb129
--	---	--	--

Owner 481453507501	Inbound rules count 3 Permission entries	Outbound rules count 1 Permission entry
-----------------------	---	--

[Inbound rules](#)

[Outbound rules](#)

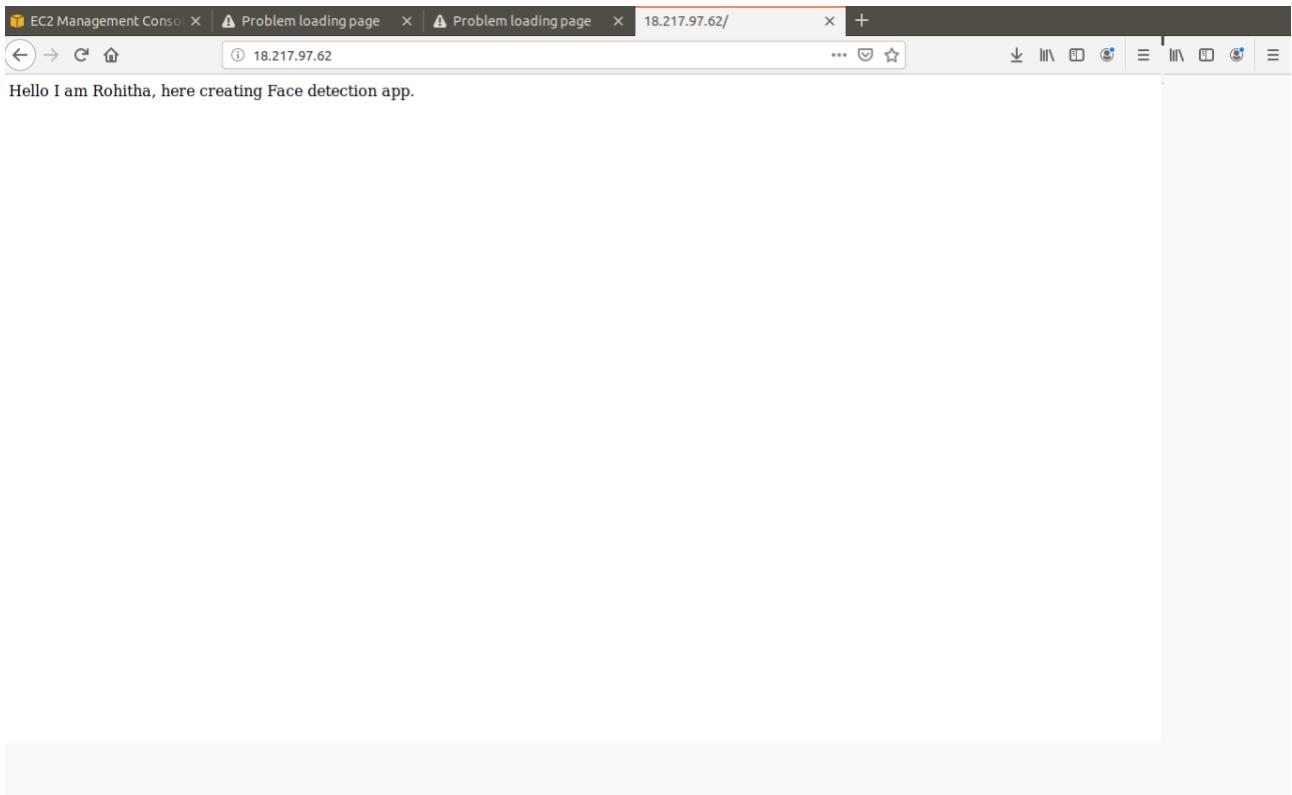
[Tags](#)

Inbound rules

[Edit inbound rules](#)

Type	Protocol	Port range	Source	Description - optional
HTTP	TCP	80	0.0.0.0/0	-
HTTP	TCP	80	::/0	-
SSH	TCP	22	0.0.0.0/0	-

once done copy the IP address and browse



PART-2

S3: Storage service which will store the image.

STEPS:

- 1)Terminate your EC2 instance (Web Server)
- 2)Create an S3 Bucket.
- 3)Working with S3 Buckets.
- 4)S3 Tiering.
- 5)Hosting a static website.

3 important keywords in EC2 dashboard

1)Terminate: Completely shut down the instance and take that instance away from you; you cannot again restart that instance.

2)Reboot: (Restarting), switch off and switch it on again.

Aws billing will not get stopped.

3)Stop: Different from terminate, will just stop the instance meaning just shut down in case if we need, we can restart that instance.

We will not be billed for stop instance, in case if we need, we can start it again.

We can't connect to stopped instance.

After terminating,

we get warning, this means as we have created the instance there was a point where we need to configure our storage, so we asked for 8GB of storage, when we are terminating this instance that storage will also be deleted.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, Instances (with sub-links for Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations), Images (AMIs, Bundle Tasks), and Elastic Block Store (Volumes, Snapshots). The main content area has tabs for Launch Instance, Connect, and Actions. A search bar at the top right says "Filter by tags and attributes or search by keyword". Below it is a table with columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS (IPv4), and IP. Two rows are visible: one for an instance named i-0ad69d15508c44475 which is running in us-east-2c, and another for an instance named i-0e3765d82d6152ae3 which is terminated in us-east-2a. At the bottom, there's a detailed view for the terminated instance, showing its ID (i-0e3765d82d6152ae3), state (terminated), type (t2.micro), and public DNS (IPv4) which is empty.

S3: Simple Storage Service

It is a storage service for the internet provided by AWS.
It is designated to make web-scale computing easy.
We can only store objects in S3 (file refers to object).

Bucket:

Container for objects.

Object:

Actual data and metadata

KEYS:

Unique identifier for an object.

Go to services -> S3

The screenshot shows the AWS Services Catalog interface. At the top, there's a search bar with placeholder text "Find a service by name or feature (for example, EC2, S3 or VM, storage)." Below the search bar are two buttons: "Group" and "A-Z". The main area is divided into several sections:

- Compute**: EC2, Lightsail, Lambda, Batch, Elastic Beanstalk, Serverless Application Repository, AWS Outposts, EC2 Image Builder.
- Blockchain**: Amazon Managed Blockchain.
- Analytics**: Athena, EMR, CloudSearch, Elasticsearch Service, Kinesis, QuickSight, Data Pipeline, AWS Data Exchange, AWS Glue, AWS Lake Formation, MSK.
- End User Computing**: WorkSpaces, AppStream 2.0, WorkDocs, WorkLink.
- Storage**: S3, EFS, FSx, S3 Glacier, Storage Gateway, AWS Backup.
- Management & Governance**: AWS Organizations, CloudWatch, AWS Auto Scaling, CloudFormation, CloudTrail, Config, OpsWorks, Service Catalog, Systems Manager, AWS AppConfig, Trusted Advisor.
- Security, Identity, & Compliance**: IAM, Resource Access Manager, Cognito, Secrets Manager, GuardDuty, Inspector, Amazon Macie, AWS Single Sign-On.
- Internet Of Things**: IoT Core, FreeRTOS, IoT 1-Click, IoT Analytics, IoT Device Defender, IoT Device Management, IoT Events, IoT Greengrass, IoT SiteWise, IoT Things Graph.
- Database**: RDS, DynamoDB, ElastiCache.
- Game Development**: Amazon GameLift.

The screenshot shows the Amazon S3 console. At the top, there's a message: "We're gradually updating the design of the Amazon S3 console. You will notice some updated screens as we improve the performance and user interface. To help us improve the experience, [give feedback](#) on the recent updates." Below this message, the main interface is titled "Amazon S3". On the left, there's a sidebar with links: "Buckets", "Batch operations", "Access analyzer for S3", "Block public access (account settings)", and "Feature spotlight". The main content area is titled "Buckets (0)". It features a search bar with placeholder text "Find bucket by name" and a "Create bucket" button. A table header includes columns for "Name", "Region", "Access", and "Bucket created". Below the table, a message says "No buckets" and "You don't have any buckets." with a "Create bucket" button.

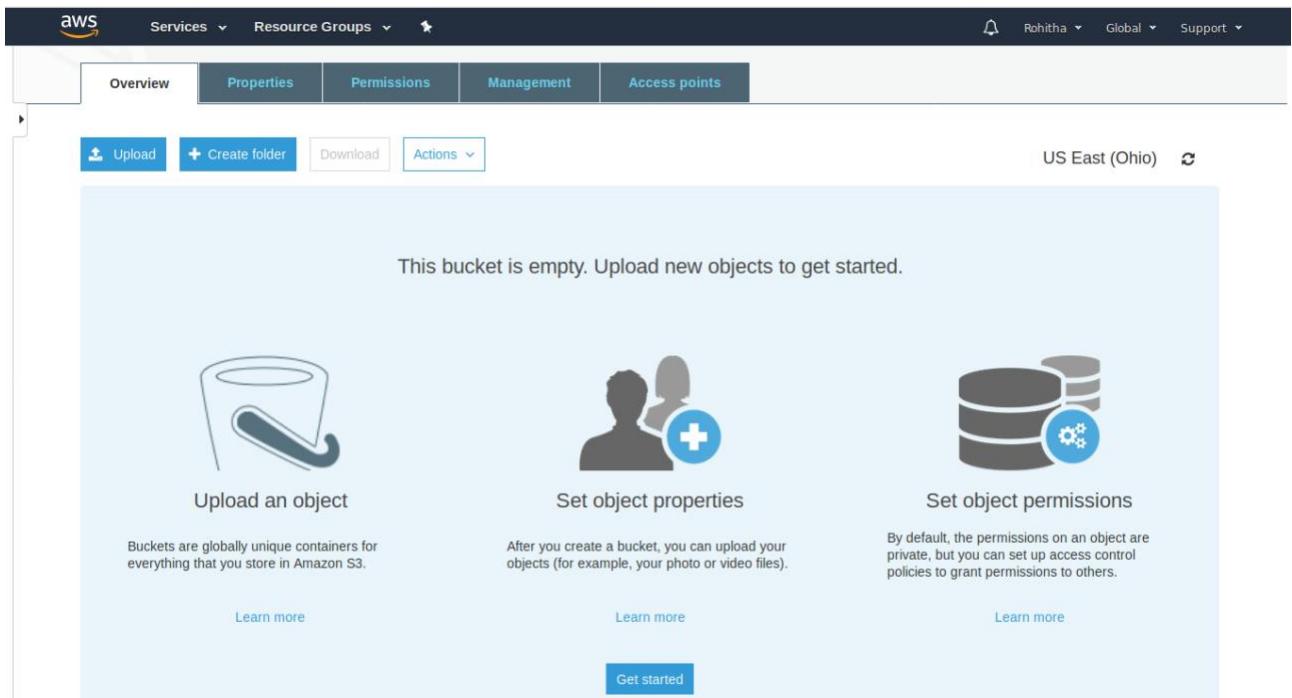
choose create bucket
name the bucket with a unique name
and click create bucket

CREATING BUCKET:

