

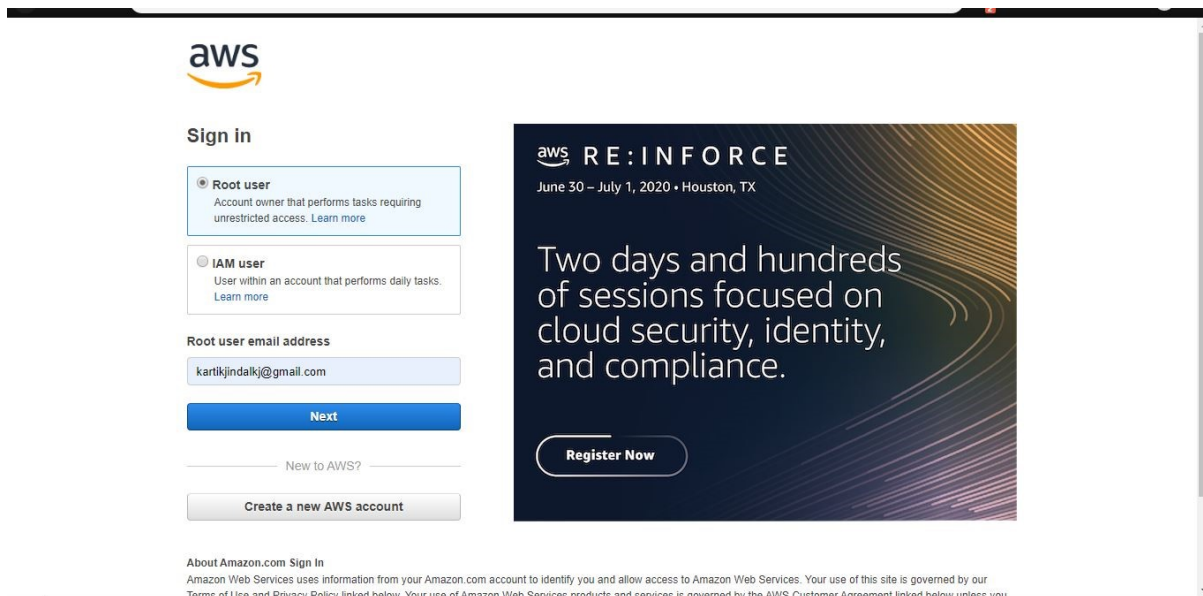
# FACE DETECTION APP

Kartik Jindal

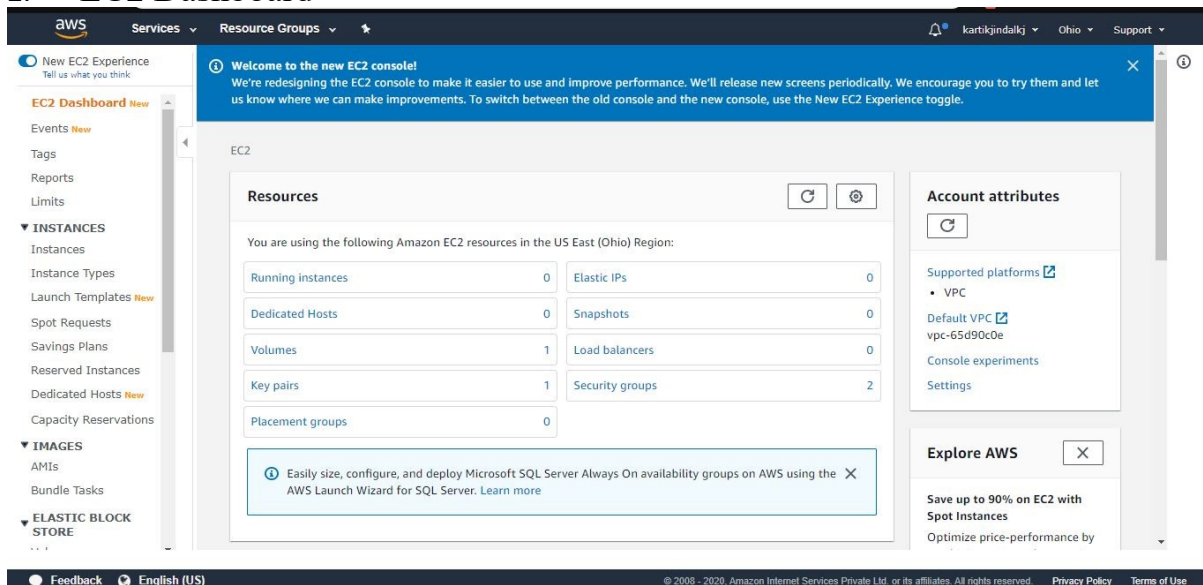
Username: kartikjindalkj

## SCREENSHOTS -DASHBOARDS

### 1. AWS Login screen with username



### 2. EC2 Dashboard



### 3. S3 Dashboard

The screenshot shows the Amazon S3 console interface. At the top, there's a navigation bar with the AWS logo, 'Services', 'Resource Groups', and user information. A blue banner at the top right contains a message about the console update. The left sidebar lists 'Amazon S3' with sub-links for 'Buckets', 'Batch operations', 'Access analyzer for S3', 'Block public access (account settings)', and 'Feature spotlight'. The main content area is titled 'Amazon S3' and shows a 'Buckets (1)' section. It includes a search bar 'Find bucket by name', a table with one bucket named 'aws-bot-trial' in 'US East (Ohio) us-east-2' region, and buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'. The footer contains 'Feedback', 'English (US)', and copyright information.

Name	Region	Access	Bucket created
aws-bot-trial	US East (Ohio) us-east-2	Objects can be public	2020-04-04T09:19:33.000Z

### 4. Rekognition Dashboard

The screenshot shows the Amazon Rekognition dashboard. The top navigation bar is similar to the S3 console. The left sidebar lists 'Amazon Rekognition' with sub-links for 'Demos' (Object and scene detection, Image moderation, Facial analysis, Celebrity recognition, Face comparison, Text in image), 'Video Demos' (Video analysis), 'Metrics', and 'Additional Resources' (Getting started guide, Download SDKs). The main content area features a large hero section with the title 'Amazon Rekognition' and the description 'Deep learning-based visual analysis service. Search, verify, and organize millions of images and videos.' Below this are three columns: ' Easily Integrate Powerful Visual Analysis into Your App', 'Continuously Learning' (describing the service's use of deep learning technology), and 'Integrated with AWS Services' (describing integration with other AWS services). The footer includes 'Feedback', 'English (US)', and copyright information.

# SCREENSHOTS-EC2

## 1. Choosing an AMI

aws

Services

Resource Groups

🔔

kartikjindalkj

Ohio

Support

1. Choose AMI

2. Choose Instance Type

3. Configure Instance

4. Add Storage

5. Add Tags

6. Configure Security Group

7. Review

Cancel and Exit

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.

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

×

Quick Start

1 to 40 of 40 AMIs

My AMIs

AWS Marketplace

Community AMIs

