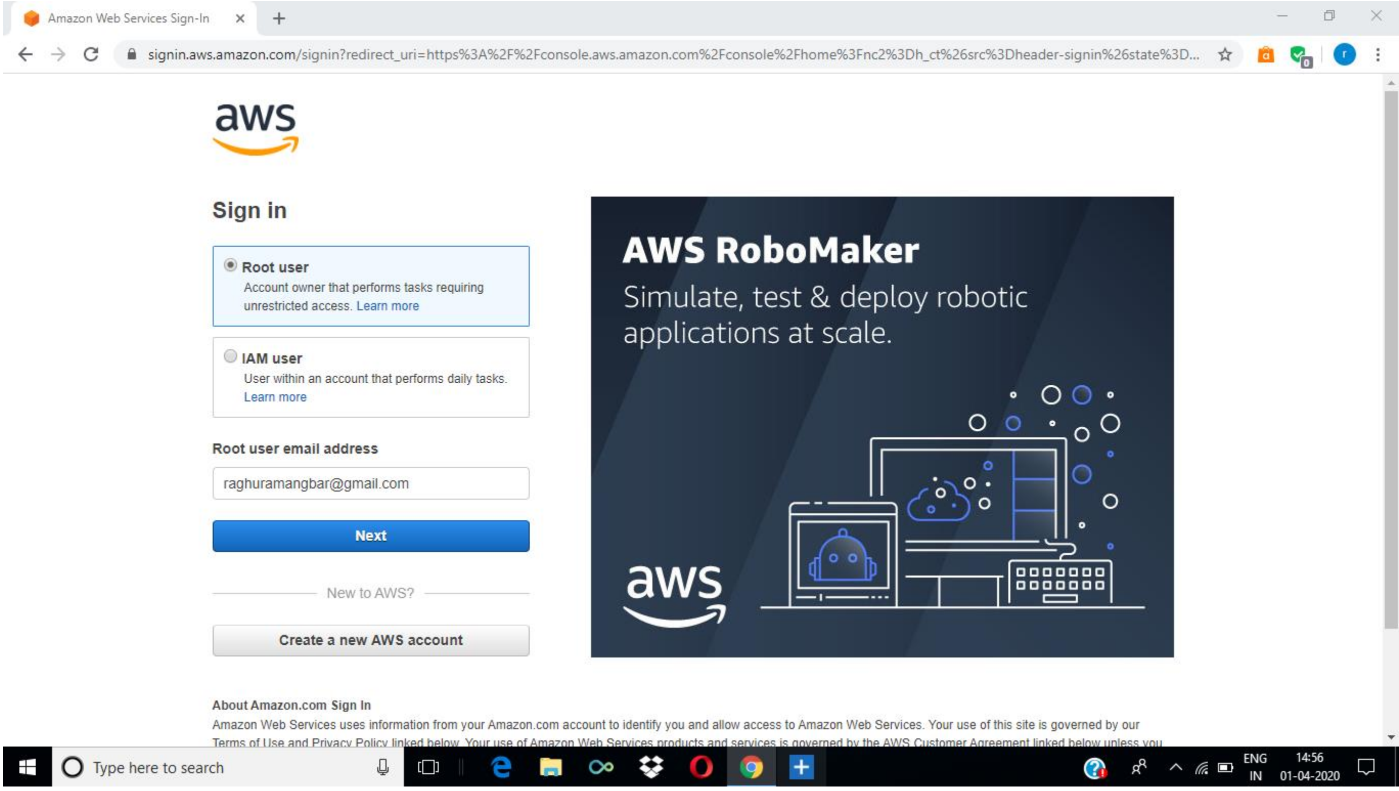
Building a Face Detection Application on Amazon Web Services

Project by Raghuraman G

1. AWS Login screen with username



Amazon Web Services Sign-In

signin.aws.amazon.com/signin?redirect_uri=https%3A%2F%2Fconsole.aws.amazon.com%2Fconsole%2Fhome%3Fnc2%3Dh_ct%26src%3Dheader-signin%26state%3D...

Sign in

● Root user
Account owner that performs tasks requiring unrestricted access. Learn more

