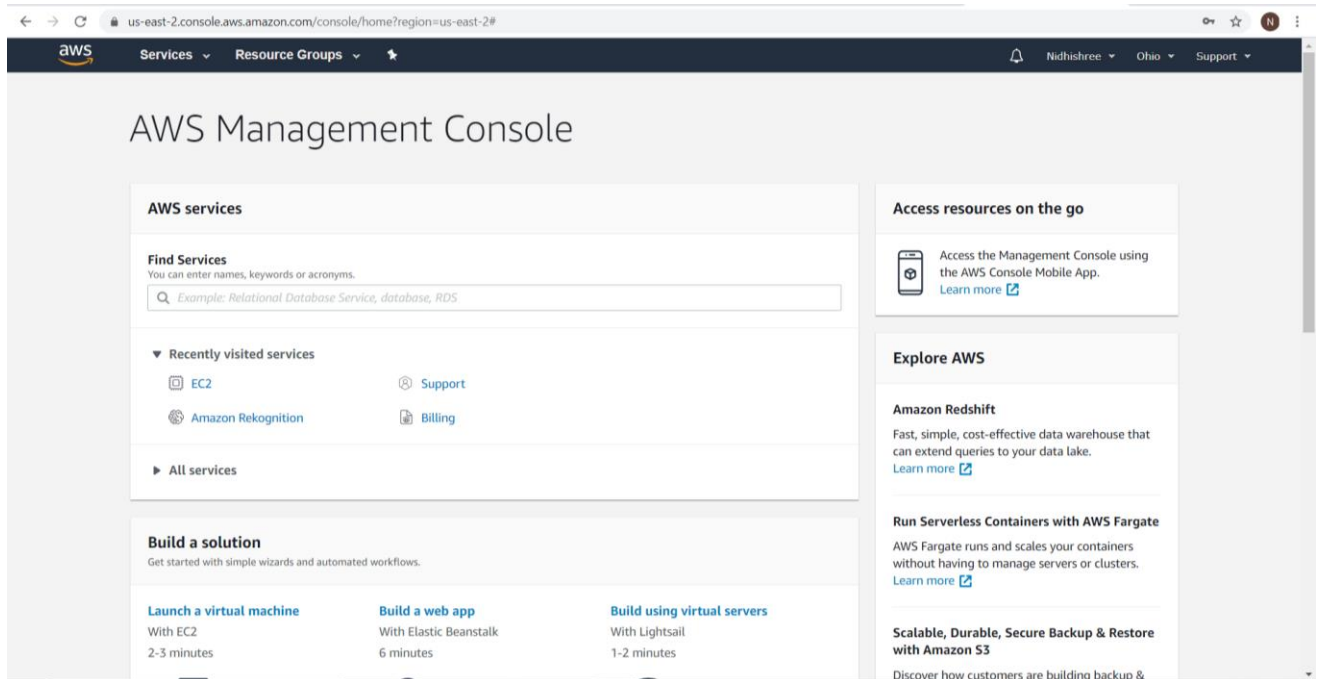
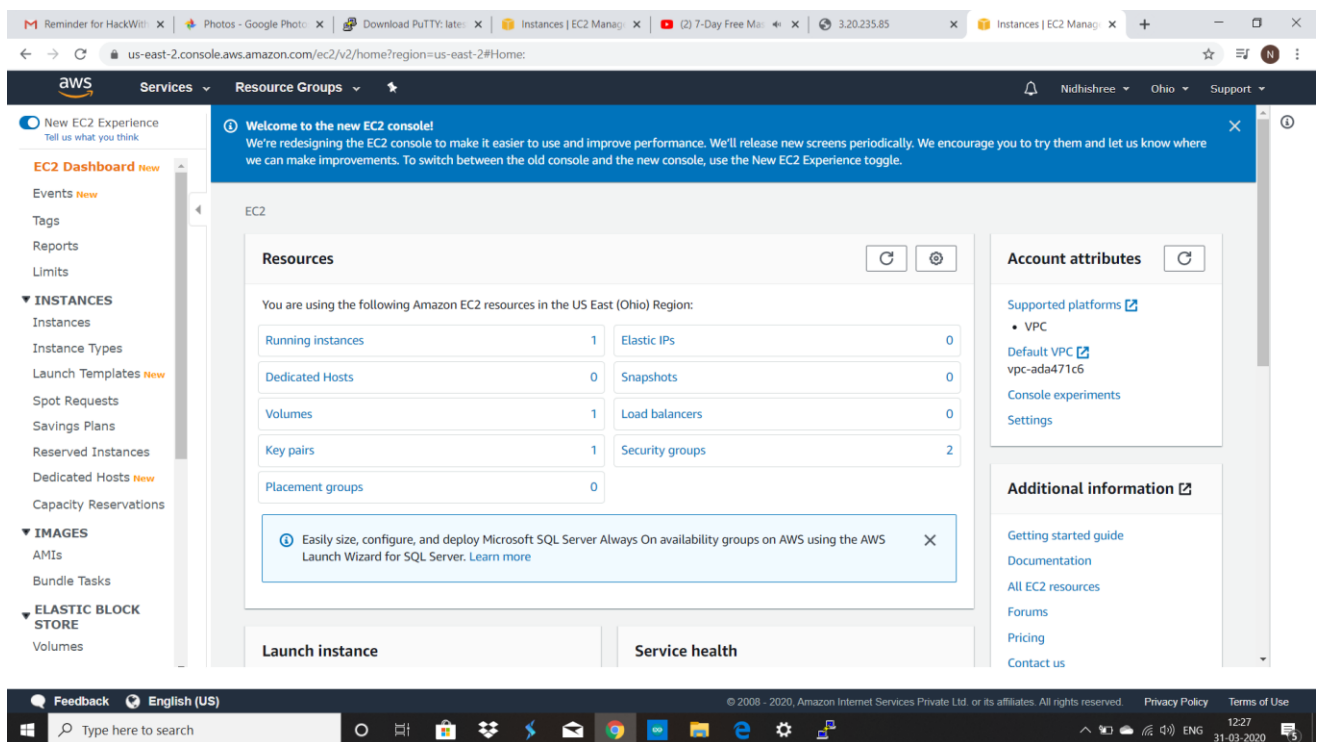


# Dashboards:

## 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', and 'Resource Groups'. A left-hand sidebar contains links to 'Buckets', 'Batch operations', 'Access analyzer for S3', 'Block public access (account settings)', and 'Feature spotlight'. The main content area displays 'Buckets (1)' with a search bar and buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'. Below this is a table with one bucket:

Name	Region	Access	Bucket created
aws-my-project-1	US East (Ohio) us-east-2	Objects can be public	2020-03-31T11:12:28.000Z

The bottom of the screen shows a Windows taskbar with various application icons and a system clock indicating 23:16 on 03-04-2020.