Free tier only

ⓘ

Amazon Linux

Free tier eligible

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.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)

64-bit (Arm)

Amazon Linux

Free tier eligible

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.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)

Red Hat

Free tier eligible

Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-0520e698dd500b1d1 (64-bit x86) / ami-0099847d600887c9f (64-bit Arm)

Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)

64-bit (Arm)

## 2. Choosing an Instance Type

aws

Services

Resource Groups

🔔

kartikjindalkj

Ohio

Support

1. Choose AMI

2. Choose Instance Type

3. Configure Instance

4. Add Storage

5. Add Tags

6. Configure Security Group

7. Review

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 ECU's, 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
<input type="checkbox"/>	General purpose	t2.2xlarge	8	32	EBS only	-	Moderate	Yes

Cancel

Previous

Review and Launch

Next: Configure Instance Details

## 3. Adding Storage

aws

Services

Resource Groups

🔔

kartikjindalkj

Ohio

Support

1. Choose AMI

2. Choose Instance Type

3. Configure Instance

4. Add Storage

5. Add Tags

6. Configure Security Group

7. Review

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	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.

Cancel

Previous

Review and Launch

Next: Add Tags

## 4. Configuring 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:

Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

[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](#)

## 5. Key Pair Download

Step 7: Review Instance Launch

Stop - Hibernate behavior Disabled open  
Capacity Reservation IAM role None  
Tenancy default  
T2/T3 Unlimited Disabled  
Host ID  
Host resource group name  
Affinity Off  
Kernel ID Use default  
RAM disk ID Use default  
Metadata accessible Enabled  
Metadata version V1 and V2 (token)  
Metadata token response hop limit 1  
User data  
Assign Public IP Use subnet setti  
Assign IPv6 IP Use subnet setti

Storage

Volume Type Device Snapshot  
Root /dev/xvda snap-0f54692056a

Tags

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

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

[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](#)

[Edit storage](#)

[Encrypted](#) [Not Encrypted](#)

[Edit tags](#)

[Cancel](#) [Previous](#) [Launch](#)

## 6. PuTTYgen conversion from pem to ppk

PuTTY Key Generator

File Key Conversions Help

Key

Public key for pasting into OpenSSH authorized\_keys file:

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCEtKSFVlaYqCQC  
+PxjsKOFaURb4ylioWf5Zz+2sAozIRLMJfoXISwgeimeC1JN6myAiD6ojrd8/n6/  
+xfzZzgQ1YH971kdyldNHWrG  
+7bCnJ93E2hCE56iW44Zn6v9BLoXtkLDXKl2PCulQeffNFE5mR1HWsmGDNIIE2F8kR  
dwol0EEkJPpGyG1YfXYuYrle9xQa7ROrTLtwiVejRFctgFgDj/egs621eyPXNQdzqQhmtm
```

Key fingerprint: ssh-rsa 2048 af:c4:0b:11:0e:9c:0b:d6:0d:12:b6:8f:01:1f:77:c1

Key comment: imported-openssh-key

Key passphrase:

Confirm passphrase:

Actions

Generate a public/private key pair [Generate](#)

Load an existing private key file [Load](#)

Save the generated key [Save public key](#) [Save private key](#)

Parameters

Type of key to generate:  
☒ RSA ☐ DSA ☐ ECDSA ☐ Ed25519 ☐ SSH-1 (RSA)

Number of bits in a generated key:



## 7. Logged in EC2 black screen

```
ec2-user@ip-172-31-21-200:~$
login as: ec2-user
Authenticating with public key "imported-openssh-key"

 _ | _ | _ |
 _ | ( _ | _ |
 _ | \ _ | _ |

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
1 package(s) needed for security, out of 7 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-21-200 ~]$ 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: httpd-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_64
--> 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.amzn2.0.1.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]: y
Downloading packages:
(1/9): apr-util-1.6.1-5.amzn2.0.2.x86_64.rpm | 99 kB 00:00
(2/9): apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64.rpm | 19 kB 00:00
(3/9): apr-1.6.3-5.amzn2.0.2.x86_64.rpm | 118 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): httpd-2.4.41-1.amzn2.0.1.x86_64.rpm | 1.3 MB 00:00
(8/9): mailcap-2.1.41-2.amzn2.noarch.rpm | 31 kB 00:00
(9/9): mod_http2-1.15.3-2.amzn2.x86_64.rpm | 146 kB 00:00
-----
Total 9.8 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
```

ec2-user@ip-172-31-21-200:~

```
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-172-31-21-200 ~]$ sudo service httpd start
```

```
Redirecting to /bin/systemctl start httpd.service
```

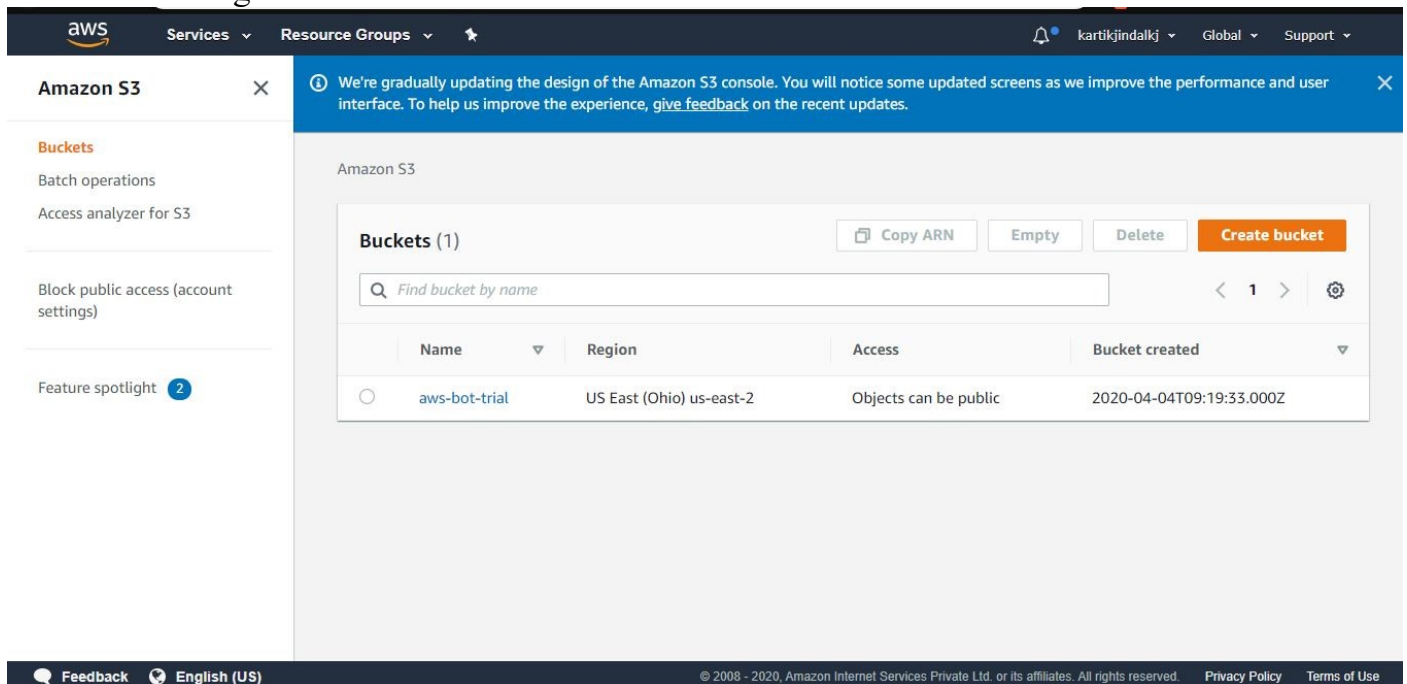
```
[ec2-user@ip-172-31-21-200 ~]$ sudo service httpd status
```