○ IAM user
User within an account that performs daily tasks. Learn more

Root user email address

raghuramangbar@gmail.com

Next

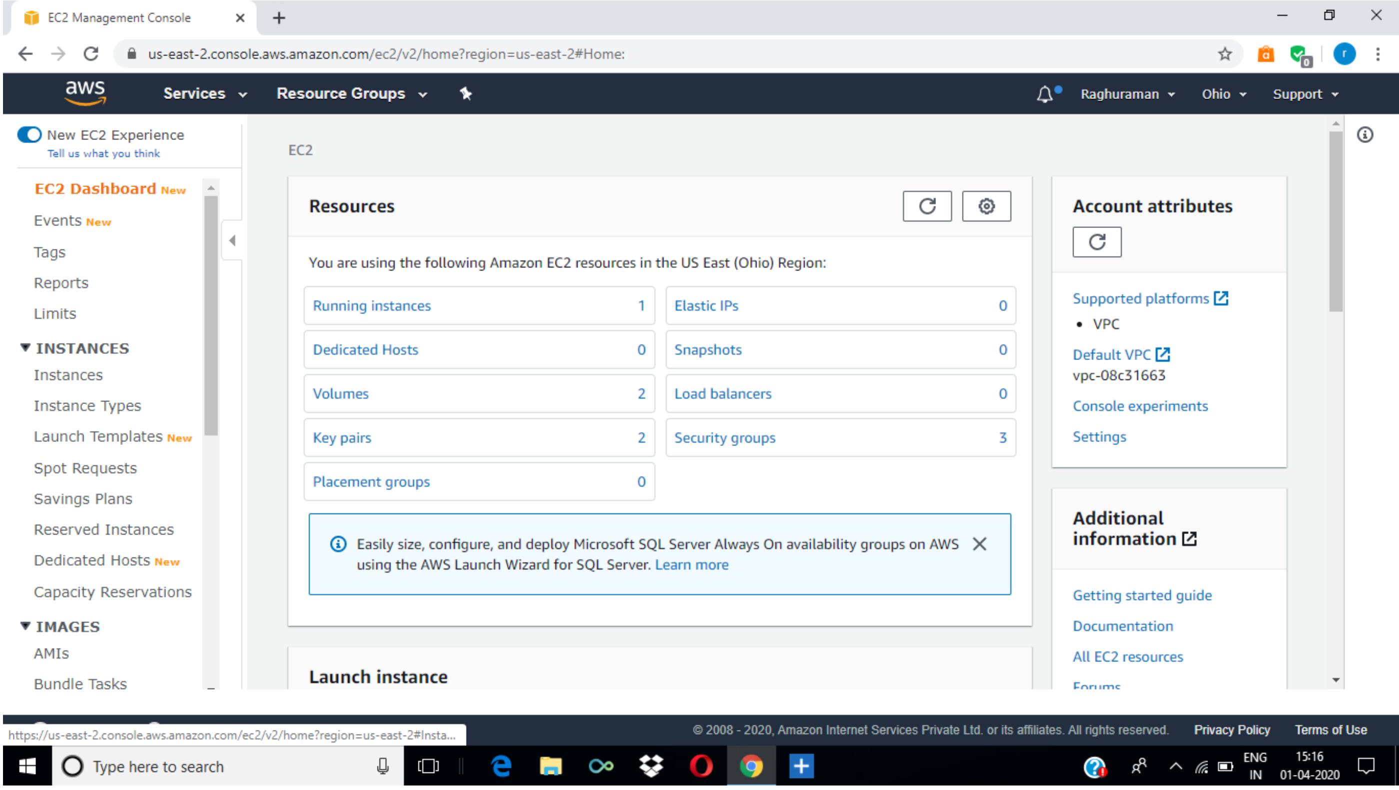
New to AWS?

Create a new AWS account

About Amazon.com Sign In

Amazon Web Services uses information from your Amazon.com account to identify you and allow access to Amazon Web Services. Your use of this site is governed by our Terms of Use and Privacy Policy linked below. Your use of Amazon Web Services products and services is governed by the AWS Customer Agreement linked below unless you

2. EC2 Dashboard



EC2 Management Console

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

Resources

You are using the following Amazon EC2 resources in the US East (Ohio) Region:

Running instances	1	Elastic IPs	0
Dedicated Hosts	0	Snapshots	0
Volumes	2	Load balancers	0
Key pairs	2	Security groups	3
Placement groups	0		

Easily size, configure, and deploy Microsoft SQL Server Always On availability groups on AWS using the AWS Launch Wizard for SQL Server. Learn more