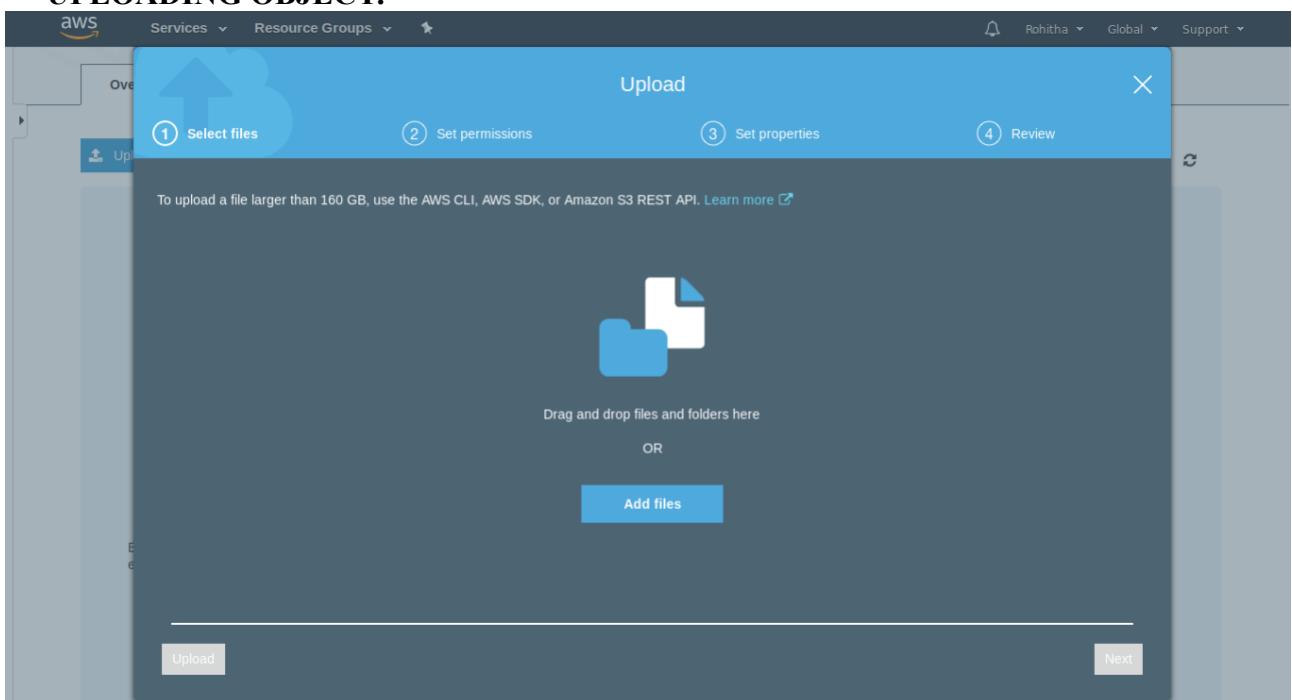
The screenshot shows the 'Create bucket' page in the AWS S3 console. In the 'General configuration' section, the 'Bucket name' field contains 'facedetection15'. A note below it states: 'Bucket name must be unique and must not contain spaces or uppercase letters. See rules for bucket naming'. The 'Region' dropdown is set to 'US East (Ohio) us-east-2'. Below this, the 'Bucket settings for Block Public Access' section is expanded, showing the 'Block all public access' checkbox is checked. A note explains: 'Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.' Underneath are two collapsed sections: 'Block public access to buckets and objects granted through new access control lists (ACLs)' and 'Block public access to buckets and objects granted through old access control lists (ACLs)'.

The screenshot shows the 'Buckets' page in the AWS S3 console. A green banner at the top indicates: 'Successfully created bucket facedetection15. To upload files and folders, or to configure additional bucket settings such as Bucket Versioning, tags, and default encryption, choose Go to bucket details.' On the left sidebar, there are links for 'Batch operations', 'Access analyzer for S3', 'Block public access (account settings)', and 'Feature spotlight'. The main content area displays a table titled 'Buckets (1)'. The table has columns: Name, Region, Access, and Bucket created. It shows one entry: 'facedetection15' (Region: US East (Ohio) us-east-2, Access: Not Public, Bucket created: 2020-03-27T08:42:18.000Z). Action buttons 'Copy ARN', 'Empty', 'Delete', and 'Create bucket' are located at the top right of the table.

S3 does not require region selection, whereas bucket creation requires region selection
click the bucket created

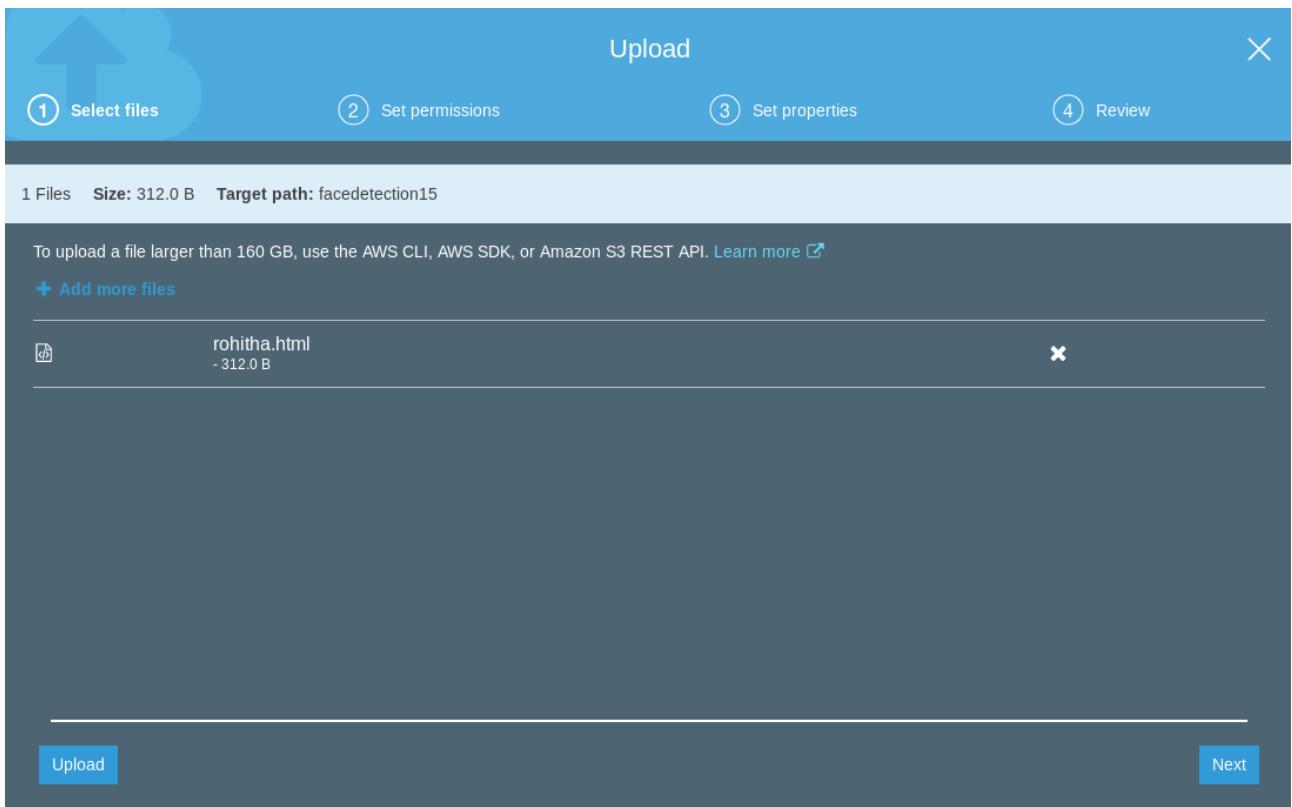


click upload
UPLOADING OBJECT:



click add files
create a .html file

```
rohitha@rohitha-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
rohitha@rohitha-VirtualBox:~$ cd Desktop
rohitha@rohitha-VirtualBox:~/Desktop$ gedit rohitha.html &
[1] 23791
rohitha@rohitha-VirtualBox:~/Desktop$
```



click next

Upload

(1) Select files (2) Set permissions (3) Set properties (4) Review

1 Files Size: 312.0 B Target path: facedetection15

Manage users

User ID	Objects	Object permissions
avula.rohitha2017(Owner)	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Read <input checked="" type="checkbox"/> Write

Access for other AWS account [+ Add account](#)

Account	Objects	Object permissions
---------	---------	--------------------

Manage public permissions

The block public access settings turned on for this bucket prevent granting public access.

Do not grant public read access to this object(s) (Recommended)

[Upload](#) [Previous](#) [Next](#)

click next

Tiers will be present in the below page.

Upload

(1) Select files (2) Set permissions (3) Set properties (4) Review

1 Files Size: 312.0 B Target path: facedetection15

Storage class

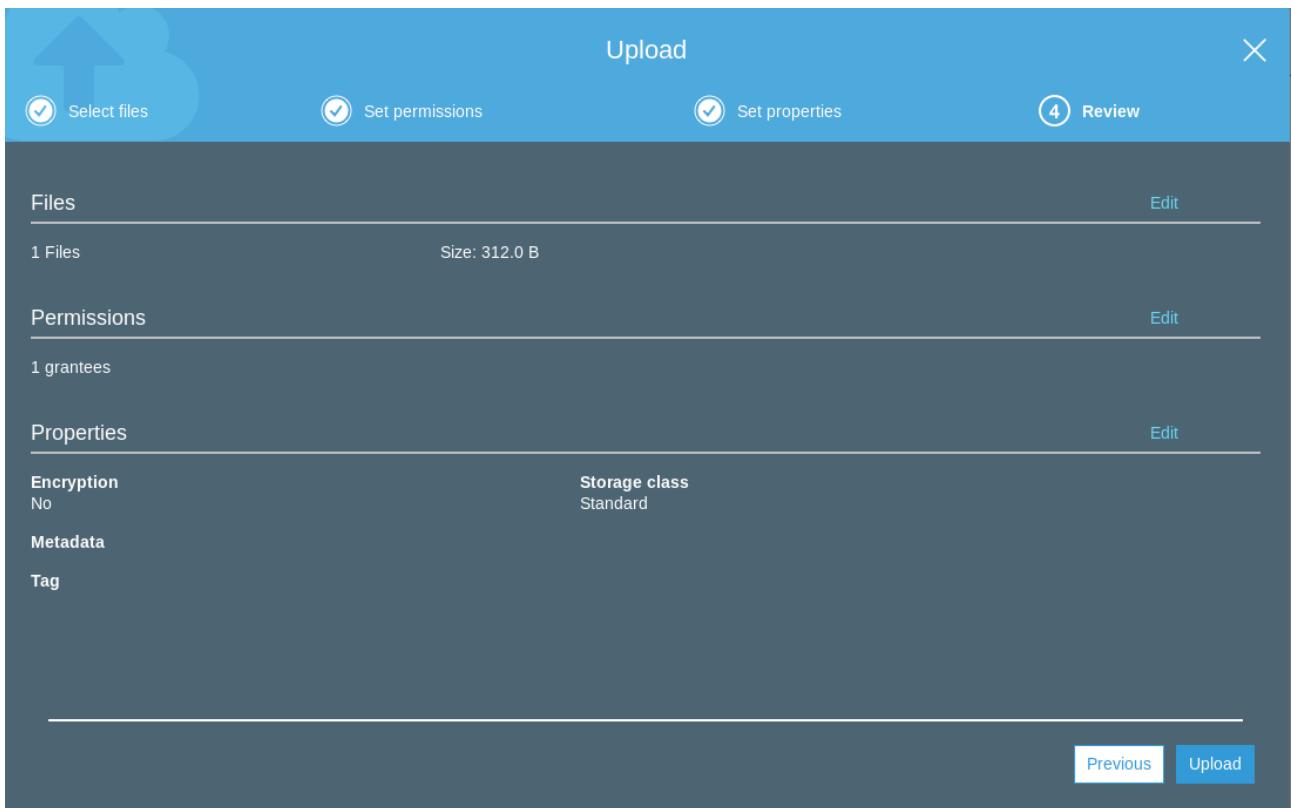
Choose a storage class based on your use case and access requirements. [Learn more](#) or see [Amazon S3 pricing](#)