### 4. Rekognition Dashboard

The screenshot displays the Amazon Rekognition console. The left sidebar lists navigation options: 'Amazon Rekognition', 'Custom Labels', 'Demos', 'Image moderation', 'Facial analysis', 'Celebrity recognition', 'Face comparison', 'Text in image', 'Video Demos', 'Metrics', and 'Additional Resources'. The main content area has a header for 'Amazon Rekognition' with the tagline 'Deep learning-based visual analysis service' and a 'Try Demo' button. Below the header, there are three columns of information:

- Easily Integrate Powerful Visual Analysis into Your App**: You don't need computer vision or deep learning expertise to take advantage of Rekognition's high quality image and video analysis for your web, mobile, enterprise or device applications. Amazon Rekognition removes the complexity of building visual recognition capabilities by making powerful
- Continuously Learning**: Amazon Rekognition is designed to use deep learning technology to analyze billions of images and videos daily. It is continuously learning as we add support for new capabilities and learn from more and more data.
- Integrated with AWS Services**: Amazon Rekognition is designed to work seamlessly with other AWS services. Rekognition integrates directly with Amazon S3 and AWS Lambda so you can build scalable, affordable, and reliable visual analysis applications. You can start analyzing images and videos stored in Amazon S3 without moving any data. You can also run real-time

The bottom of the screen shows a Windows taskbar with various application icons and a system clock indicating 23:19 on 03-04-2020.

## EC2:

### 1. Choosing an AMI

The screenshot shows the AWS Management Console interface for the 'Launch Instance Wizard'. The 'Choose AMI' step is active, displaying a search bar and a list of available AMIs. The 'Quick Start' section on the left includes 'My AMIs', 'AWS Marketplace', and 'Community AMIs'. The main list shows three AMIs: Amazon Linux 2 AMI (HVM, SSD Volume Type), Amazon Linux AMI 2018.03.0 (HVM, SSD Volume Type), and Red Hat Enterprise Linux 8 (HVM, SSD Volume Type). Each AMI entry includes its name, description, root device type, virtualization type, and a 'Select' button. The bottom of the console shows the Windows taskbar with various application icons and the system clock.

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

Quick Start

My AMIs

AWS Marketplace

Community AMIs

Free tier only

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0e01ce4ee18447327 (64-bit x86) / ami-03201f374ab66a26e (64-bit Arm)

Amazon Linux

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 ENA Enabled: Yes

Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-01b01bbd08f24c7a8

Amazon Linux

Free tier eligible

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

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

Red Hat

Free tier eligible

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

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

SIUE Linux Enterprise Server 15 SP1 (HVM), SSD Volume Type - ami-04e5eb51ce148076 (64-bit x86) / ami-02c73009019019171 (64-bit Arm)

### 2. Choosing an Instance Type

The screenshot shows the AWS Management Console interface for the 'Choose Instance Type' step in the EC2 Launch Wizard. The 'Choose Instance Type' step is active, displaying a table of instance types. The 'Filter by' section shows 'All instance types' and 'Current generation'. The table lists various instance types, including t2.nano, t2.micro, t2.small, t2.medium, t2.large, t2.xlarge, t2.2xlarge, and t3a.nano. The 't2.micro' instance type is highlighted as 'Free tier eligible'. The bottom of the console shows the Windows taskbar with various application icons and the system clock.

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
<input type="checkbox"/>	General purpose	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
<input type="checkbox"/>	General purpose	t3a.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

### 3. Adding Storage

us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:

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 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](#)

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

Type here to search

### 4. Configuring Security Group

us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:

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 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](#)

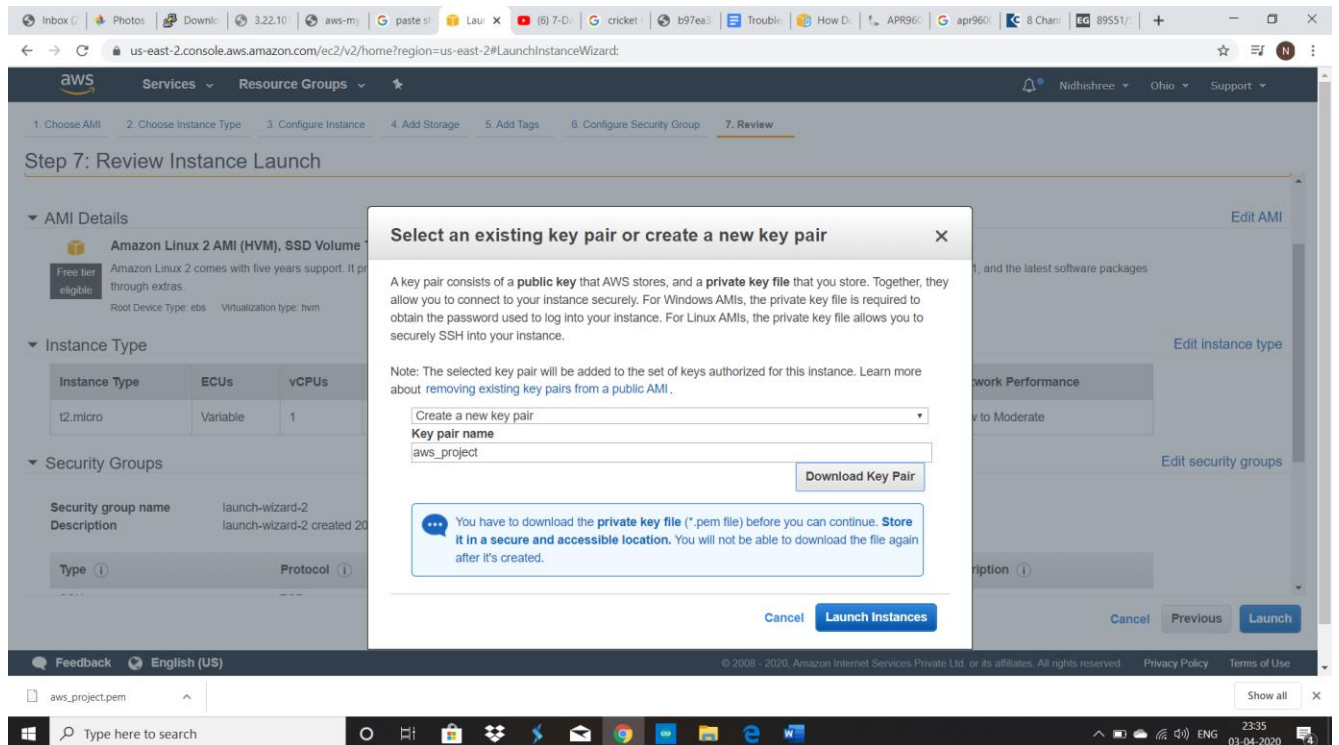
**Warning**  
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