Launch instance

Account attributes

Supported platforms
- VPC

Default VPC
vpc-08c31663

Console experiments

Settings

Additional information

Getting started guide

Documentation

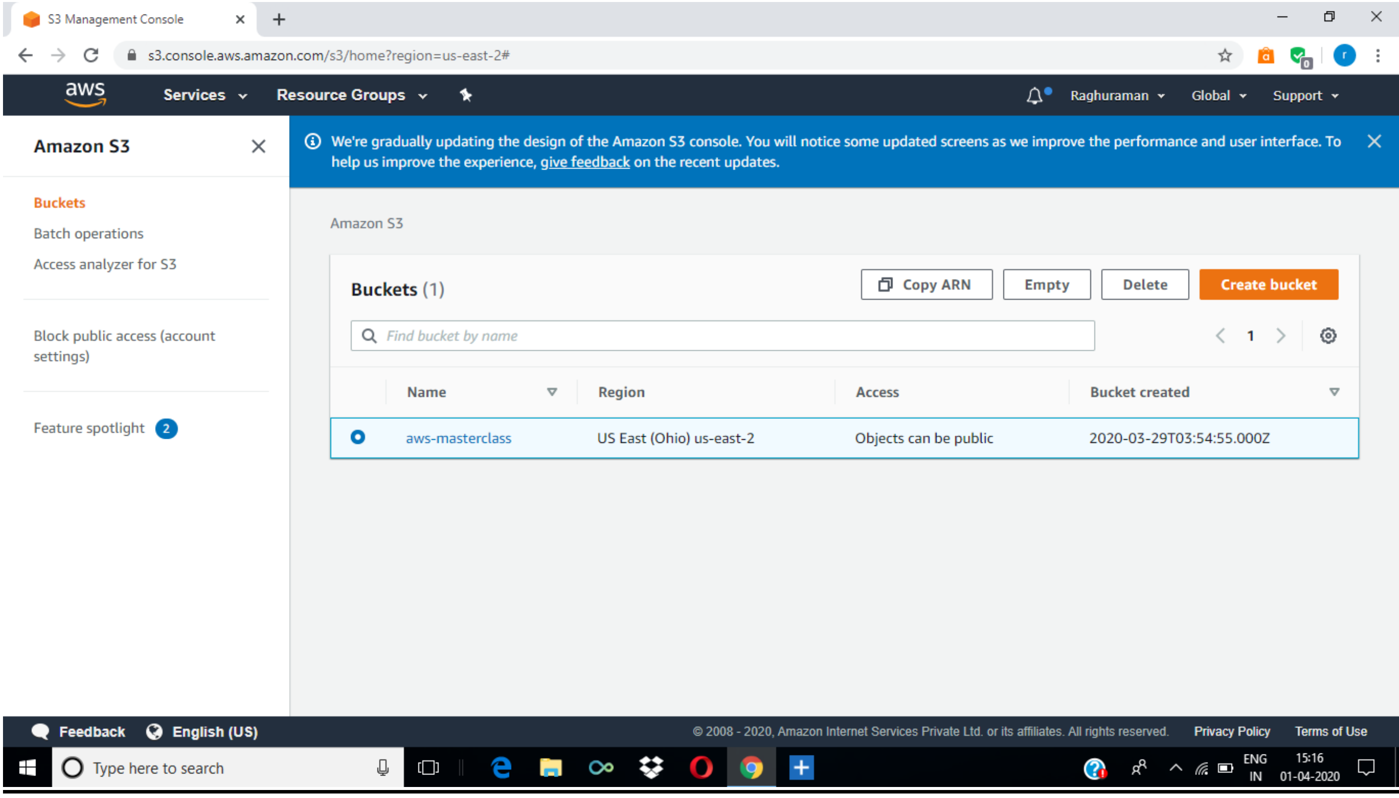
All EC2 resources

Forums

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

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

3. S3 Dashboard



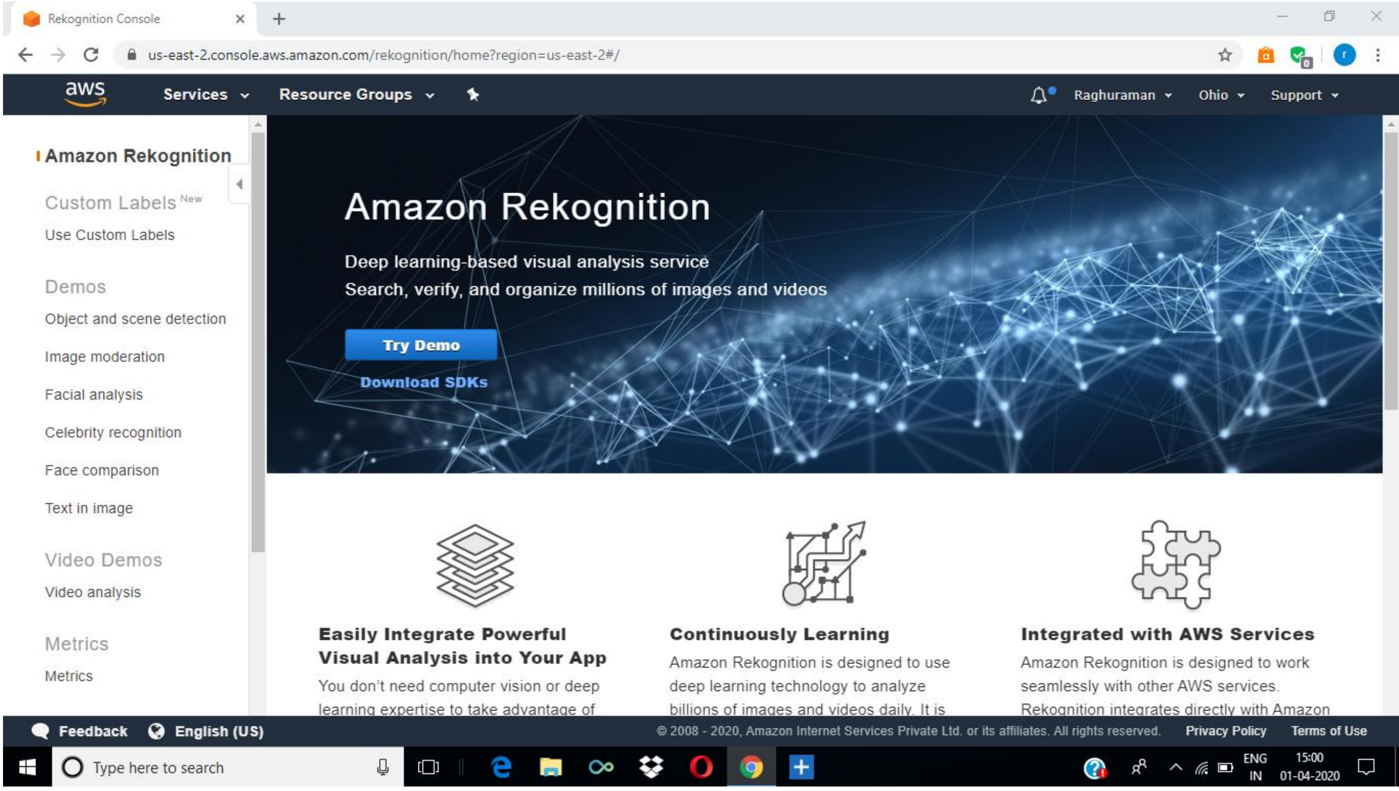
We're gradually updating the design of the Amazon S3 console. You will notice some updated screens as we improve the performance and user interface. To help us improve the experience, give feedback on the recent updates.