```
Redirecting to /bin/systemctl status httpd.service
```

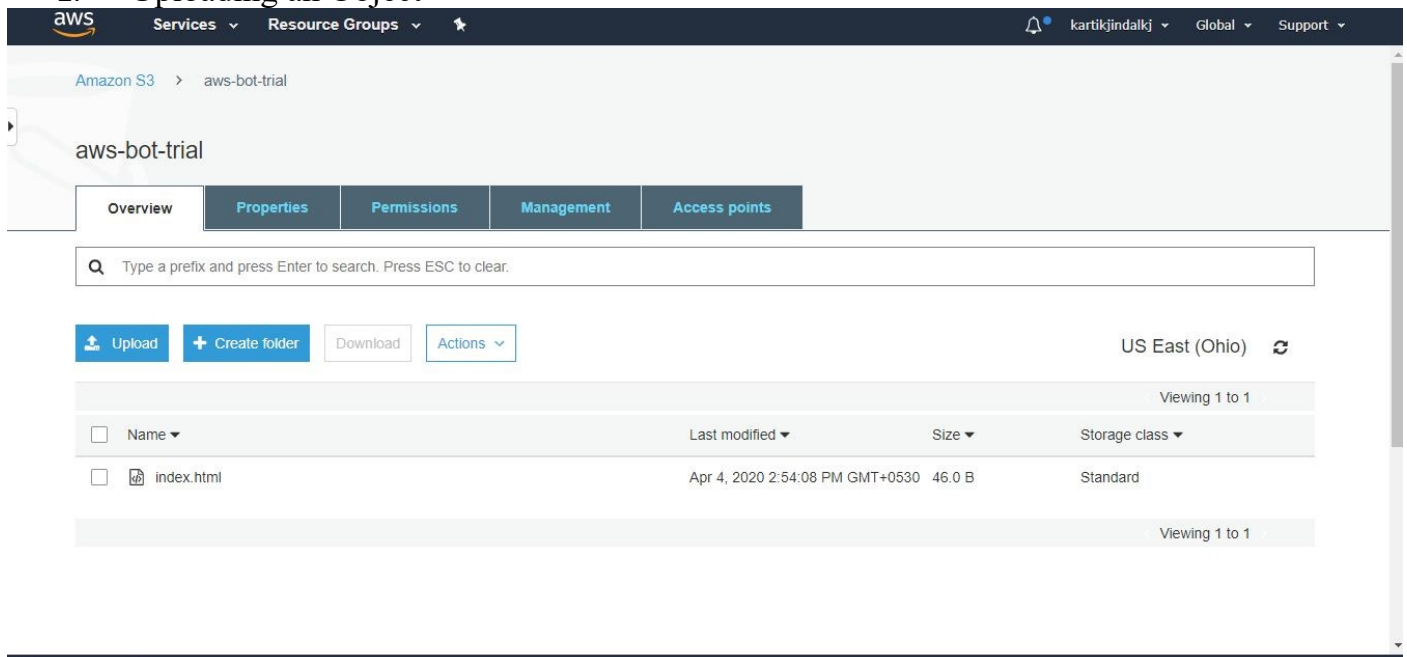
```
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor prese
t: disabled)
   Active: active (running) since Sat 2020-04-04 08:11:08 UTC; 14s ago
     Docs: man:httpd.service(8)
  Main PID: 3656 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes se
rved/sec:  0 B/sec"
    CGroup: /system.slice/httpd.service
            └─3656 /usr/sbin/httpd -DFOREGROUND
            └─3657 /usr/sbin/httpd -DFOREGROUND
            └─3658 /usr/sbin/httpd -DFOREGROUND
            └─3659 /usr/sbin/httpd -DFOREGROUND
            └─3660 /usr/sbin/httpd -DFOREGROUND
            └─3661 /usr/sbin/httpd -DFOREGROUND
```

# SCREENSHOTS NEEDED FOR S3

## 1. Creating a bucket



## 2. Uploading an Object



### 3. Enabling Static Website

The screenshot shows the 'Static website hosting' configuration page in the AWS console. The endpoint is `http://aws-bot-trial.s3-website.us-east-2.amazonaws.com`. The 'Use this bucket to host a website' option is selected. The index document is `index.html` and the error document is `error.html`. There are no redirection rules. The 'Bucket hosting' option is selected at the bottom. To the right, the 'Object-level logging' panel is shown as 'Disabled'.

### 4. Making the Object Public

The screenshot shows the 'Public access settings' page in the AWS console. A green message bar at the top states 'Public access settings updated successfully'. Below, the 'Block all public access' toggle is set to 'Off'. A list of settings is shown, all with 'Off' toggles:

- 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
- Block public and cross-account access to buckets and objects through *any* public bucket or access point policies

### 5. Checking the S3 link on the browser





# SCREENSHOTS NEEDED FOR REKOGNITION

## 1. Face Detect

The screenshot shows the AWS Rekognition 'Facial analysis' demo. The left sidebar contains navigation links for Amazon Rekognition, Custom Labels, Use Custom Labels, Demos, Object and scene detection, Image moderation, Facial analysis (highlighted), Celebrity recognition, Face comparison, Text in image, Video Demos, Video analysis, Metrics, and Additional Resources. The main content area is titled 'Facial analysis' with the subtitle 'Get a complete analysis of facial attributes, including confidence scores.' It features a large image of a group of people with bounding boxes around their faces. Below this, there are two options to 'Choose a sample image' or 'Use your own image' (with an 'Upload' button). On the right, a 'Results' section displays a list of attributes and their confidence scores:

Attribute	Confidence Score
looks like a face	99.9 %
appears to be male	97.8 %
age range	52 - 70 years old