Storage class	Designed for	Availability Zones	Min storage duration	Min billable object size	Monitoring and automation fees	Retrieval fees
<input checked="" type="radio"/> Standard	Frequently accessed data	≥ 3	-	-	-	-
<input type="radio"/> Intelligent-Tiering	Long-lived data with changing or unknown access patterns	≥ 3	30 days	-	Per-object fees apply	-
<input type="radio"/> Standard-IA	Long-lived, infrequently accessed data	≥ 3	30 days	128KB	-	Per-GB fees apply
<input type="radio"/> One Zone-IA	Long-lived, infrequently accessed, non-critical data	≥ 1	30 days	128KB	-	Per-GB fees apply
<input type="radio"/> Glacier	Archive data with retrieval times ranging from minutes to hours	≥ 3	90 days	40KB	-	Per-GB fees apply
<input type="radio"/> Glacier Deep Archive	Archive data that rarely, if ever, needs to be retrieved	≥ 3	180 days	40KB	-	Per-GB fees

[Upload](#) [Previous](#) [Next](#)

Make sure it is in standard

and click next



click upload
file gets uploaded

The screenshot shows the AWS S3 'Overview' page for a bucket. The top navigation bar includes the AWS logo, Services, Resource Groups, a search bar, and user information (Rohitha, Global, Support). Below the navigation is a toolbar with 'Upload', 'Create folder', 'Download', and 'Actions' dropdown. The main content area displays a table of objects. The table has columns for Name, Last modified, Size, and Storage class. One object, 'rohitha.html', is listed with the details: Last modified Mar 27, 2020 2:39:38 PM GMT+0530, Size 312.0 B, and Storage class Standard. At the bottom, a progress bar shows 0 In progress, 1 Success, and 0 Error.

Name	Last modified	Size	Storage class
rohitha.html	Mar 27, 2020 2:39:38 PM GMT+0530	312.0 B	Standard

1 success, 0 Error indicates there is no error

OBJECT:

It is a fundamental entity stored in Amazon S3.
Object is data and metadata

Data is opaque
It is name and value pack
It also stores metadata (data last modified, and standard HTTP metadata)