Buckets (1)

	Name	Region	Access	Bucket created
●	aws-masterclass	US East (Ohio) us-east-2	Objects can be public	2020-03-29T03:54:55.000Z

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

4. Rekognition Dashboard



Amazon Rekognition

Deep learning-based visual analysis service
Search, verify, and organize millions of images and videos

Easily Integrate Powerful Visual Analysis into Your App

You don't need computer vision or deep learning expertise to take advantage of

Continuously Learning

Amazon Rekognition is designed to use deep learning technology to analyze billions of images and videos daily. It is

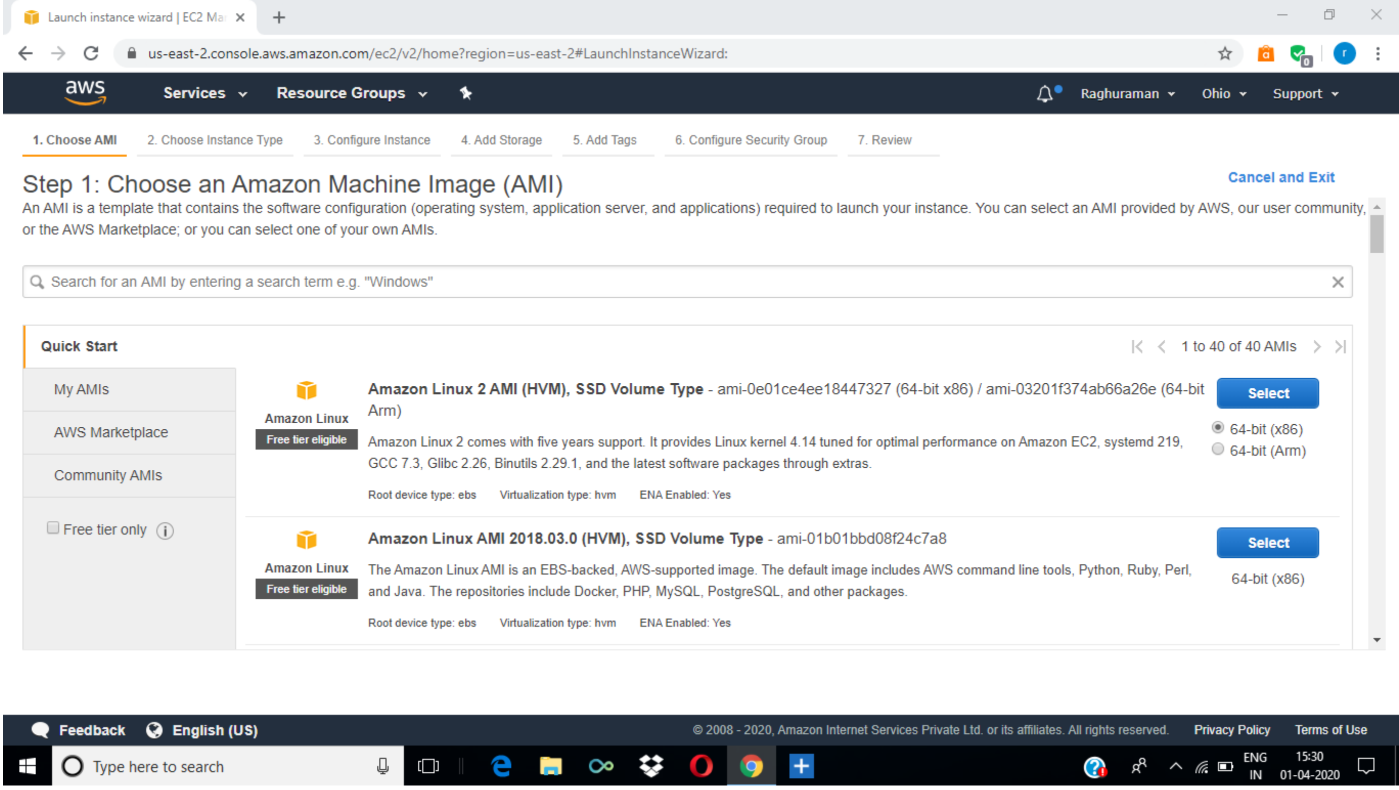
Integrated with AWS Services

Amazon Rekognition is designed to work seamlessly with other AWS services. Rekognition integrates directly with Amazon

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

EC2

1. Choosing an AMI



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"

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

Select ● 64-bit (x86) ○ 64-bit (Arm)