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

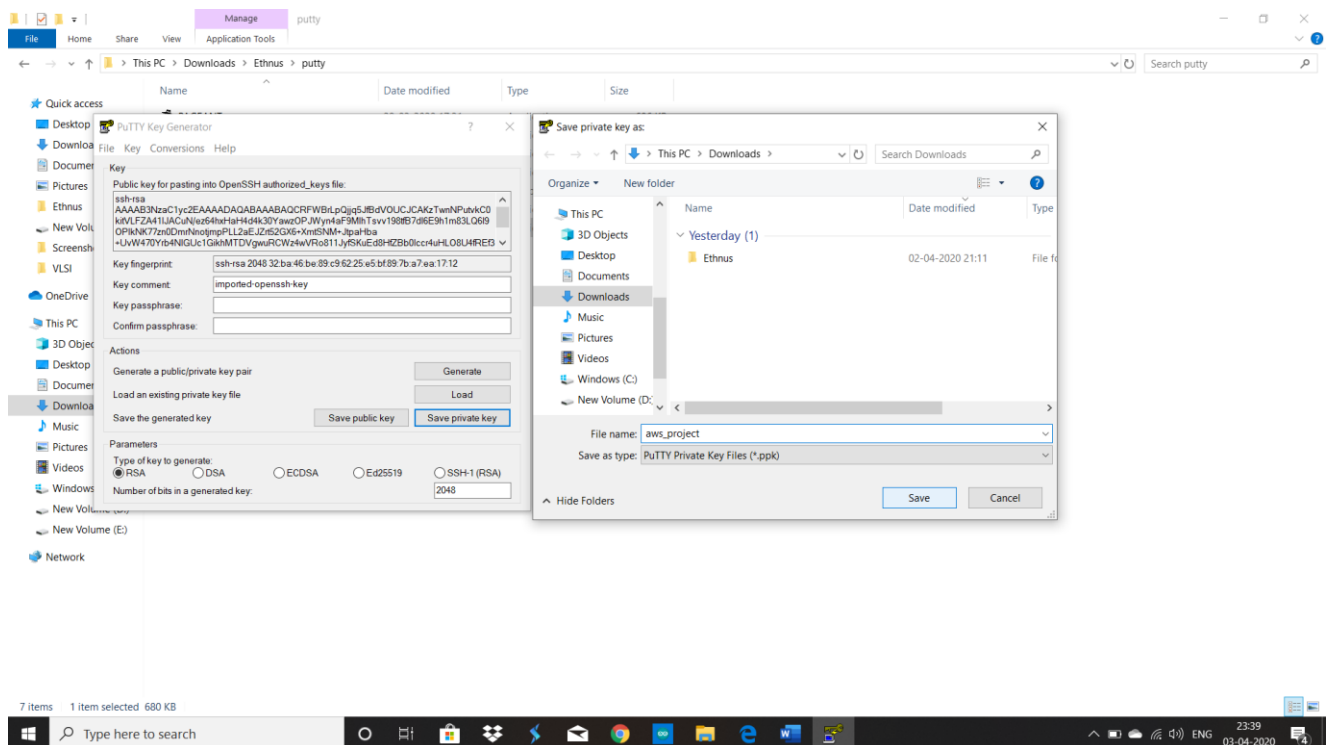
Feedback English (US) © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Type here to search