click the html file

rohitaha.html Latest version

Overview Properties Permissions Select from

Open Download Download as Make public Copy path

Owner
c4686f7b3e0a9cd4c5bd2d936676bebb04b4a4594d10e5ee4d2bc619d6ba5505

Last modified
Mar 27, 2020 2:39:38 PM GMT+0530

Etag
a03c53b5ca95acf1424e4008001b4c69

Storage class
Standard

Server-side encryption
None

Size
312.0 B

Key
rohitaha.html

Object URL
<https://facedetection15.s3.us-east-2.amazonaws.com/rohitaha.html>

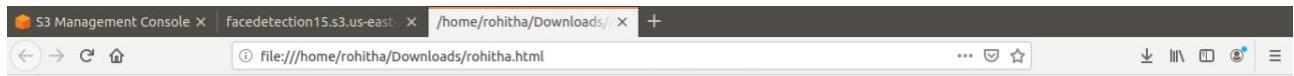
Open the URL in new tab

S3 Management Console X facedetection15.s3.us-east- X +

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<Error>
<Code>AccessDenied</Code>
<Message>Access Denied</Message>
<RequestId>27518F56C4B5C1F7</RequestId>
<HostId>
xUrkvKtlyXmsd0AM32dm18sP4PS64nN7KU6YtbumB/FzvMG5rGvhvNDAUc3TrdPkBeXFyuZXrBrQ=
</HostId>
</Error>
```

To make the access connected we have to configure
Click download to download the html file
after downloading click the downloaded file



Hello I am Rohitha here, Creating Face detection app Everything you think..... everything you feel.... every action you feel.....is because of your beliefs and values. you, may not be aware of it, you may not be conscious of it, but your values shape your decisions....and your decisions shape your life.

REGION:

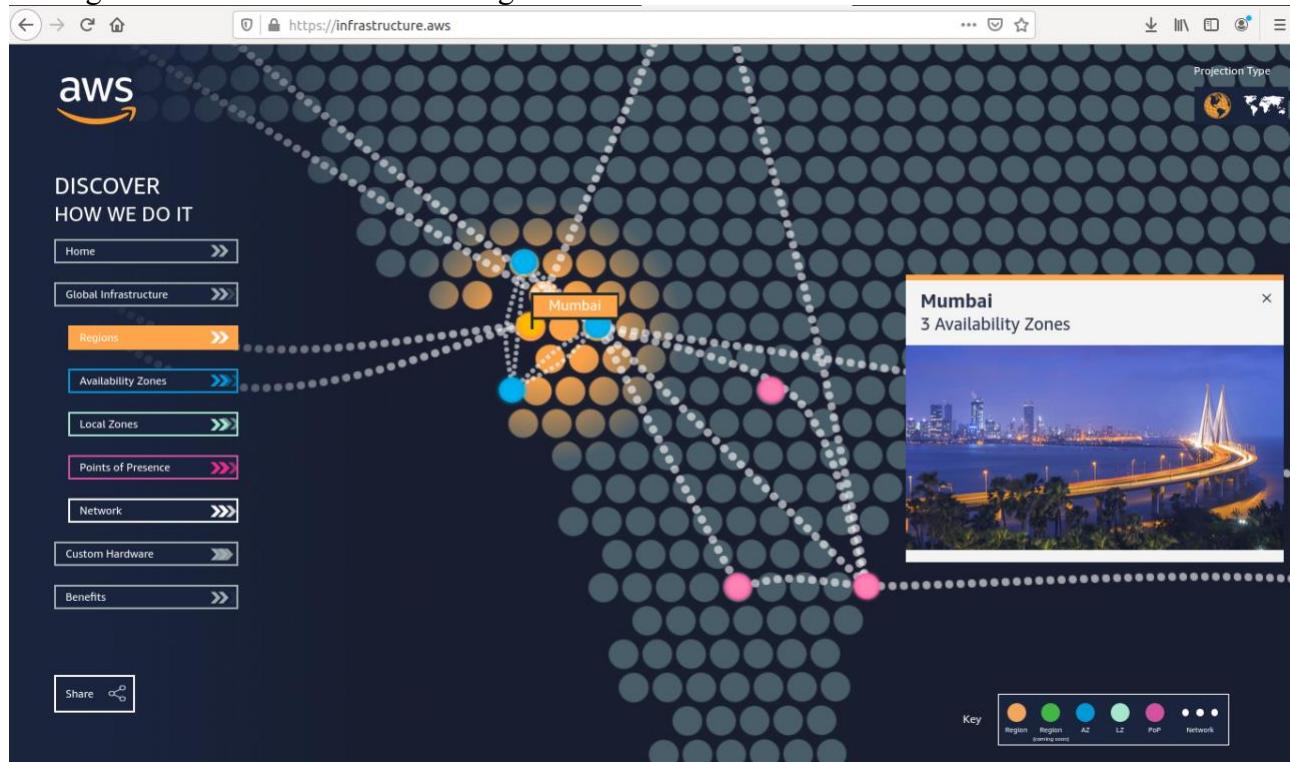
Regions, availability zones, edge locations these are three important entities of an aws infrastructure. Aws operates these 3 important entities.

Region is geographical location.

Data centres are place where all of the servers are physically located.

Availability zone is within a particular region. To increase amazons network availability.

Edge locations are used for crashing the file.



S3 Tiering:

- 1)Amazon S3 standard: Accessed very frequently
- 2)Amazon S3 standard-Infrequent access: Ability to make that object infrequent and pay less amount to amazon. Availability will be in 3 zones/regions.
- 3)Amazon S3 One zone-infrequent Access: Availability will only be in one zone. Less amount when compared to Amazon S3 standard-Infrequent access.
- 4)Amazon S3 glacier: After putting an object we sometime forget about it; it becomes cold storage.

The screenshot shows the AWS S3 console with a bucket named 'facedetection15'. In the main list, there is one object named 'rohitha.html'. A modal window is open for this object, displaying its properties. The modal has tabs for 'Overview', 'Properties', and 'Permissions'. The 'Overview' tab shows the following details:

	Value
Key	rohitha.html
Size	312.0 B
Expiration date	N/A
Expiration rule	N/A
ETag	a03c53b5ca95acf1424e4008001b4c69
Last modified	Mar 27, 2020 2:39:38 PM GMT+0530
Object URL	https://facedetection15.s3.us-east-2.amazonaws.com/rohitha.html

The 'Properties' tab shows:

	Value
Storage class	Standard
Encryption	None
Metadata	1
Tags	0 Tags
Object lock	Disabled

The 'Permissions' tab shows:

	Value
Owner	[Owner Information]

Click properties

The screenshot shows the 'Properties' tab for the 'facedetection15' bucket. There are five sections displayed:

- Versioning**: Keep multiple versions of an object in the same bucket. Status: Disabled.
- Server access logging**: Set up access log records that provide details about access requests. Status: Disabled.
- Static website hosting**: Host a static website, which does not require server-side technologies. Status: Disabled.
- Object-level logging**: Record object-level API activity using the CloudTrail data events feature (additional cost). Status: Disabled.
- Default encryption**: Automatically encrypt objects when stored in Amazon S3. Status: Disabled.

click on static website hosting

ENABLING STATIC WEBSITE:

The screenshot shows the AWS S3 console with the bucket 'facedetection15'. The 'Static website hosting' dialog is open, displaying the endpoint <http://facedetection15.s3-website.us-east-2.amazonaws.com>. The 'Use this bucket to host a website' option is selected. Other options like 'Redirect requests' and 'Disable website hosting' are available. The dialog includes 'Cancel' and 'Save' buttons.

click use this host as a website
give the name of the file in index document

The screenshot shows the AWS S3 console with the bucket 'facedetection15'. The 'Static website hosting' dialog is open, displaying the endpoint <http://facedetection15.s3-website.us-east-2.amazonaws.com>. The 'Use this bucket to host a website' option is selected. The 'Index document' field is set to 'rohitha.html' and the 'Error document' field is set to 'error.html'. The dialog includes 'Cancel' and 'Save' buttons.

click save

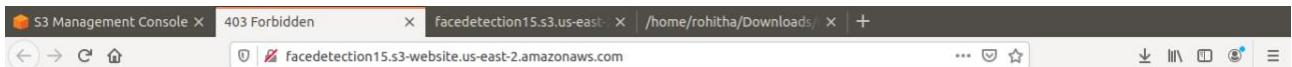
The screenshot shows the AWS S3 console with the 'Static website hosting' feature selected. The 'Bucket hosting' checkbox is checked. Other options like 'Versioning', 'Server access logging', and 'Default encryption' are also listed but disabled.

Advanced settings

click on static website hosting again,

The screenshot shows the 'Static website hosting' configuration dialog. It includes fields for 'Index document' (set to 'rohitaha.html') and 'Error document' (set to 'error.html'). The 'Bucket hosting' checkbox is checked. Buttons for 'Cancel' and 'Save' are at the bottom.

we can see an end point, click on the link



403 Forbidden

- Code: AccessDenied
- Message: Access Denied
- RequestId: 0BBA0CC1F01A38D3
- HostId: +66CTggKAW3gVUM/mWYw0M740Qr0kKLp6t5ChQlrUoZlYrXD8TE6P34NCsK6zGtjTvu/NqWQ4s=

To enable public access,
go to permissions,
MAKING THE OBJECT PUBLIC:

aws Services Resource Groups ★ Rohitha Global Support

facedetection15

Overview Properties Permissions Management Access points

Block public access Access Control List Bucket Policy CORS configuration

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access

On

- Block public access to buckets and objects granted through *new* access control lists (ACLS)
On
- Block public access to buckets and objects granted through *any* access control lists (ACLS)
On
- Block public access to buckets and objects granted through *new* public bucket or access point policies
On
- Block public and cross-account access to buckets and objects through *any* public bucket or access point policies
On

Edit

click edit

The screenshot shows the AWS S3 console with the bucket 'facedetection15'. The 'Management' tab is selected. Under the 'Block public access' section, several checkboxes are listed:

- Block all public access**: Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.
- Block public access to buckets and objects granted through new access control lists (ACLS)**: S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any access control lists (ACLS)**: S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through new public bucket or access point policies**: S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through any public bucket or access point policies**: S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

At the bottom right of the dialog are 'Cancel' and 'Save' buttons.

click save

Type confirm

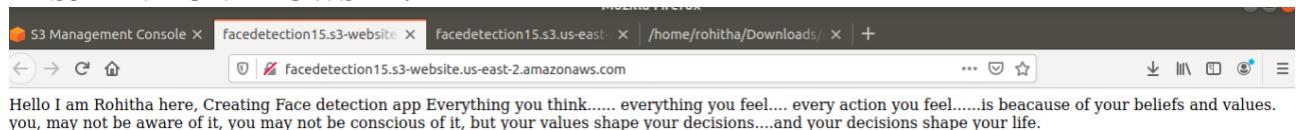
The dialog title is 'Edit block public access (bucket settings)'. It contains a message: 'Updating the block public access (bucket settings) will affect this bucket and all objects within. This may result in some objects becoming public.' Below the message is a text input field containing 'confirm'. At the bottom are 'Cancel' and 'Confirm' buttons.

The screenshot shows the AWS S3 console under the 'Permissions' tab. The 'Block public access' section is selected. A green success message box is displayed: 'Public access settings updated successfully'. Below it, the 'Block all public access' setting is set to 'Off'. Underneath, four specific options are listed, each also set to 'Off': 'Block public access to buckets and objects granted through new access control lists (ACLS)', 'Block public access to buckets and objects granted through any access control lists (ACLS)', 'Block public access to buckets and objects granted through new public bucket or access point policies', and 'Block public and cross-account access to buckets and objects through any public bucket or access point policies'.

The screenshot shows the AWS S3 console under the 'Properties' tab for the object 'rohitha.html'. The object was last modified on Mar 27, 2020 at 2:39:38 PM GMT+0530. It has an Etag of a03c53b5ca95acf1424e4008001b4c69 and a Storage class of Standard. The object is not encrypted. Its size is 312.0 B and its key is rohitha.html. The Object URL is https://facedetection15.s3.us-east-2.amazonaws.com/rohitha.html. Action buttons include Open, Download, Download as, Make public, and Copy path.

click make as public

S3 LINK ON BROWSER:



URL:

<http://facedetection15.s3-website.us-east-2.amazonaws.com/>

PART-3:

CONNECTING EC2 AND S3 AND GETTING RESPONSE BACK FROM S3

- 1)Introduction to VPC
- 2)How subnetting works
- 3)Uploading an object from EC2
- 4)Invoking recognition from EC2
- 5)Creating telegram Bot

VIRTUAL PRIVATE CLOUD

It gives provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define.

We can have complete control over our virtual networking environment.

Public cloud – Entire world can access it (Ex: Amazon)

Services -> Networking and content delivery -> VPC

AWS Services ▾ Resource Groups ▾

History

Console Home

S3

EC2

Find a service by name or feature (for example, EC2, S3 or VM, storage).

Group A-Z

Networking & Content Delivery

- VPC
- CloudFront
- Route 53
- API Gateway
- Direct Connect
- AWS App Mesh
- AWS Cloud Map
- Global Accelerator

Developer Tools

- CodeStar
- CodeCommit
- CodeBuild
- CodeDeploy
- CodePipeline
- Cloud9
- X-Ray

Machine Learning

- Amazon SageMaker
- Amazon CodeGuru
- Amazon Comprehend
- Amazon Forecast
- Amazon Fraud Detector
- Amazon Kendra
- Amazon Lex
- Amazon Machine Learning
- Amazon Personalize
- Amazon Polly
- Amazon Rekognition
- Amazon Textract
- Amazon Transcribe
- Amazon Translate
- AWS DeepLens
- AWS DeepRacer

Mobile Hub

- AWS AppSync
- Device Farm

AR & VR

- Amazon Sumerian

Elemental Appliances & Software

- Amazon Comprehend
- Amazon Forecast
- Amazon Fraud Detector
- Amazon Kendra
- Amazon Lex
- Amazon Machine Learning
- Amazon Personalize
- Amazon Polly
- Amazon Rekognition
- Amazon Textract
- Amazon Transcribe
- Amazon Translate

Application Integration

- Step Functions
- Amazon EventBridge
- Amazon MQ
- Simple Notification Service
- Simple Queue Service
- SWF

AWS Cost Management

- AWS Cost Explorer
- AWS Budgets
- AWS Marketplace Subscriptions

Customer Engagement

▲ close

Click VPC

The screenshot shows the AWS VPC Dashboard. On the left, there's a sidebar with navigation links for VPCs, Security, and Virtual Private Network (VPN). The main area displays 'Resources by Region' with a note that instances will launch in the US East (Ohio) region. It lists various VPC components with their counts in the Ohio region: VPCs (1), Subnets (3), Route Tables (1), Internet Gateways (1), NAT Gateways (0), Peering Connections (0), Security Groups (3), Customer Gateways (0), Virtual Private Gateways (0), Site-to-Site VPN Connections (0), and Elastic IPs (0). To the right, the 'Service Health' section shows a green status for Amazon EC2 - US East (Ohio) with the message 'Service is operating normally'. Below it are sections for 'Account Attributes' (Resource ID length management), 'Additional Information' (VPC Documentation, All VPC Resources, Forums, Report an Issue), and 'Transit Gateway Network Manager' (Network Manager enables centrally manage your global network across AWS and on-premises. Learn more, Get started with Network Manager). A 'Site-to-Site VPN Connections' section is also present.

click VPCs to see how many VPCs we have

This screenshot shows the AWS VPC Dashboard with a single VPC entry listed. The table header includes columns for Name, VPC ID, State, IPv4 CIDR, IPv6 CIDR, DHCP options set, and Main Route table. The entry for 'vpc-427bb129' is shown with the state 'available', IPv4 CIDR '172.31.0....', and Main Route table 'rtb-f2be2c99'. Below the table, a detailed view of the VPC 'vpc-427bb129' is provided, showing its configuration and settings. The VPC ID is vpc-427bb129, State is available, IPv4 CIDR is 172.31.0.0/16, IPv6 Pool is -, Tenancy is default, Default VPC is Yes, IPv6 CIDR is -, and DNS resolution is Enabled.

To create one more VPC click on create Vpc

Description:

Description		CIDR Blocks	Flow Logs	Tags	...
VPC ID	vpc-427bb129			Tenancy	default
State	available			Default VPC	Yes
IPv4 CIDR	172.31.0.0/16			IPv6 CIDR	-
IPv6 Pool	-			DNS resolution	Enabled
Network ACL	acl-24fe4e4f			DNS hostnames	Enabled
DHCP options set	dopt-6d6ec106			Route table	rtb-f2be2c99
Owner	481453507501				

CIDR Blocks:

Description		CIDR Blocks	Flow Logs	Tags	...						
IPv4 CIDR Blocks:											
<table border="1"><thead><tr><th>CIDR</th><th>Status</th><th>Status reason</th></tr></thead><tbody><tr><td>172.31.0.0/16</td><td>associated</td><td>-</td></tr></tbody></table>						CIDR	Status	Status reason	172.31.0.0/16	associated	-
CIDR	Status	Status reason									
172.31.0.0/16	associated	-									

One VPC will be divided into multiple subnets
Click on subnets will be able to see multiple subnets

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Ava
subnet-0aaa5761	subnet-0aaa5761	available	vpc-427bb129	172.31.0.0/20	4091	-	us-e
subnet-0e200b74	subnet-0e200b74	available	vpc-427bb129	172.31.16.0/20	4091	-	us-e
subnet-a743d0eb	subnet-a743d0eb	available	vpc-427bb129	172.31.32.0/20	4090	-	us-e

Dividing one network into multiple networks is called subnetting.

SUBNET:

Subnet is a logical subdivision of an IP network.

The practice of dividing a network into two or more networks is called subnetting.

Can EC2 upload objects to S3?

IAM:

1)Users: It can be humans, applications (one who consumes aws service). Users will be belonging to groups. One who has credentials

2)Roles: temporary position that we are holding so that we can do particular task and come and back.

3)Groups: To which group you belong to. Groups can be associated with policies.

4) Policies

EC2 will have role to access S3

Go to Ec2

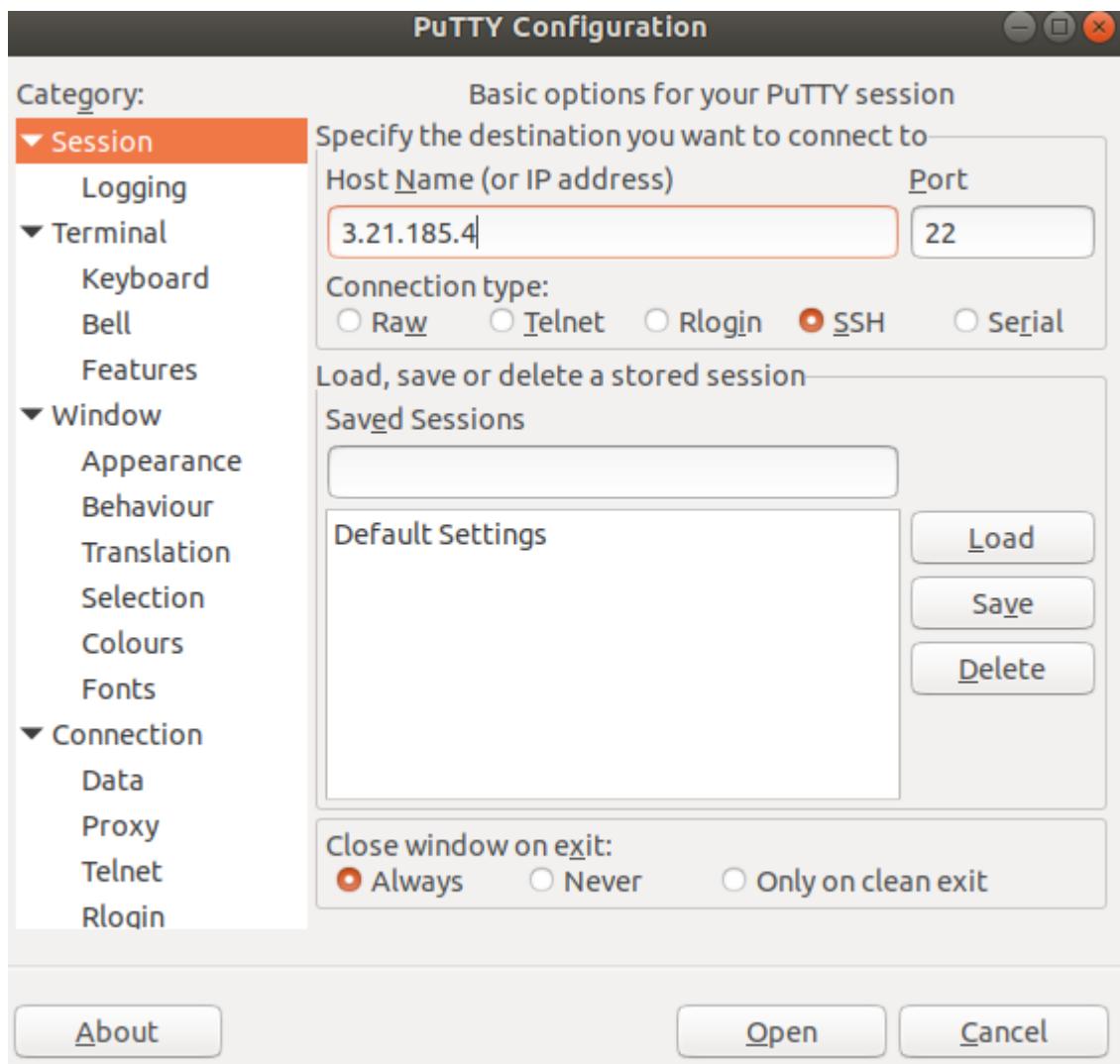
Create a new Instance

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, and Bundle Tasks. Below that is the Elastic Block Store section with Volumes and Snapshots. The main content area has a search bar and a table with two rows of instance data. A message at the bottom says "Select an instance above".

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IP
	i-0224356be073ccc91	t2.micro	us-east-2b	running	Initializing	None	ec2-3-21-185-4.us-east...	3.2
	i-0ad69d15508c44475	t2.micro	us-east-2c	running	2/2 checks ...	None	ec2-18-217-97-62.us-e...	18

copy the IP address: 3.21.185.4

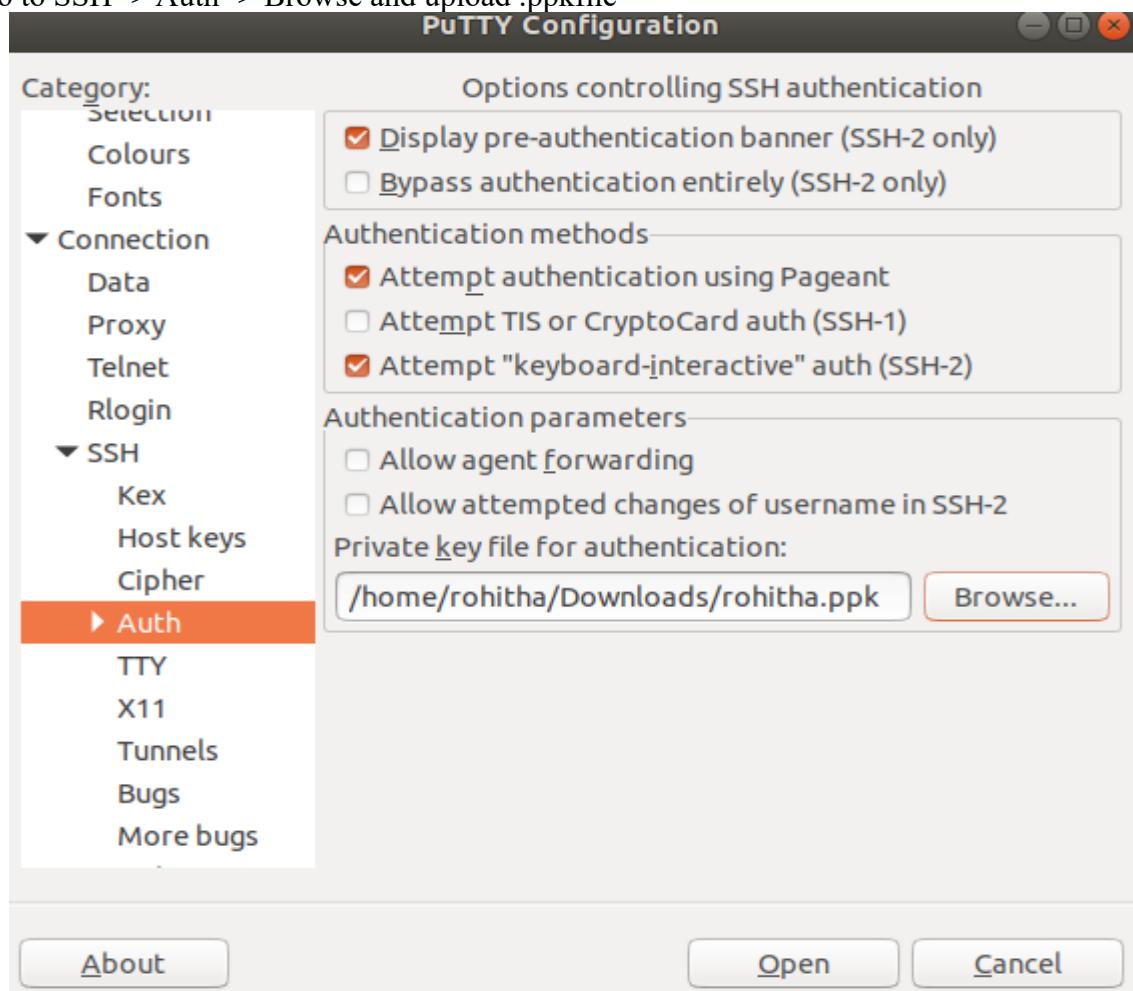
Go to putty and copy the IP address



convert .pem file to .ppk file