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

Amazon Linux — Free tier eligible

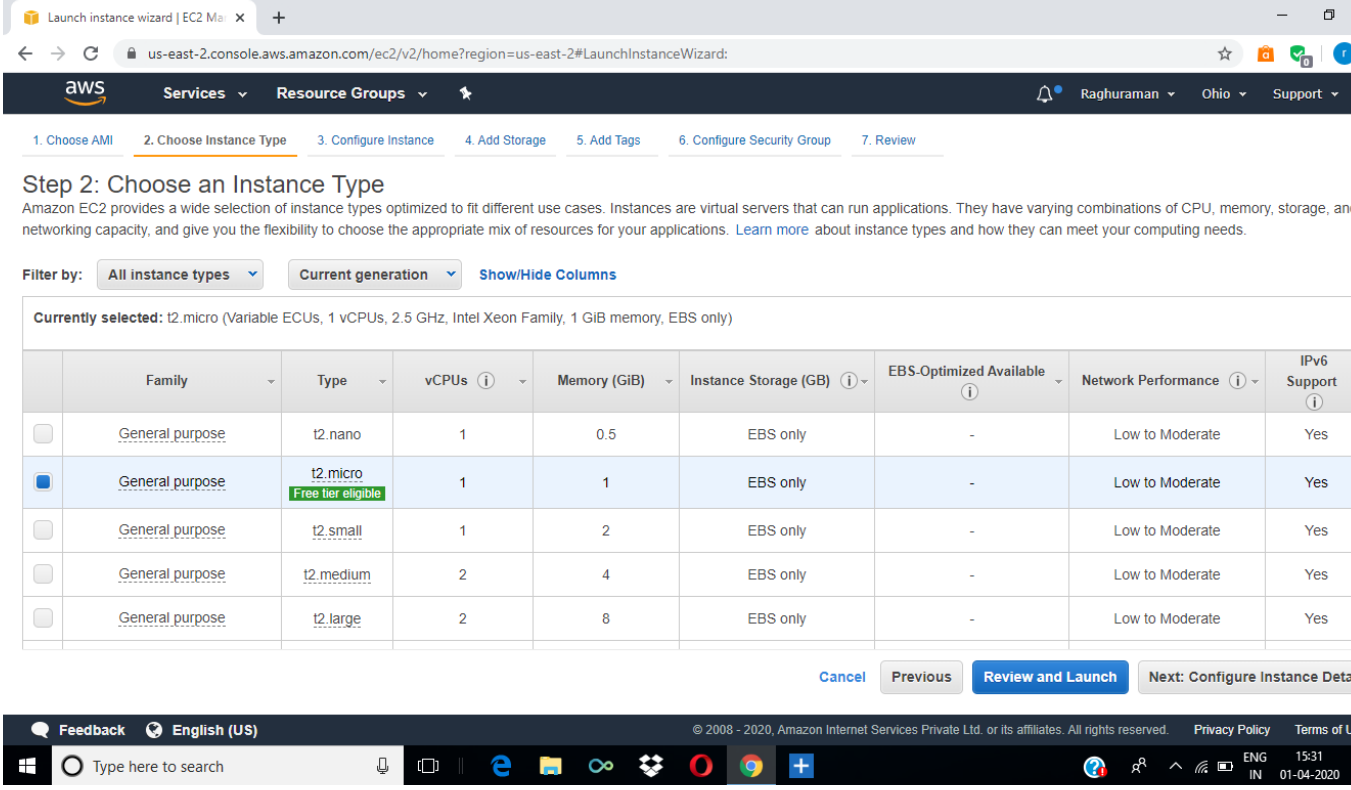
The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.

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

Select 64-bit (x86)

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
ENG IN
15:30
01-04-2020

2. Choosing an Instance Type



Launch instance wizard | EC2 Ma
us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:
aws
Services
Resource Groups
Raghuraman
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, an 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.

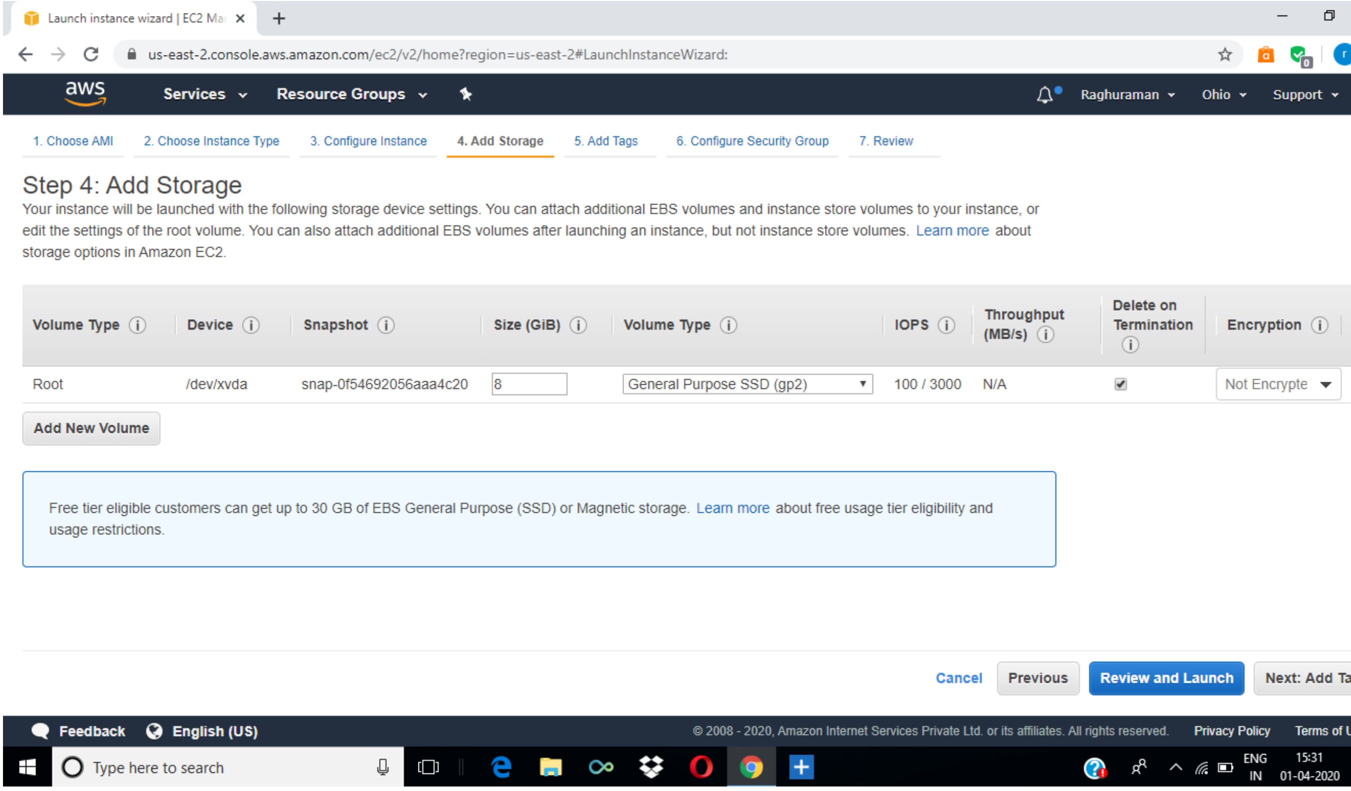
Filter by: All instance types | Current generation | Show/Hide Columns