smiling	99.1 %
appears to be happy	99.4 %
not wearing glasses	98.1 %
not wearing sunglasses	99.5 %

The footer includes 'Feedback', 'English (US)', and copyright information: '© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use'.

## 2. Face Compare

The screenshot shows the AWS Rekognition 'Face comparison' demo. The left sidebar is identical to the first screenshot, with 'Face comparison' highlighted. The main content area is titled 'Face comparison' with the subtitle 'Compare faces to see how closely they match based on a similarity percentage.' It features two image upload sections: 'Reference face' and 'Comparison faces'. Below these, there are two options to 'Choose a sample image' or 'Use your own image' (with an 'Upload' button). On the right, a 'Results' section displays a comparison of two faces with an equals sign between them and a 'Similarity' score of 98.2 % shown on a blue progress bar. Below the similarity score, there are links for 'Request' and 'Response'. The footer is identical to the first screenshot.

### 3. Celebrity Recognition

aws

Services

Resource Groups

Feedback

English (US)

Amazon Rekognition

Custom Labels

Use Custom Labels

Demos

Object and scene detection

Image moderation

Facial analysis

Celebrity recognition

Face comparison

Text in image

Video Demos

Video analysis

Metrics

Additional Resources

Getting started guide

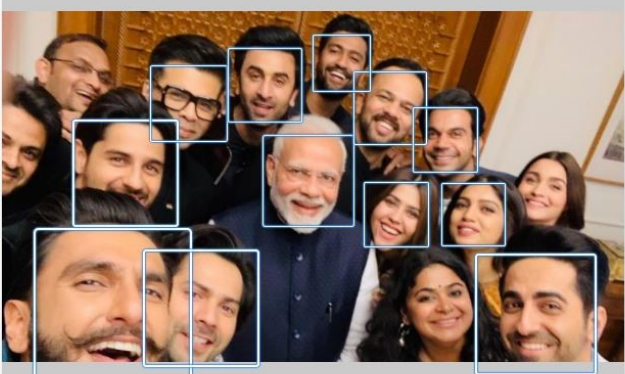
© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.

Privacy Policy

Terms of Use

### Celebrity recognition

Rekognition automatically recognizes celebrities in images and provides confidence scores.



Choose a sample image

Use your own image

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


Upload or drag and drop

Go

Done with the demo?

Learn more


▼ Results



Ranveer Singh

Learn More


Match confidence100 %



Ayushmann Khurrana

Learn More

Match confidence87 %



Varun Dhawan

Learn More

### 4. Text in Image

aws

Services

Resource Groups

Feedback

English (US)

Amazon Rekognition

Custom Labels

Use Custom Labels

Demos

Object and scene detection

Image moderation

Facial analysis

Celebrity recognition

Face comparison

Text in image

Video Demos

Video analysis

Metrics

Additional Resources

Getting started guide


© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.

Privacy Policy

Terms of Use

### Text in image

Rekognition automatically detects and extracts text in your images. [Learn More](#)



Choose a sample image

Use your own image

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

Upload or drag and drop

Go

Done with the demo?

Learn more

▼ Results

US English only

| C |

| J389 | NLT |

► Request

► Response

# SCREENSHOTS NEEDED FOR EC2 & S3

## 1. Installing aws-sdk