```
rohitha@rohitha-VirtualBox: ~/Downloads
File Edit View Search Terminal Help
rohitha@rohitha-VirtualBox:~$ cd Downloads
rohitha@rohitha-VirtualBox:~/Downloads$ puttygen rohitha.pem -o private -o rohit
ha.ppk
rohitha@rohitha-VirtualBox:~/Downloads$
```

Go to SSH -> Auth -> Browse and upload .ppkfile



click open -> accept
login: ec2-user

The screenshot shows a terminal window with the following text:

```
ec2-user@ip-172-31-17-138:~  
login as: ec2-user  
Authenticating with public key "imported-openssh-key"  
[ec2-user@ip-172-31-17-138 ~]$
```

The terminal window has a dark background and light-colored text. It includes standard window controls at the top right.

INSTALLING PHP:

```
ec2-user@ip-172-31-17-138:~ Is this ok [y/d/N]: sudo yum install php  
Is this ok [y/d/N]: y  
Downloading Packages:  
(1/9): apr-1.6.3-5.amzn2.0.2.x86_64.rpm | 118 kB 00:00  
(2/9): apr-util-1.6.1-5.amzn2.0.2.x86_64.rpm | 99 kB 00:00  
(3/9): apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64.rpm | 19 kB 00:00  
(4/9): generic-logos-httpd-18.0.0-4.amzn2.noarch.rpm | 19 kB 00:00  
(5/9): httpd-filesystem-2.4.41-1.amzn2.0.1.noarch.rpm | 23 kB 00:00  
(6/9): httpd-tools-2.4.41-1.amzn2.0.1.x86_64.rpm | 87 kB 00:00  
(7/9): mailcap-2.1.41-2.amzn2.noarch.rpm | 31 kB 00:00  
(8/9): mod_http2-1.15.3-2.amzn2.x86_64.rpm | 146 kB 00:00  
(9/9): httpd-2.4.41-1.amzn2.0.1.x86_64.rpm | 1.3 MB 00:00  
-----  
Total 5.0 MB/s | 1.8 MB 00:00  
Running transaction check  
Running transaction test  
Transaction test succeeded  
Running transaction  
  Installing : apr-1.6.3-5.amzn2.0.2.x86_64 1/9  
  Installing : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64 2/9  
  Installing : apr-util-1.6.1-5.amzn2.0.2.x86_64 3/9  
  Installing : httpd-tools-2.4.41-1.amzn2.0.1.x86_64 4/9  
  Installing : generic-logos-httpd-18.0.0-4.amzn2.noarch 5/9  
  Installing : mailcap-2.1.41-2.amzn2.noarch 6/9  
  Installing : httpd-filesystem-2.4.41-1.amzn2.0.1.noarch 7/9  
  Installing : mod_http2-1.15.3-2.amzn2.x86_64 8/9  
  Installing : httpd-2.4.41-1.amzn2.0.1.x86_64 9/9  
  Verifying : apr-util-1.6.1-5.amzn2.0.2.x86_64 1/9  
  Verifying : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64 2/9  
  Verifying : httpd-2.4.41-1.amzn2.0.1.x86_64 3/9  
  Verifying : httpd-filesystem-2.4.41-1.amzn2.0.1.noarch 4/9  
  Verifying : mod_http2-1.15.3-2.amzn2.x86_64 5/9  
  Verifying : apr-1.6.3-5.amzn2.0.2.x86_64 6/9  
  Verifying : mailcap-2.1.41-2.amzn2.noarch 7/9  
  Verifying : generic-logos-httpd-18.0.0-4.amzn2.noarch 8/9  
  Verifying : httpd-tools-2.4.41-1.amzn2.0.1.x86_64 9/9  
  
Installed:  
httpd.x86_64 0:2.4.41-1.amzn2.0.1  
  
Dependency Installed:  
apr.x86_64 0:1.6.3-5.amzn2.0.2  
apr-util.x86_64 0:1.6.1-5.amzn2.0.2  
apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2  
generic-logos-httpd.noarch 0:18.0.0-4.amzn2  
httpd-filesystem.noarch 0:2.4.41-1.amzn2.0.1  
httpd-tools.x86_64 0:2.4.41-1.amzn2.0.1  
mailcap.noarch 0:2.1.41-2.amzn2  
mod_http2.x86_64 0:1.15.3-2.amzn2  
  