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
□	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
☑	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
□	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
□	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
□	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes

Cancel
Previous
Review and Launch
Next: Configure Instance Deta
Feedback
English (US)
© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of U
Type here to search
ENG IN
15:31
01-04-2020

3. Adding Storage



Launch instance wizard | EC2 Ma
us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:
aws
Services
Resource Groups
Raghuraman
Ohio
Support
1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review

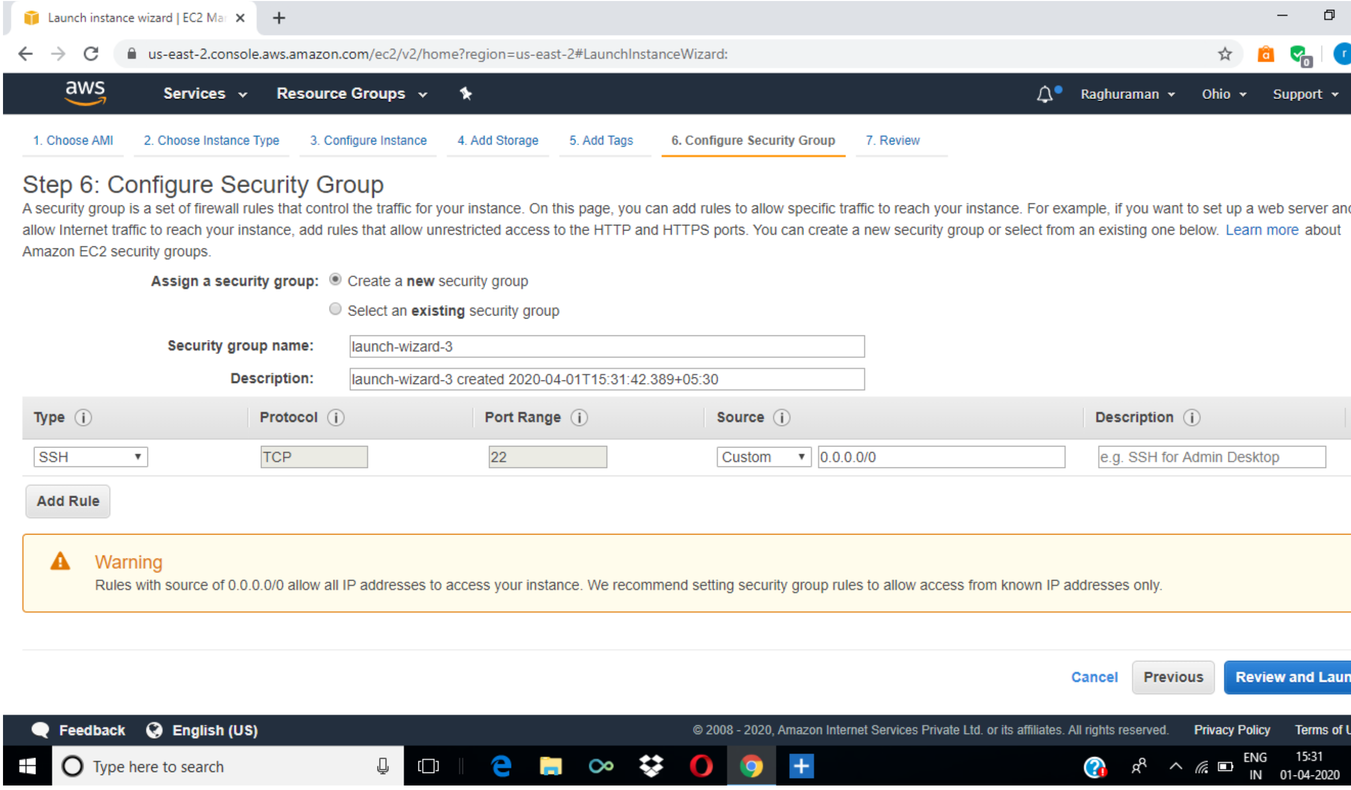
Step 4: Add Storage

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	☑	Not Encrypte

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.

4. Configuring Security Group



Launch instance wizard | EC2 Ma
us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:
aws
Services
Resource Groups
Raghuraman
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 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