## 5. Key Pair Download



## 6. PuTTYgen Conversion from pem to ppk



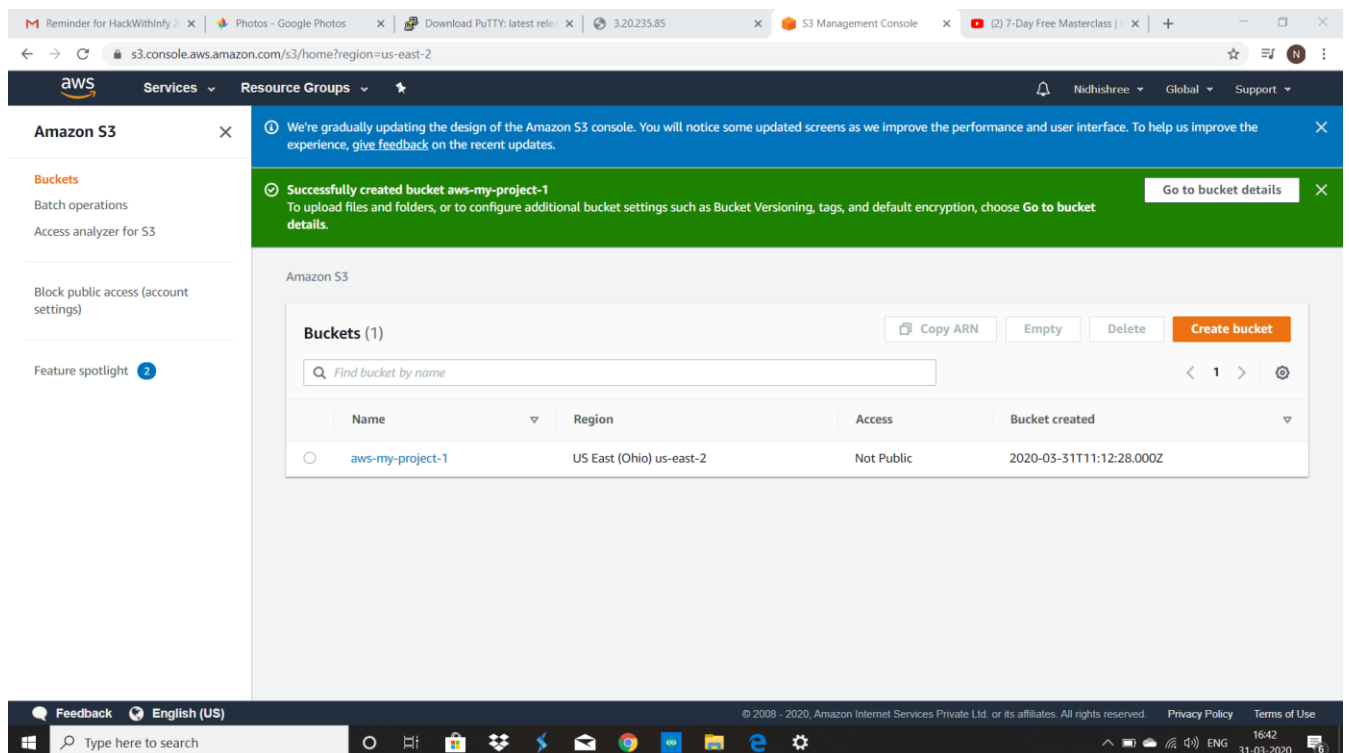


## 7. Logged in EC2 black Screen

```
ec2-user@ip-172-31-23-0:~  
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-172-31-23-0 ~]$
```

S3:

### 1. Creating a bucket



## 2. Uploading an object

The screenshot shows the AWS S3 console interface. The breadcrumb navigation indicates the path: Amazon S3 > aws-my-project-1. The bucket name 'aws-my-project-1' is displayed. Below the bucket name, there are tabs for Overview, Properties, Permissions, Management, and Access points. The Overview tab is selected. A search bar is present with the placeholder text 'Type a prefix and press Enter to search. Press ESC to clear.' Below the search bar, there are buttons for Upload, Create folder, Download, and Actions. The region is set to US East (Ohio). A table lists the objects in the bucket:

Name	Last modified	Size	Storage class
index.html	Mar 31, 2020 4:47:21 PM GMT+0530	34.0 B	Standard

Below the table, there is a status bar for the upload operation, showing '100% Successful' and '1 Success'.

## 3. Enabling Static Website

The screenshot shows the AWS S3 console interface with the Static website hosting configuration dialog open. The dialog has the following fields and options:

- Endpoint: `http://aws-my-project-1.s3-website-us-east-2.amazonaws.com`
- Use this bucket to host a website: ☒ (Learn more)
- Index document: `index.html`
- Error document: `error.html`
- Redirection rules (optional): (empty text area)
- Redirect requests: ☐ (Learn more)
- Disable website hosting: ☐
- Bucket hosting: ☒ (selected)

Buttons for Cancel and Save are at the bottom right of the dialog. The background shows the bucket 'aws-my-project-1' with tabs for Versioning, Server access logging, and Static website hosting. The Static website hosting tab is selected, and the status bar shows '2 Success'.

aws-my-project-1

Overview Properties Permissions Management Access points

**Versioning**  
Keep multiple versions of an object in the same bucket.  
[Learn more](#)  
Disabled

**Server access logging**  
Set up access log records that provide details about access requests.  
[Learn more](#)  
Disabled

**Static website hosting**  
Host a static website, which does not require server-side technologies.  
[Learn more](#)  
Bucket hosting

**Object-level logging**  
Record object-level API activity using the CloudTrail data events feature (additional cost).  
[Learn more](#)  
Disabled

**Default encryption**  
Automatically encrypt objects when stored in Amazon S3  
[Learn more](#)

Operations 0 In progress 1 Success 0 Error

#### 4. Making the object public

Amazon S3 > aws-my-project-1 > index.html

index.html Latest version

Overview Properties Permissions Select from

Open Download Download as Make public Copy path

**Owner**  
44d4e881e689233632b70533a89f6cb74b1d2d4d6d096899bcc796a7fc7d7bf7

**Last modified**  
Mar 31, 2020 4:47:21 PM GMT+0530

**Etag**  
31c6ca476ea108b454919633cd5b3207

**Storage class**  
Standard

**Server-side encryption**  
None

**Size**

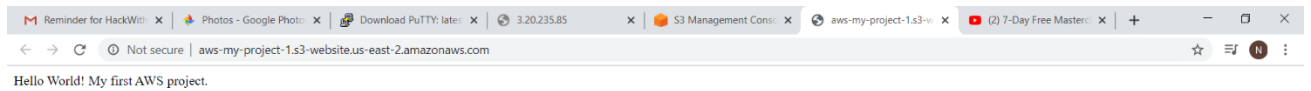
Upload 100% Successful  
[View details](#)

Make public 100% Successful  
[View details](#)