Complete!
```

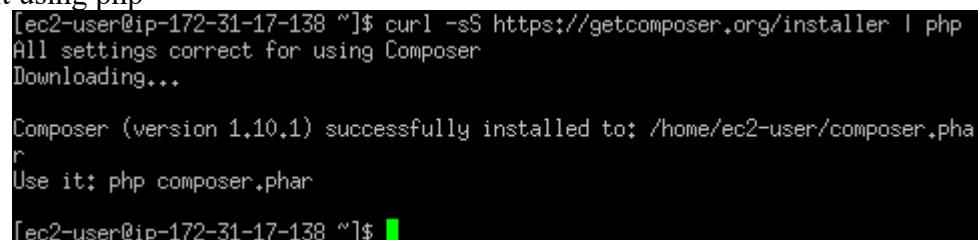
we can clear the screen using clear command (optional)



A screenshot of a terminal window titled "ec2-user@ip-172-31-17-138:~". The window is mostly black, indicating the "clear" command has been run. At the top, there are three small icons: a minus sign, a square, and a red circle with a white cross.

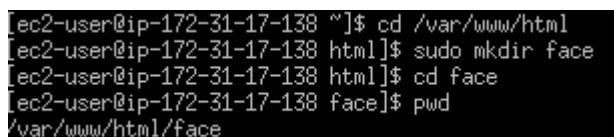
```
[ec2-user@ip-172-31-17-138 ~]$
```

curl -sS https://getcomposer.org/installer | php this command gets the composer and it installs it using php



```
[ec2-user@ip-172-31-17-138 ~]$ curl -sS https://getcomposer.org/installer | php
All settings correct for using Composer
Downloading...
Composer (version 1.10.1) successfully installed to: /home/ec2-user/composer.phar
Use it: php composer.phar
[ec2-user@ip-172-31-17-138 ~]$
```

cd /var/www /html



```
[ec2-user@ip-172-31-17-138 ~]$ cd /var/www/html
[ec2-user@ip-172-31-17-138 html]$ sudo mkdir face
[ec2-user@ip-172-31-17-138 html]$ cd face
[ec2-user@ip-172-31-17-138 face]$ pwd
/var/www/html/face
```

To download php sdk:

- 1)sudo swapon --show
- 2)df -h
- 3)sudo swapoff -a
- 4)sudo /bin/dd if=/dev/zero of=/swapfile bs=1M count=1024
- 5)sudo /sbin/mkswap /swapfile
- 6)sudo /sbin/swapon /swapfile
- 7)sudo php -d memory_limit=-1 ~/composer.phar require aws/aws-sdk-php

INSTALLING AWS-SDK-PHP:

```
[ec2-user@ip-172-31-17-138 face]$ sudo swapon --show
[ec2-user@ip-172-31-17-138 face]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        475M    0  475M   0% /dev
tmpfs          492M    0  492M   0% /dev/shm
tmpfs          492M  468K  492M   1% /run
tmpfs          492M    0  492M   0% /sys/fs/cgroup
/dev/xvda1     8.0G  1.4G  6.7G  17% /
tmpfs          99M    0   99M   0% /run/user/1000
[ec2-user@ip-172-31-17-138 face]$ sudo swapoff -a
[ec2-user@ip-172-31-17-138 face]$ sudo /bin/dd if=/dev/zero of=/var/swap.1 bs=1M
count=1024
/bin/dd: failed to open '/var/swap.1': No such file or directory
[ec2-user@ip-172-31-17-138 face]$ sudo /bin/dd if=/dev/zero of=/swapfile bs=1M c
ount=1024
1024+0 records in
1024+0 records out
1073741824 bytes (1.1 GB) copied, 13.438 s, 79.9 MB/s
[ec2-user@ip-172-31-17-138 face]$ sudo /sbin/mkswap /swapfile
mkswap: /swapfile: insecure permissions 0644, 0600 suggested.
Setting up swapspace version 1, size = 1024 MiB (1073737728 bytes)
no label, UUID=cec33691-bb4b-4745-9080-b962eb79a378
[ec2-user@ip-172-31-17-138 face]$ sudo /sbin/swaponp /swapfile
sudo: /sbin/swaponp: command not found
[ec2-user@ip-172-31-17-138 face]$ sudo /sbin/swapon /swapfile
swapon: /swapfile: insecure permissions 0644, 0600 suggested.
[ec2-user@ip-172-31-17-138 face]$ sudo php -d memory_limit=-1 ~/composer.phar re
quire aws/aws-sdk-php
Using version ^2.8 for aws/aws-sdk-php
./composer.json has been created
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 3 installs, 0 updates, 0 removals
- Installing symfony/event-dispatcher (v2.8.52): Loading from cache
- Installing guzzle/guzzle (v3.9.3): Downloading (100%)
- Installing aws/aws-sdk-php (2.8.31): Downloading (100%)
symfony/event-dispatcher suggests installing symfony/dependency-injection
symfony/event-dispatcher suggests installing symfony/http-kernel
guzzle/guzzle suggests installing guzzlehttp/guzzle (Guzzle 5 has moved to a new
package name. The package you have installed, Guzzle 3, is deprecated.)
aws/aws-sdk-php suggests installing doctrine/cache (Adds support for caching of
credentials and responses)
aws/aws-sdk-php suggests installing ext-apc (Allows service description opcode c
aching, request and response caching, and credentials caching)
aws/aws-sdk-php suggests installing monolog/monolog (Adds support for logging HT
TP requests and responses)
aws/aws-sdk-php suggests installing symfony/yaml (Eases the ability to write man
ifests for creating jobs in AWS Import/Export)
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/gu
zze instead.
Writing lock file
Generating autoload files
[ec2-user@ip-172-31-17-138 face]$
```

To get image from browser and save use:
get image from browser:



copy the Url and use it as below:

```
[ec2-user@ip-172-31-17-138 face]$ sudo wget https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
--2020-03-28 09:31:55-- https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
Resolving i.pinimg.com (i.pinimg.com)... 104.18.15.176, 104.18.14.176, 2600:1408
:20:ab2::1931, ...
Connecting to i.pinimg.com (i.pinimg.com)|104.18.15.176|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 215551 (210K) [image/jpeg]
Saving to: 'b97ea33b5842c7894b804923c6c05580.jpg'

100%[=====] 215,551 --.-K/s   in 0.03s
2020-03-28 09:31:55 (6.18 MB/s) - 'b97ea33b5842c7894b804923c6c05580.jpg' saved [215551/215551]

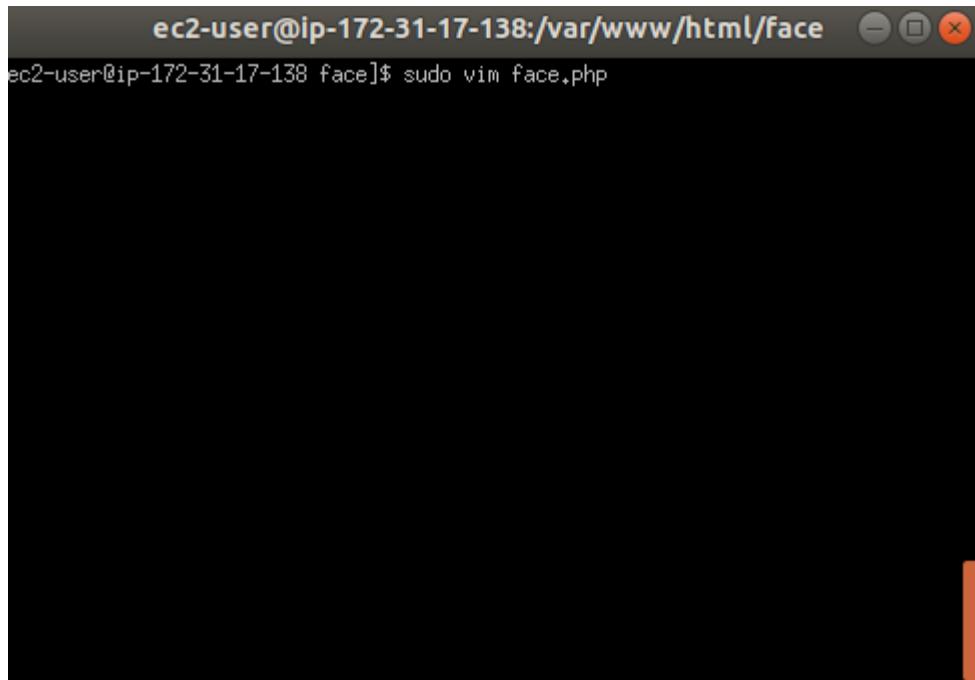
[ec2-user@ip-172-31-17-138 face]$
```

give command ls

```
[ec2-user@ip-172-31-17-138 face]$ ls
b97ea33b5842c7894b804923c6c05580.jpg  vendor
[ec2-user@ip-172-31-17-138 face]$
```

mv command is used to rename the file

```
[ec2-user@ip-172-31-17-138 face]$ sudo mv b97ea33b5842c7894b804923c6c05580.jpg rohitha.jpg
[ec2-user@ip-172-31-17-138 face]$ ls
aws.php  c.jpg  composer.json  composer.lock  rohitha.jpg  vendor
[ec2-user@ip-172-31-17-138 face]$
```



ec2-user@ip-172-31-17-138:/var/www/html/face

```
ec2-user@ip-172-31-17-138 face]$ sudo vim face.php
```

Click i to insert text

Type the code

PHP FILE CODE:

```

<?php
error_reporting(0);

require_once(__DIR__ . '/vendor/autoload.php');

use Aws\S3\S3Client;
use Aws\Rekognition\RekognitionClient;

$bucket = 'facedetection15';
$keyname = 'rohitha.jpg';