```
ec2-user@ip-172-31-21-200:~$ ssh -i /home/ec2-user/.ssh/face
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Sat Apr  4 13:17:06 2020 from 182.68.197.116

 _ _ | _ _ | _ _ |
 _ _ | ( _ _ | /   Amazon Linux 2 AMI
 _ _ | \ _ _ | _ _ |

https://aws.amazon.com/amazon-linux-2/
1 package(s) needed for security, out of 7 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-21-200 ~]$ Install php - sudo yum install php
-bash: Install: command not found
[ec2-user@ip-172-31-21-200 ~]$ sudo yum install php
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 2.4 kB 00:00
amzn2extra-docker | 1.8 kB 00:00
Package php-5.4.16-46.amzn2.0.x86_64 already installed and latest version
Nothing to do
[ec2-user@ip-172-31-21-200 ~]$ 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-172-31-21-200 ~]$ cd /var/www/html
[ec2-user@ip-172-31-21-200 html]$ cd face
[ec2-user@ip-172-31-21-200 face]$ sudo php -d memory_limit=-1 ~/composer.phar require aws/aws-sdk-php
Using version ^2.8 for aws/aws-sdk-php
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Nothing to install or update
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/guzzle instead.
Generating autoload files
[ec2-user@ip-172-31-21-200 face]$
```

## 2. Installing php



ec2-user@ip-172-31-21-200:/var/www/html/face

```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Sat Apr  4 13:17:06 2020 from 182.68.197.116

 _ _ | _ _ | _ )
 _ | ( _ _ /
 _ | \ _ _ | _ _ |

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
1 package(s) needed for security, out of 7 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-21-200 ~]$ Install php - sudo yum install php
-bash: Install: command not found
[ec2-user@ip-172-31-21-200 ~]$ sudo yum install php
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 2.4 kB 00:00
amzn2extra-docker | 1.8 kB 00:00
Package php-5.4.16-46.amzn2.0.2.x86_64 already installed and latest version
Nothing to do
[ec2-user@ip-172-31-21-200 ~]$ 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-172-31-21-200 ~]$ cd /var/www/html
[ec2-user@ip-172-31-21-200 html]$ cd face
[ec2-user@ip-172-31-21-200 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 updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Nothing to install or update
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/gu
zle instead.
Generating autoload files
[ec2-user@ip-172-31-21-200 face]$
```

### 3. index.php file code

```
ec2-user@ip-172-31-21-200:/var/www/html/face

sudo mv b97ead33b0e443c7894b00492304c05580.jpg sample.jpg

</
error_reporting(0);