Operations 0 In progress 2 Success 0 Error

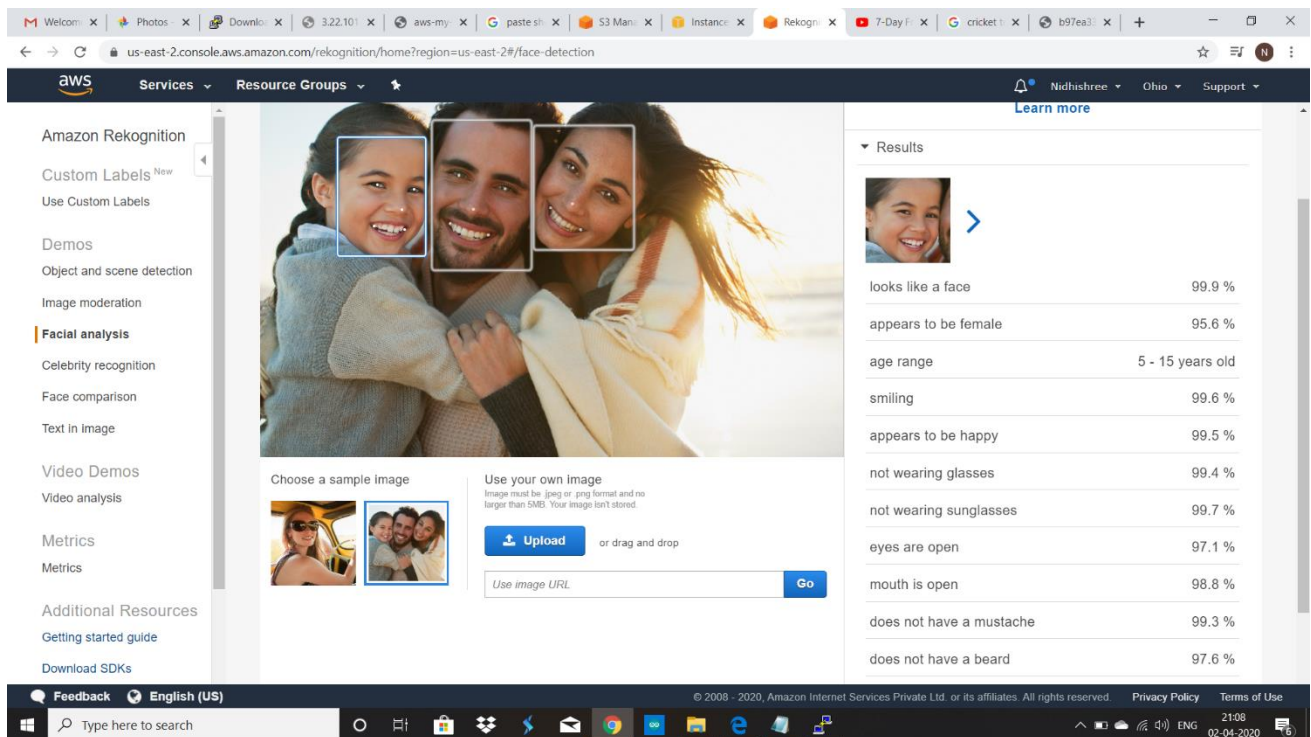


## 5. Checking the S3 link on the browser



## Rekognition :

### 1. Face Detect

A screenshot of the Amazon Rekognition console. The left sidebar shows the navigation menu with options like 'Amazon Rekognition', 'Custom Labels', 'Demos', 'Object and scene detection', 'Image moderation', 'Facial analysis' (selected), 'Celebrity recognition', 'Face comparison', 'Text in image', 'Video Demos', 'Video analysis', 'Metrics', and 'Additional Resources'. The main content area displays a large image of a family (a man, a woman, and a child) with bounding boxes around their faces. Below this image, there are two sections: 'Choose a sample image' with a small thumbnail, and 'Use your own image' with an 'Upload' button and a text input field for 'Use image URL'. On the right, a 'Results' panel lists various facial attributes and their confidence scores. The Windows taskbar is visible at the bottom.

Attribute	Confidence
looks like a face	99.9 %
appears to be female	95.6 %
age range	5 - 15 years old
smiling	99.6 %
appears to be happy	99.5 %
not wearing glasses	99.4 %
not wearing sunglasses	99.7 %
eyes are open	97.1 %
mouth is open	98.8 %
does not have a mustache	99.3 %
does not have a beard	97.6 %

## 2. Face Compare

The screenshot shows the Amazon Rekognition console's 'Face comparison' page. The left sidebar lists various services, with 'Face comparison' highlighted. The main area is titled 'Face comparison' and includes a description: '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 are 'Choose a sample image' and 'Choose a sample image' buttons. The 'Results' section on the right shows a comparison of two faces with a similarity score of 99.8%. The interface is in English (US) and includes a feedback button and a search bar at the bottom.

Amazon Rekognition

Custom Labels <sup>New</sup>

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

Face comparison

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

Reference face

Comparison faces

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

Results

Similarity 99.8 %

Choose a sample image

Choose a sample image

Feedback English (US)

Type here to search

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

## 3. Celebrity Recognition

The screenshot shows the Amazon Rekognition console's 'Celebrity recognition' page. The left sidebar lists various services, with 'Celebrity recognition' highlighted. The main area is titled 'Celebrity recognition' and includes a description: 'Rekognition automatically recognizes celebrities in images and provides confidence scores.' It features a large image upload section with a 'Choose a sample image' button and a 'Use your own image' section with an 'Upload' button. The 'Results' section on the right shows a recognition of Jeff Bezos with a match confidence of 100%. The interface is in English (US) and includes a feedback button and a search bar at the bottom.

Amazon Rekognition

Custom Labels <sup>New</sup>

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

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 5MB. Your image isn't stored.

Upload or drag and drop

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

Results

Jeff Bezos [Learn More](#)