$s3 = S3Client::factory([
    'profile'      => 'default',
    'region'       => 'us-east-2',
    'version'      => '2006-03-01',
    'signature'    => 'v4'
]);
try {
    $result = $s3->putObject([
        'Bucket'           => $bucket,
        'Key'              => $keyname,
        'SourceFile'       => __DIR__ . "/$keyname",
        'ACL'              => 'public-read'
    ]);
    $imageUrl = $result['ObjectURL'];
    if($imageUrl) {
        echo "Image upload done... Here is the URL: " . $imageUrl;
    }
} catch (Exception $e) {
    echo $e->getMessage() . PHP_EOL;
}
^
^
^

```

29,2-9

All

Once done, press escape
:wq and press enter

To create role
go to services-> search for IAM
go to roles present in dashboard

Welcome to Identity and Access Management

IAM users sign-in link: <https://481453507501.signin.aws.amazon.com/console>

IAM Resources

Users: 0 Roles: 3

Groups: 0 Identity Providers: 0

Customer Managed Policies: 0

Security Status

1 out of 5 complete.

- Delete your root access keys
- Activate MFA on your root account
- Create individual IAM users
- Use groups to assign permissions
- Apply an IAM password policy

Feature Spotlight

Introduction to AWS IAM

Additional Information

IAM best practices

IAM documentation

Web Identity Federation Playground

Policy Simulator

Videos, IAM release history and additional resources

choose create role

Identity and Access Management (IAM)

Dashboard

Access management

Groups

Users

Roles

Policies

Identity providers

Account settings

Access reports

Access analyzer

Archive rules

Analyzer details

Credential report

Organization activity

Service control policies (SCPs)

Create role

Delete role

Showing 3 results

Role name	Trusted entities	Last activity
AWSServiceRoleForSupport	AWS service: support (Service-Linked role)	None
AWSServiceRoleForTrustedAdvisor	AWS service: trustedadvisor (Service-Linked ...)	None

go to Ec2



Or select a service to view its use cases

API Gateway	CodeDeploy	EMR	KMS	RoboMaker
AWS Backup	CodeGuru	ElastiCache	Kinesis	S3
AWS Chatbot	CodeStar Notifications	Elastic Beanstalk	Lambda	SMS
AWS Support	Comprehend	Elastic Container Service	Lex	SNS
Amplify	Config	Elastic Transcoder	License Manager	SWF
AppStream 2.0	Connect	Elastic Load Balancing	Machine Learning	SageMaker
AppSync	DMS	Forecast	Macie	Security Hub
Application Auto Scaling	Data Lifecycle Manager	Global Accelerator	MediaConvert	Service Catalog
Application Discovery Service	Data Pipeline	Glue	Migration Hub	Step Functions
Batch	DataSync	Greengrass	OpsWorks	Storage Gateway
Chime	DeepLens	GuardDuty	Personalize	Textract
CloudFormation	Directory Service	Health Organizational View	QLDB	Transfer
CloudHSM	DynamoDB	IAM Access Analyzer	RAM	Trusted Advisor
CloudTrail	EC2	Inspector	RDS	VPC
CloudWatch Application Insights	EC2 - Fleet	IoT	Redshift	WorkLink
CloudWatch Events	EC2 Auto Scaling	IoT Things Graph	Rekognition	WorkMail
CodeBuild	EKS			

Select use case as allows EC2 instances to call AWS services on your behalf

Select your use case

EC2

Allows EC2 instances to call AWS services on your behalf.

EC2 - Scheduled Instances

Allows EC2 Scheduled Instances to manage instances on your behalf.

EC2 - Spot Fleet

Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.

EC2 - Spot Fleet Auto Scaling

Allows Auto Scaling to access and update EC2 spot fleets on your behalf.

EC2 - Spot Fleet Tagging

Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.

EC2 - Spot Instances

Allows EC2 Spot Instances to launch and manage spot instances on your behalf.

EC2 Role for AWS Systems Manager

Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.

EC2 Spot Fleet Role

Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.

* Required

Cancel

Next: Permissions

click next permissions

Give S3 in filter policies and enable AmazonS3 full access



Create role

▼ Attach permissions policies

Choose one or more policies to attach to your new role.

[Create policy](#)



Filter policies ▾



S3

Showing 4 results



▶ AmazonDMSRedshiftS3Role

None



▶ AmazonS3FullAccess

Permissions policy (1)



▶ AmazonS3ReadOnlyAccess

None



▶ QuickSightAccessForS3StorageManagementAnalyticsReadOnly

None

▶ Set permissions boundary

* Required

[Cancel](#)

[Previous](#)

[Next: Tags](#)

click next tags

name the role



Create role

Review

Provide the required information below and review this role before you create it.

Role name*

recognition

Use alphanumeric and '+=_@-' characters. Maximum 64 characters.

Role description

Allows EC2 instances to call AWS services on your behalf.

Maximum 1000 characters. Use alphanumeric and '+=_@-' characters.

Trusted entities AWS service: ec2.amazonaws.com

Policies [AmazonS3FullAccess](#)

Permissions boundary Permissions boundary is not set

No tags were added.

* Required

[Cancel](#)

[Previous](#)

[Create role](#)

click create role

The role **recognition** has been created.

[Create role](#) [Delete role](#) [Edit](#) [Delete](#) [?](#)

Role name	Trusted entities	Last activity
AWSServiceRoleForSupport	AWS service: support (Service-Linked role)	None
AWSServiceRoleForTrustedAdvisor	AWS service: trustedadvisor (Service-Linked ...)	None
recognition	AWS service: ec2	None
testingrole	AWS service: ec2	None

Now go to Ec2 instances,
choose the particular instance -> action -> instance settings -> Attach/replace IAM role

Instances | EC2 Manager X rohitha.jpg (JPEG Image, 160x120) X +

https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#instances:sort=instanceId

Services Resource Groups

New EC2 Experience Tell us what you think

EC2 Dashboard New

Events New

Tags

Reports

Limits

INSTANCES Instances

- Instance Types
- Launch Templates New
- Spot Requests
- Savings Plans
- Reserved Instances
- Dedicated Hosts New
- Capacity Reservations

IMAGES AMIs

Bundle Tasks

ELASTIC BLOCK STORE Volumes Snapshots

Actions ▾

Connect Get Windows Password

Create Template From Instance

Launch More Like This

Instance State

Instance Settings

Image

Networking

CloudWatch Monitoring

1 to 2 of 2

Name	Instance ID	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IP
i-0224356be073ccc91	i-0224356be073ccc91	us-east-2b	running	2/2 checks ...	None	ec2-3-21-185-4.us-east-2...	3.21.185.4
i-0ad69d15508c	i-0ad69d15508c	us-east-2b	terminated	0 checks ...	None	ec2-18-217-97-62.us-e...	18.217.97.62

Instance: i-0224356be073ccc91 Public DNS: ec2-3-21-185-4.us-east-2.compute.amazonaws.com

Description Status Checks Monitoring Tags

Instance ID: i-0224356be073ccc91 Public DNS (IPv4): ec2-3-21-185-4.us-east-2.compute.amazonaws.com

Instance state: running IPv4 Public IP: 3.21.185.4

Feedback English (US) © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Rohitha Ohio Support

select the role you created

Instances > Attach/Replace IAM Role

Attach/Replace IAM Role

Select an IAM role to attach to your instance. If you don't have any IAM roles, choose Create new IAM role to create a role in the IAM console. If an IAM role is already attached to your instance, the IAM role you choose will replace the existing role.

Instance ID: i-0224356be073ccc91 ⓘ

IAM role* recognition

Create new IAM role ⓘ

* Required

Cancel Apply

click apply

now run the command sudo php face.php, it successfully executes.

UPLOAD SUCCESS SCREENSHOT:

```
[ec2-user@ip-172-31-17-138 face]$ sudo vim face.php
[ec2-user@ip-172-31-17-138 face]$ sudo php face.php
Error retrieving credentials from the instance profile metadata server. When you
are not running inside of Amazon EC2, you must provide your AWS access key ID and
secret access key in the "key" and "secret" options when creating a client or
provide an instantiated Aws\Common\Credentials\CredentialsInterface object. (Client
error response)
[status code] 404
[reason phrase] Not Found
[url] http://169.254.169.254/latest/meta-data/iam/security-credentials/
[ec2-user@ip-172-31-17-138 face]$ sudo php rohitha.php
Could not open input file: rohitha.php
[ec2-user@ip-172-31-17-138 face]$ sudo php face.php
Image upload done... Here is the URL: https://facedetection15.s3.us-east-2.amazonaws.com/rohitha.jpg[ec2-user@ip-172-31-17-138 face]$
```

So

now my .jpg file got uploaded in facedetection15 bucket

The screenshot shows the AWS S3 console interface. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, 'Resource Groups' dropdown, and user information ('Rohitha', 'Global', 'Support'). Below the navigation is a breadcrumb trail: 'Amazon S3 > facedetection15'. The main area shows a folder named 'facedetection15'. Below the folder name are five tabs: 'Overview' (selected), 'Properties', 'Permissions', 'Management', and 'Access points'. A search bar with placeholder text 'Type a prefix and press Enter to search. Press ESC to clear.' is present. Below the search bar are four buttons: 'Upload', '+ Create folder', 'Download', and 'Actions'. To the right of these buttons is the region 'US East (Ohio)' with a refresh icon. A table lists the contents of the bucket:

Name	Last modified	Size	Storage class
rohitha.html	Mar 27, 2020 2:39:38 PM GMT+0530	312.0 B	Standard
rohitha.jpg	Mar 28, 2020 7:52:59 PM GMT+0530	210.5 KB	Standard

At the bottom of the table, it says 'Viewing 1 to 2'.

click the above jpg file

AWS Services Resource Groups

Amazon S3 > facedetection15 > rohitha.jpg

rohitha.jpg Latest version

Overview Properties Permissions Select from

Open Download Download as Make public Copy path

Owner
c4686f7b3e0a9cd4c5bd2d936676bebb04b4a4594d10e5ee4d2bc619d6ba5505

Last modified
Mar 28, 2020 7:52:59 PM GMT+0530

Etag
5715388228d652105c7ac4331a990052

Storage class
Standard

Server-side encryption
None

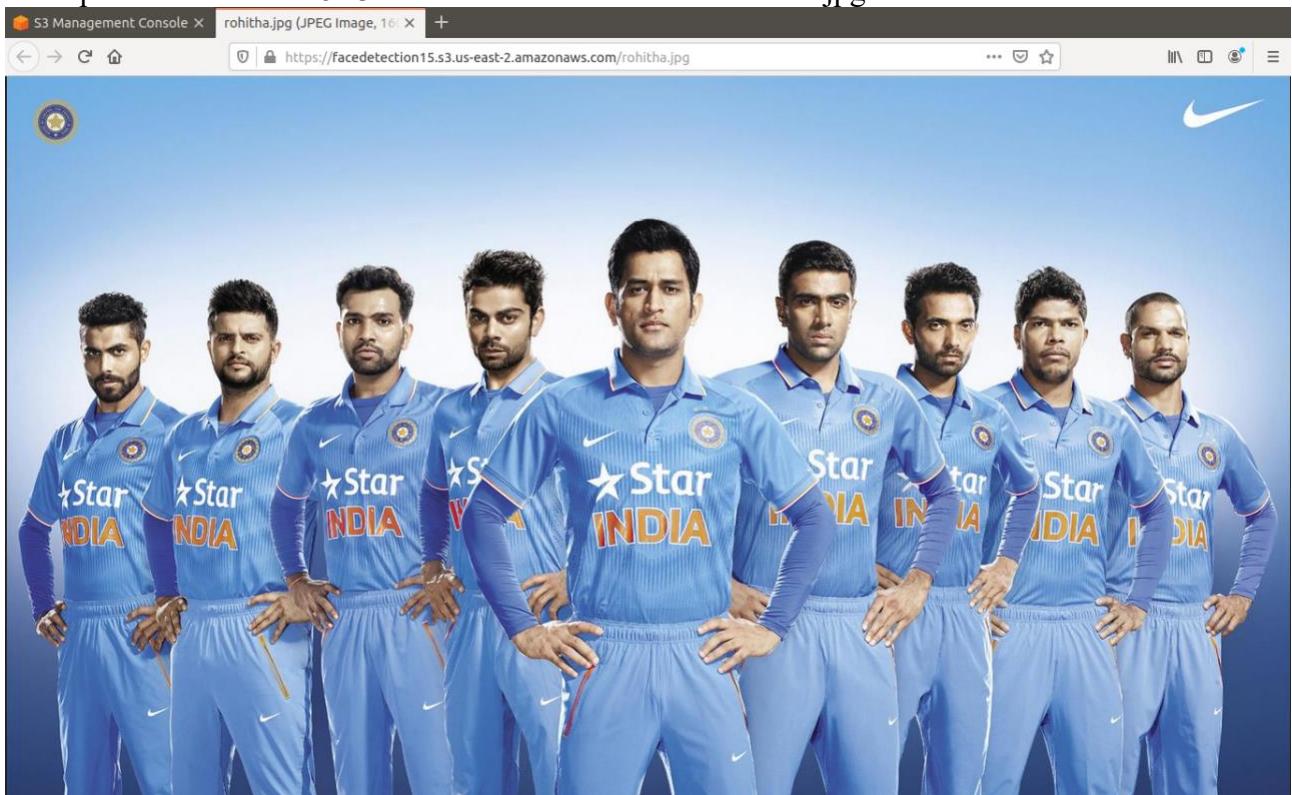
Size
210.5 KB

Key
rohitha.jpg

Object URL
<https://facedetection15.s3.us-east-2.amazonaws.com/rohitha.jpg>

click the URL over there

<https://facedetection15.s3.us-east-2.amazonaws.com/rohitha.jpg>



**AMAZON REKOGNITION:
OBJECT AND SCENE DETECTION:**

AWS Services Resource Groups

Amazon Rekognition

- Custom Labels New
- Use Custom Labels
- Demos
 - Object and scene detection**
 - Image moderation
 - Facial analysis
 - Celebrity recognition
 - Face comparison
 - Text in image
- Video Demos
- Video analysis
- Metrics
- Metrics
- Additional Resources
- Getting started guide
- Download SDKs

Object and scene detection

Rekognition automatically labels objects, concepts and scenes in your images, and provides a confidence score.

Done with the demo? [Learn more](#)

Results	
Transportation	98.8 %
Vehicle	98.8 %
Car	98.8 %
Automobile	98.8 %
Person	98.3 %
Human	98.3 %

Show more

Request

Response

AWS Services Resource Groups

Amazon Rekognition

- Custom Labels New
- Use Custom Labels
- Demos
 - Facial analysis**
 - Object and scene detection
 - Image moderation
- Celebrity recognition
- Face comparison
- Text in image
- Video Demos
- Video analysis
- Metrics
- Metrics
- Additional Resources
- Getting started guide
- Download SDKs

Facial analysis

Get a complete analysis of facial attributes, including confidence scores.

Done with the demo? [Learn more](#)

Results	
looks like a face	99.9 %
appears to be female	99.9 %
age range	17 - 29 years old
smiling	91.7 %
appears to be happy	99.5 %
wearing glasses	99.8 %

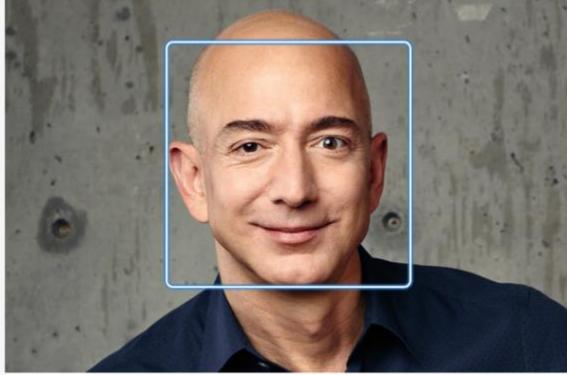
Show more

Request

CELEBRITY RECOGNITION:

Celebrity recognition

Rekognition automatically recognizes celebrities in images and provides confidence scores.



Done with the demo? [Learn more](#)

Results

 **Jeff Bezos** [Learn More](#)

Match confidence 100 %

Request

Response

Choose a sample image



Use your own image
Image must be .jpeg or .png format and no larger than 5MB. Your image isn't stored.

Upload or drag and drop

Use image URL **Go**

Additional Resources

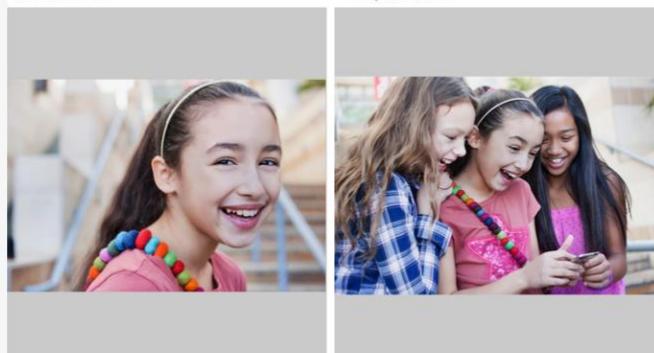
Getting started guide

Download SDKs

FACE COMPARISON:

Face comparison

Compare faces to see how closely they match based on a similarity percentage.



Done with the demo? [Learn more](#)

Results

 = 
Similarity 99.8 %

 ≠ 

 ≠ 

Reference face

Comparison faces

Choose a sample image



Choose a sample image



Additional Resources

Getting started guide

Download SDKs

TEXT IN IMAGE:

The screenshot shows the AWS Rekognition 'Text in image' demo. On the left sidebar, under 'Text in image', there is a link to 'Video Demos'. The main content area displays a red mug with a smiley face and text overlays: 'IT'S MONDAY but keep Smiling'. Below the image are two sections: 'Choose a sample image' (with a thumbnail of the mug) and 'Use your own image' (with a 'Upload' button and a 'Use image URL' input field). To the right, a results panel shows the detected text: 'IT'S | MONDAY | but | keep | Smiling |'. There are also sections for 'Request' and 'Response'.

EC2 AND REKOGNITION:

Services -> Ec2 -> Running instances -> Copy the IP address and paste it in Putty
 Connection -> SSH -> auth -> browse -> upload .ppk file
 login as ec2-user

```
cd /var/www/html/face
sudo yum remove php*
sudo yum remove httpd*
sudo yum clean all
sudo yum upgrade -y
sudo amazon-linux-extras install php7.2
sudo yum install php-json php-xml php-cli php-mbstring
sudo yum install httpd
php -v
sudo swapon --show
df -h
sudo swapoff -a
sudo /bin/dd if=/dev/zero of=/swapfile bs=1M count=1024
sudo /sbin/mkswap /swapfile
sudo /sbin/swapon /swapfile
sudo php -d memory_limit=-1 ~/composer.phar require aws/aws-sdk-php
sudo service httpd start
```

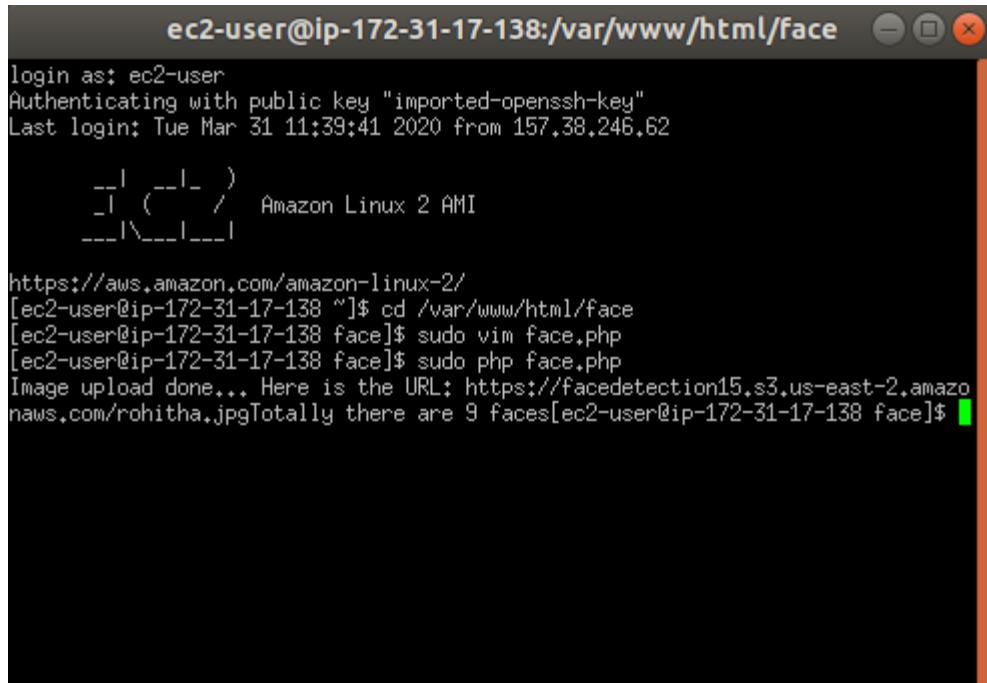
```
ec2-user@ip-172-31-17-138:/var/www/html/face
```

```
<?php  
  
require_once(__DIR__ . '/vendor/autoload.php');  
  
use Aws\S3\S3Client;  
use Aws\Rekognition\RekognitionClient;  
  
$bucket = 'facedetection15';  
$keyname = 'rohitha.jpg';  
  
$s3 = new S3Client([  
    'region'      => 'us-east-2',  
    'version'     => '2006-03-01',  
    'signature'   => 'v4'  
]);  
try {  
    $result = $s3->putObject([  
        'Bucket'          => $bucket,  
        'Key'             => $keyname,  
        'SourceFile'      => __DIR__ . "/$keyname",  
        'ACL'             => 'public-read-write'  
    ]);  
    $imageUrl = $result['ObjectURL'];  
    if($imageUrl) {  
        echo "Image upload done... Here is the URL: " . $imageUrl;  
  
        $rekognition = new RekognitionClient([  
            'region'      => 'us-east-2',  
            'version'     => 'latest',  
        ]);  
  
        $result = $rekognition->detectFaces([  
            'Attributes'  => ['DEFAULT'],  
            'Image'       => [  
                'S3Object' => [  
                    'Bucket'  => $bucket,  
                    'Name'    => $keyname,  
                    'Key'     => $keyname,  
                ],  
            ],  
        ]);  
        echo "Totally there are " . count($result["FaceDetails"]) . " faces";  
    }  
}  
catch (Exception $e) {  
    echo $e->getMessage() . PHP_EOL;  
}  
^
```

1,5

All

FINAL OUTPUT:



```
ec2-user@ip-172-31-17-138:/var/www/html/face
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Tue Mar 31 11:39:41 2020 from 157.38.246.62
[ec2-user@ip-172-31-17-138 ~]$ cd /var/www/html/face
[ec2-user@ip-172-31-17-138 face]$ sudo vim face.php
[ec2-user@ip-172-31-17-138 face]$ sudo php face.php
Image upload done... Here is the URL: https://facedetection15.s3.us-east-2.amazonaws.com/rohitha.jpgTotally there are 9 faces[ec2-user@ip-172-31-17-138 face]$
```



URL:

3.21.185.4/face/face.php