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

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

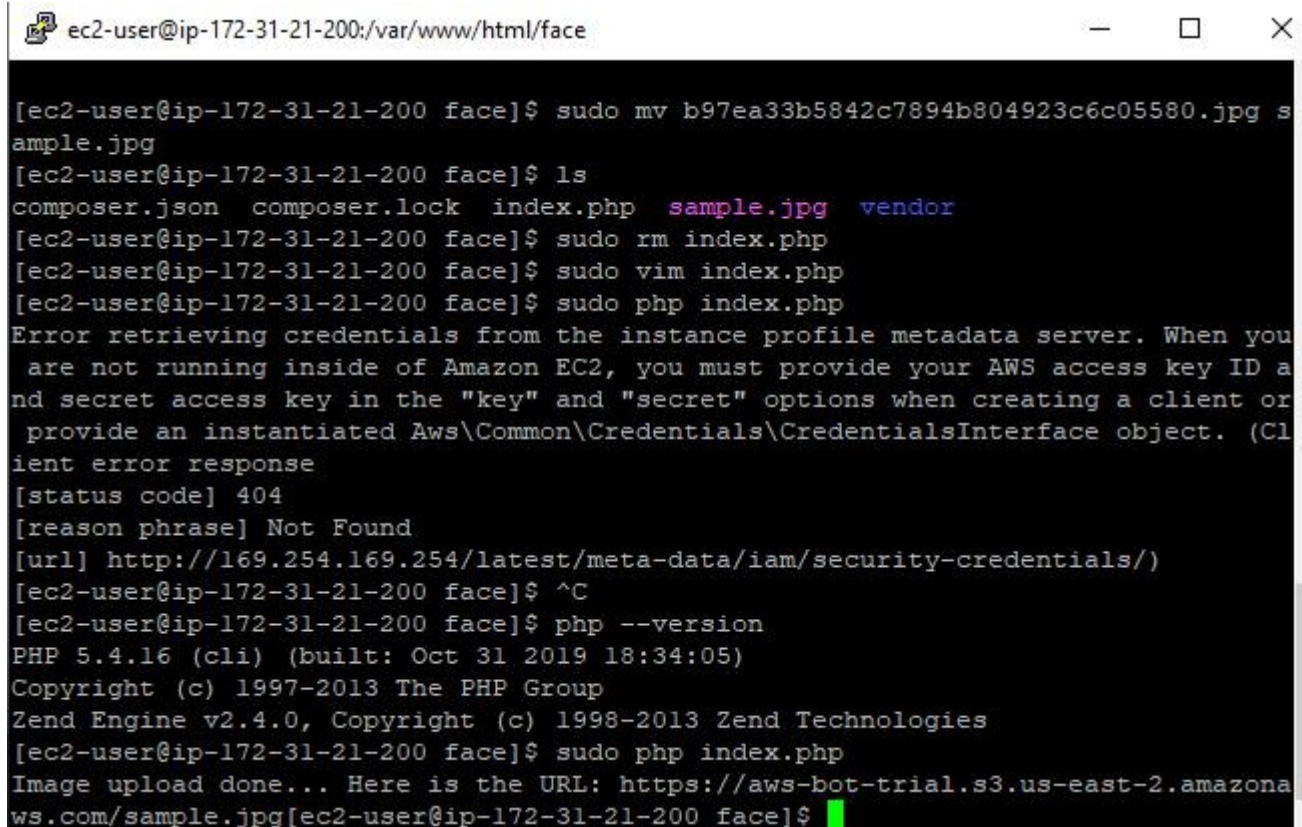
$bucket = 'aws-bot-trial';
$keyname = 'sample.jpg';

$s3 = S3Client::factory([
    'profile' => 'default',
    'region' => 'us-east-2',
    'version' => '2006-03-01',
    'signature' => 'v4'
]);

try {
    // Upload data.
-- INSERT --
35,1 53%
```



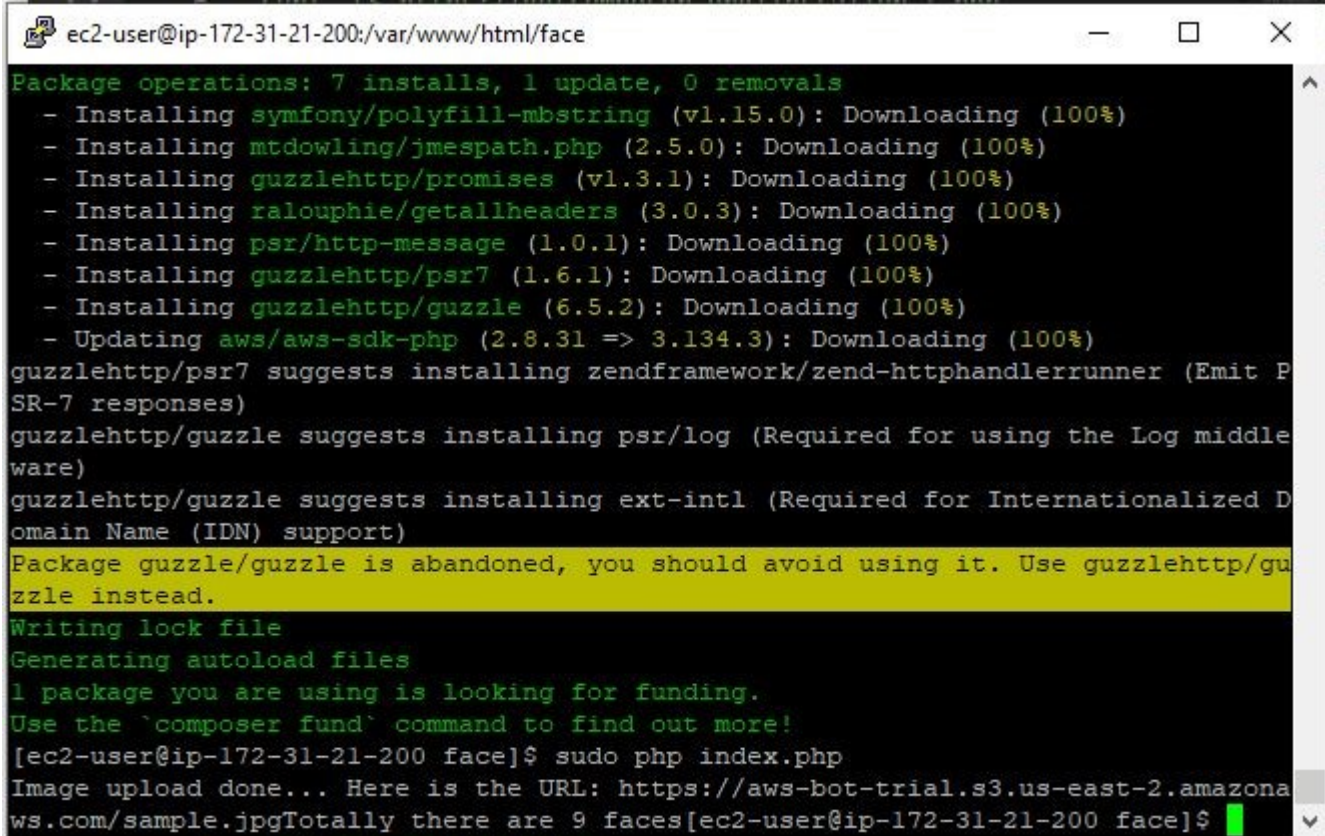
#### 4. Upload success screenshot