Match confidence 100 %

Request

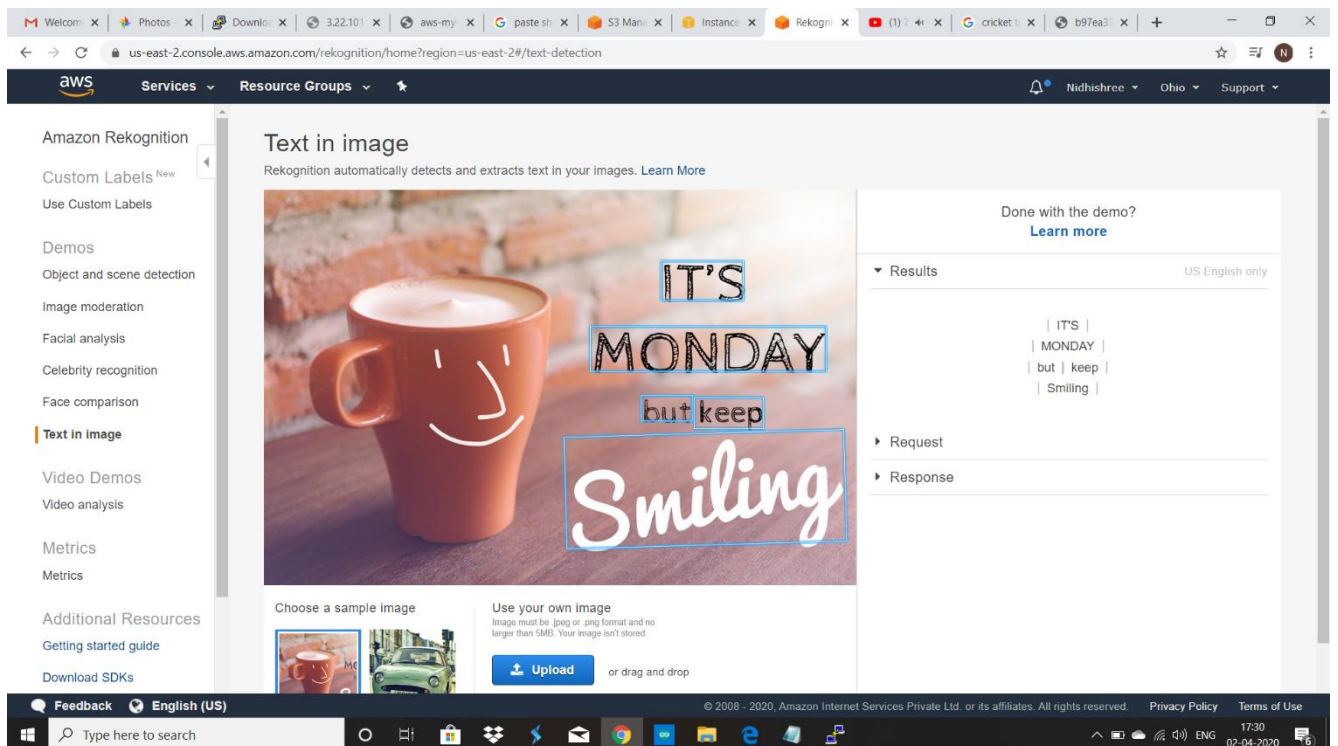
Response

Feedback English (US)

Type here to search

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

#### 4. Text in Image



## EC2 and S3:

## 1. Installing aws-sdk

```

[ec2-user@ip-172-31-23-0 /var/www/html]# cd /var/www/html
[ec2-user@ip-172-31-23-0 html]$ sudo mkdir face
[ec2-user@ip-172-31-23-0 html]$ cd face
[ec2-user@ip-172-31-23-0 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 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): Downloading (100%)
    proc_open(): fork failed - Cannot allocate memory
    The archive may contain identical file names with different capitalization (
    which fails on case insensitive filesystems)
    Unzip with unzip command failed, falling back to ZipArchive class

Installation failed, deleting ./composer.json.
The following exception is caused by a lack of memory or swap, or not having swa
p configured
check https://getcomposer.org/doc/articles/troubleshooting.md#proc-open-fork-fa
iled-errors for details

PHP Warning:  proc_open(): fork failed - Cannot allocate memory in phar:///home/
ec2-user/composer.phar/vendor/symfony/console/Application.php on line 952

Warning: proc_open(): fork failed - Cannot allocate memory in phar:///home/ec2-u
ser/composer.phar/vendor/symfony/console/Application.php on line 952

[ErrorException]
proc_open(): fork failed - Cannot allocate memory

require [--dev] [--prefer-source] [--prefer-dist] [--fixed] [--no-progress] [--n
o-suggest] [--no-update] [--no-scripts] [--update-no-dev] [--update-with-depende
ncies] [--update-with-all-dependencies] [--ignore-platform-reqs] [--prefer-stabl
e] [--prefer-lowest] [--sort-packages] [-o] [--optimize-autoloader] [-a] [--classmap
-authoritative] [--apcu-autoloader] [--] []...

[ec2-user@ip-172-31-23-0 face]$ sudo /bin/dd if=/dev/zero of=/var/swap.1 bs=1M c
ount=1024
1024+0 records in
1024+0 records out
1073741824 bytes (1.1 GB) copied, 13.8209 s, 77.7 MB/s
[ec2-user@ip-172-31-23-0 face]$ sudo /sbin/mkswap /var/swap.1
mkswap: /var/swap.1: insecure permissions 0644, 0600 suggested.
Setting up swapspace version 1, size = 1024 MiB (1073737728 bytes)
no label, UUID=eb2839ac2fb3-4410-9269-ac2ad905eb1
[ec2-user@ip-172-31-23-0 face]$ sudo /sbin/swapon /var/swap.1
swapon: /var/swap.1: insecure permissions 0644, 0600 suggested.
[ec2-user@ip-172-31-23-0 face]$

```

```
ec2-user@ip-172-31-23-0:/var/www/html/face
led-errors for details

PHP Warning: proc_open(): fork failed - Cannot allocate memory in phar:///home/
ec2-user/composer.phar/vendor/symfony/console/Application.php on line 952

Warning: proc_open(): fork failed - Cannot allocate memory in phar:///home/ec2-u
ser/composer.phar/vendor/symfony/console/Application.php on line 952

[ErrorException]
proc_open(): fork failed - Cannot allocate memory

require [--dev] [--prefer-source] [--prefer-dist] [--fixed] [--no-progress] [--n
o-suggest] [--no-update] [--no-scripts] [--update-no-dev] [--update-with-depende
ncies] [--update-with-all-dependencies] [--ignore-platform-reqs] [--prefer-stabl
e] [--prefer-lowest] [--sort-packages] [-o|--optimize-autoloader] [-a|--classmap
-authoritative] [--apcu-autoloader] [--] [<packages>]...

[ec2-user@ip-172-31-23-0 face]$ sudo /bin/dd if=/dev/zero of=/var/swap.1 bs=1M c
ount=1024
1024+0 records in
1024+0 records out
1073741824 bytes (1.1 GB) copied, 13.8209 s, 77.7 MB/s
[ec2-user@ip-172-31-23-0 face]$ sudo /sbin/mkswap /var/swap.1
mkswap: /var/swap.1: insecure permissions 0644, 0600 suggested.
Setting up swapspace version 1, size = 1024 MiB (1073737728 bytes)
no label, UUID=e3b2839a-2fb3-4410-9269-ac2ad905e1b1
[ec2-user@ip-172-31-23-0 face]$ sudo /sbin/swapon /var/swap.1
swapon: /var/swap.1: insecure permissions 0644, 0600 suggested.
[ec2-user@ip-172-31-23-0 face]$ sudo php -d memory_limit=-1 ~composer.phar requ
ire 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 caching, request and response caching, and credentials caching)
aws/aws-sdk-php suggests installing monolog/monolog (Adds support for logging HTTP requests and responses)
aws/aws-sdk-php suggests installing symfony/yaml (Eases the ability to write manifests for creating jobs in AWS Import/Export)
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/guzzle instead.
Writing lock file
Generating autoload files
[ec2-user@ip-172-31-23-0 face]$
```