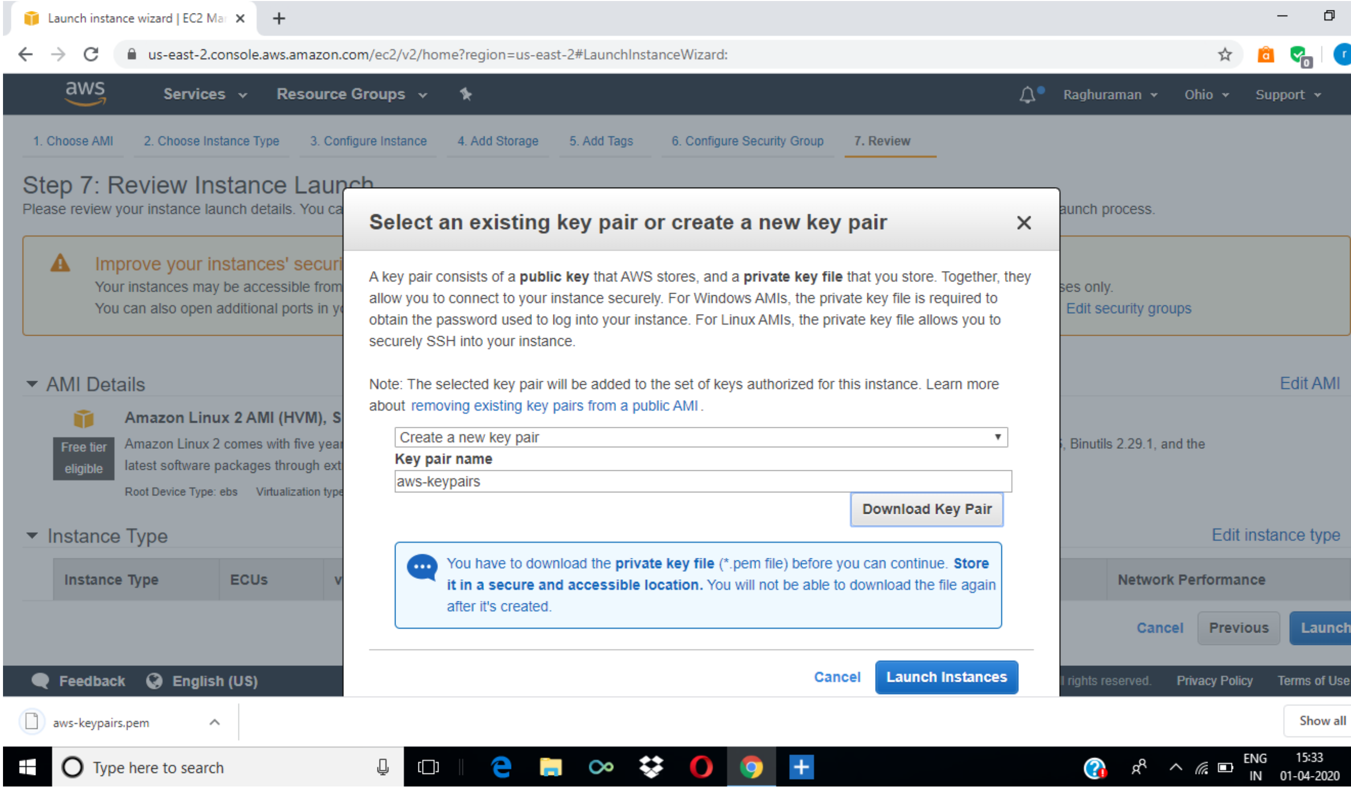
Security group name: launch-wizard-3

Description: launch-wizard-3 created 2020-04-01T15:31:42.389+05:30

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

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.