```
ec2-user@ip-172-31-21-200:/var/www/html/face
[ec2-user@ip-172-31-21-200 face]$ sudo mv b97ea33b5842c7894b804923c6c05580.jpg sample.jpg
[ec2-user@ip-172-31-21-200 face]$ ls
composer.json  composer.lock  index.php  sample.jpg  vendor
[ec2-user@ip-172-31-21-200 face]$ sudo rm index.php
[ec2-user@ip-172-31-21-200 face]$ sudo vim index.php
[ec2-user@ip-172-31-21-200 face]$ sudo php index.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 a
nd secret access key in the "key" and "secret" options when creating a client or
provide an instantiated Aws\Common\Credentials\CredentialsInterface object. (Cl
ient error response
[status code] 404
[reason phrase] Not Found
[url] http://169.254.169.254/latest/meta-data/iam/security-credentials/)
[ec2-user@ip-172-31-21-200 face]$ ^C
[ec2-user@ip-172-31-21-200 face]$ php --version
PHP 5.4.16 (cli) (built: Oct 31 2019 18:34:05)
Copyright (c) 1997-2013 The PHP Group
Zend Engine v2.4.0, Copyright (c) 1998-2013 Zend Technologies
[ec2-user@ip-172-31-21-200 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-bot-trial.s3.us-east-2.amazona
ws.com/sample.jpg[ec2-user@ip-172-31-21-200 face]$
```

## SCREENSHOTS NEEDED FOR EC2 & REKOGNITION

### 1. Face Detect success screenshot



```
ec2-user@ip-172-31-21-200:/var/www/html/face
Package operations: 7 installs, 1 update, 0 removals
- Installing symfony/polyfill-mbstring (v1.15.0): Downloading (100%)
- Installing mtdowling/jmespath.php (2.5.0): Downloading (100%)
- Installing guzzlehttp/promises (v1.3.1): Downloading (100%)
- Installing ralouphie/getallheaders (3.0.3): Downloading (100%)
- Installing psr/http-message (1.0.1): Downloading (100%)
- Installing guzzlehttp/psr7 (1.6.1): Downloading (100%)
- Installing guzzlehttp/guzzle (6.5.2): Downloading (100%)
- Updating aws/aws-sdk-php (2.8.31 => 3.134.3): Downloading (100%)
guzzlehttp/psr7 suggests installing zendframework/zend-httpdierrunner (Emit P
SR-7 responses)
guzzlehttp/guzzle suggests installing psr/log (Required for using the Log middle
ware)
guzzlehttp/guzzle suggests installing ext-intl (Required for Internationalized D
omain Name (IDN) support)
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/gu
zle instead.
Writing lock file
Generating autoload files
1 package you are using is looking for funding.
Use the `composer fund` command to find out more!
[ec2-user@ip-172-31-21-200 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-bot-trial.s3.us-east-2.amazona
ws.com/sample.jpgTotally there are 9 faces[ec2-user@ip-172-31-21-200 face]$
```