## 2. Installing php

```
ec2-user@ip-172-31-23-0:~
Installing : php-5.4.16-46.amzn2.0.2.x86_64 4/4
Verifying : php-5.4.16-46.amzn2.0.2.x86_64 1/4
Verifying : libzip010-compat-0.10.1-9.amzn2.0.5.x86_64 2/4
Verifying : php-cli-5.4.16-46.amzn2.0.2.x86_64 3/4
Verifying : php-common-5.4.16-46.amzn2.0.2.x86_64 4/4

Installed:
php.x86_64 0:5.4.16-46.amzn2.0.2

Dependency Installed:
libzip010-compat.x86_64 0:0.10.1-9.amzn2.0.5
php-cli.x86_64 0:5.4.16-46.amzn2.0.2
php-common.x86_64 0:5.4.16-46.amzn2.0.2

Complete!
[ec2-user@ip-172-31-23-0 ~]$ 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.pha
r
Use it: php composer.phar

[ec2-user@ip-172-31-23-0 ~]$
```



### 3. index.php file code

```
ec2-user@ip-172-31-23-0:/var/www/html/face
cd face
sudo php -d memory_limit=1 ~/composer.phar require aws/aws-sdk-php

In case if you get memory error -
sudo /bin/dd if=/dev/zero of=/var/swap.1 bs=1M count=1024
sudo /sbin/mkswap /var/swap.1
sudo /sbin/swapon /var/swap.1

sudo wget https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
sudo mv b97ea33b5842c7894b804923c6c05580.jpg sample.jpg

*/
error_reporting(0);

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

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

$bucket = 'aws-my-project-1';
$keyname = 'sample.jpg';

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

try {
    // Upload data.
    $result = $s3->putObject([
        'Bucket' => $bucket,
        'Key' => $keyname,
        'SourceFile' => __DIR__ . "/$keyname",
        'ACL' => 'public-read'
    ]);

    // Print the URL to the object.
    $imageUrl = $result['ObjectURL'];
    if ($imageUrl) {
        echo "Image upload done... Here is the URL: " . $imageUrl;
    }
} catch (Exception $e) {
    echo $e->getMessage() . PHP_EOL;
}

-- INSERT --
```

### 4. Upload Success screenshot

```
ec2-user@ip-172-31-23-0:/var/www/html/face
HTTP request sent, awaiting response... 200 OK
Length: 215551 (210K) [image/jpeg]
Saving to: 'b97ea33b5842c7894b804923c6c05580.jpg'

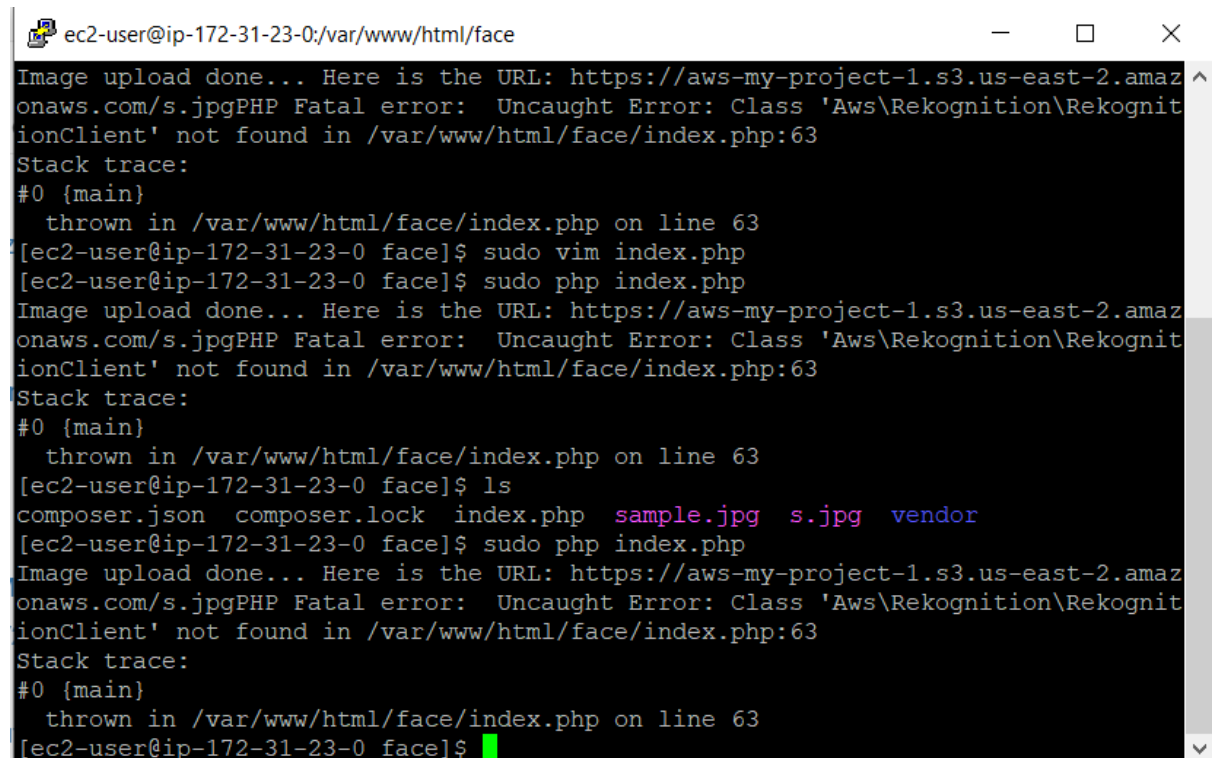
100%[=====>] 215,551 --.-K/s in 0.04s

2020-04-03 19:03:07 (4.74 MB/s) - 'b97ea33b5842c7894b804923c6c05580.jpg' saved [
215551/215551]

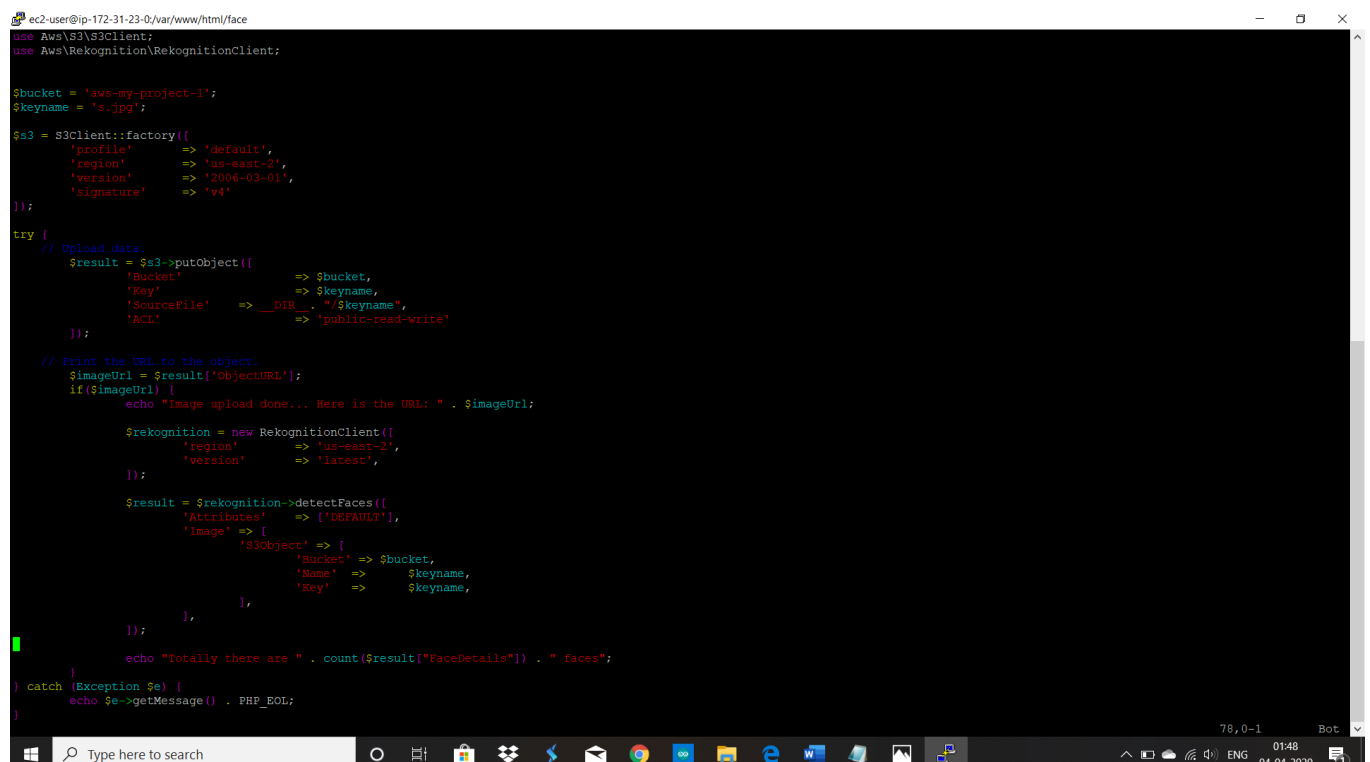
[ec2-user@ip-172-31-23-0 face]$ sudo mv b97ea33b5842c7894b804923c6c05580.jpg sam
ple.jpg
[ec2-user@ip-172-31-23-0 face]$ sudo vim index.php
[ec2-user@ip-172-31-23-0 face]$ sudo php index.php
Error retrieving credentials from the instance profile metadata server. When you
are not running inside of Amazon 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-172-31-23-0 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-my-project-1.s3.us-east-2.amaz
[ec2-user@ip-172-31-23-0 face]$
```

## EC2 and Rekognition:

### 1. Face Detect Success Screenshot



```
ec2-user@ip-172-31-23-0:/var/www/html/face
Image upload done... Here is the URL: https://aws-my-project-1.s3.us-east-2.amazonaws.com/s.jpg
PHP Fatal error:  Uncaught Error: Class 'Aws\Rekognition\RekognitionClient' not found in /var/www/html/face/index.php:63
Stack trace:
#0 {main}
   thrown in /var/www/html/face/index.php on line 63
[ec2-user@ip-172-31-23-0 face]$ sudo vim index.php
[ec2-user@ip-172-31-23-0 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-my-project-1.s3.us-east-2.amazonaws.com/s.jpg
PHP Fatal error:  Uncaught Error: Class 'Aws\Rekognition\RekognitionClient' not found in /var/www/html/face/index.php:63
Stack trace:
#0 {main}
   thrown in /var/www/html/face/index.php on line 63
[ec2-user@ip-172-31-23-0 face]$ ls
composer.json  composer.lock  index.php  sample.jpg  s.jpg  vendor
[ec2-user@ip-172-31-23-0 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-my-project-1.s3.us-east-2.amazonaws.com/s.jpg
PHP Fatal error:  Uncaught Error: Class 'Aws\Rekognition\RekognitionClient' not found in /var/www/html/face/index.php:63
Stack trace:
#0 {main}
   thrown in /var/www/html/face/index.php on line 63
[ec2-user@ip-172-31-23-0 face]$
```



```
ec2-user@ip-172-31-23-0:/var/www/html/face
use Aws\S3\S3Client;
use Aws\Rekognition\RekognitionClient;

$bucket = 'aws-my-project-1';
$keyname = 's.jpg';

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

try {
    // Upload data.
    $result = $s3->putObject([
        'Bucket' => $bucket,
        'Key' => $keyname,
        'SourceFile' => __DIR__ . "/" . $keyname,
        'ACL' => 'public-read-write'
    ]);

    // Print the URL to the object.
    $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;
}
```