5. Key Pair Download



Launch instance wizard | EC2 Ma

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

Step 7: Review Instance Launch

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

aws-keypairs

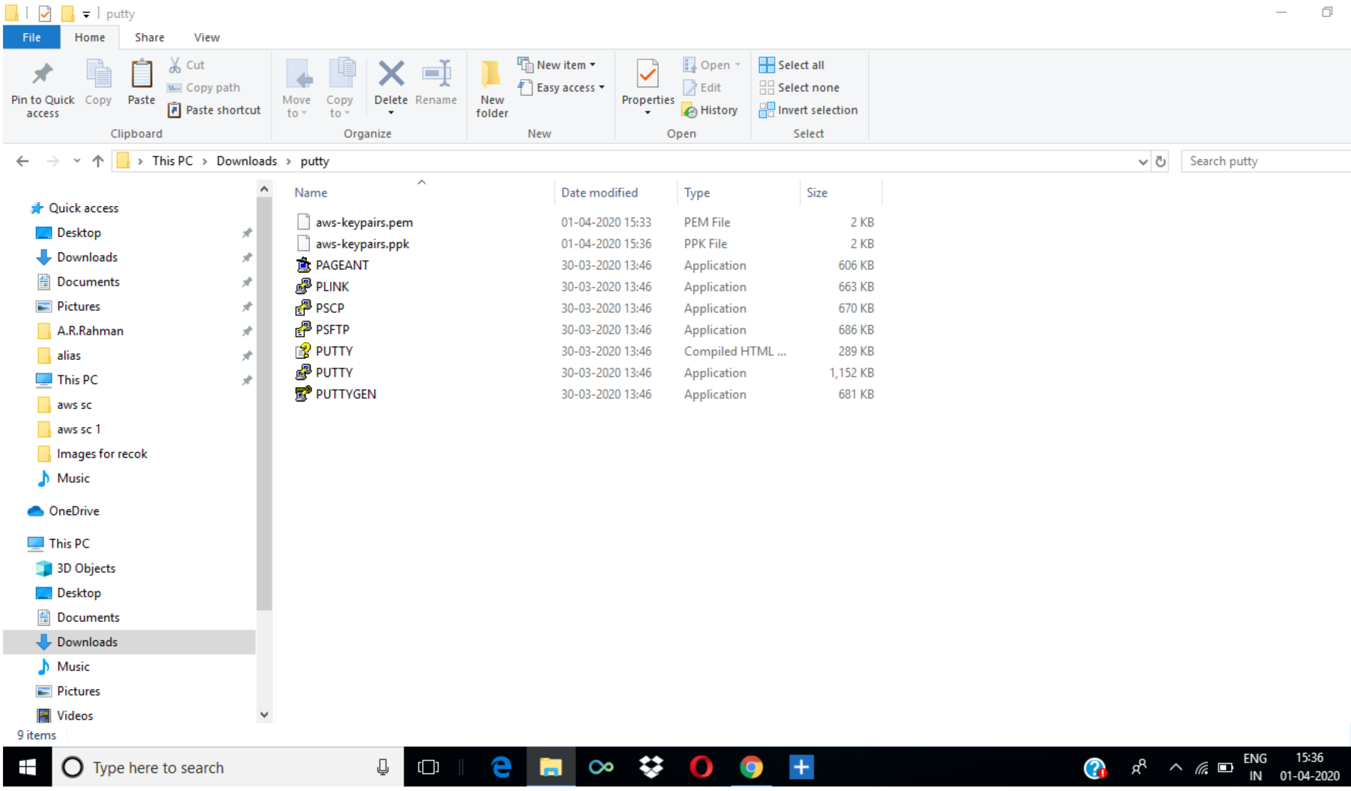
Download Key Pair

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

aws-keypairs.pem

6. puTTYgen conversion from pem to ppk

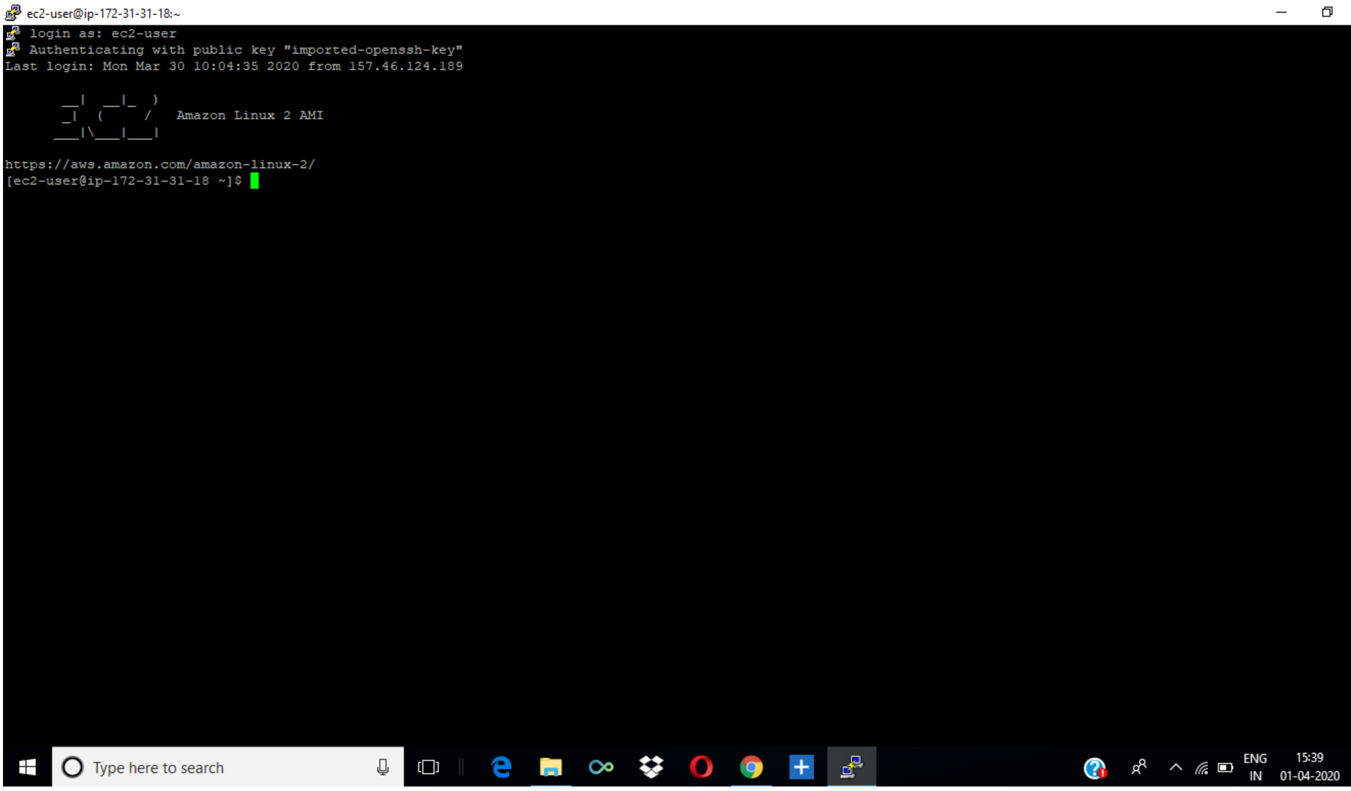


This PC > Downloads > putty

Name	Date modified	Type	Size
aws-keypairs.pem	01-04-2020 15:33	PEM File	2 KB
aws-keypairs.ppk	01-04-2020 15:36	PPK File	2 KB
PAGEANT	30-03-2020 13:46	Application	606 KB
PLINK	30-03-2020 13:46	Application	663 KB
PSCP	30-03-2020 13:46	Application	670 KB
PSFTP	30-03-2020 13:46	Application	686 KB
PUTTY	30-03-2020 13:46	Compiled HTML ...	289 KB
PUTTY	30-03-2020 13:46	Application	1,152 KB
PUTTYGEN	30-03-2020 13:46	Application	681 KB

9 items

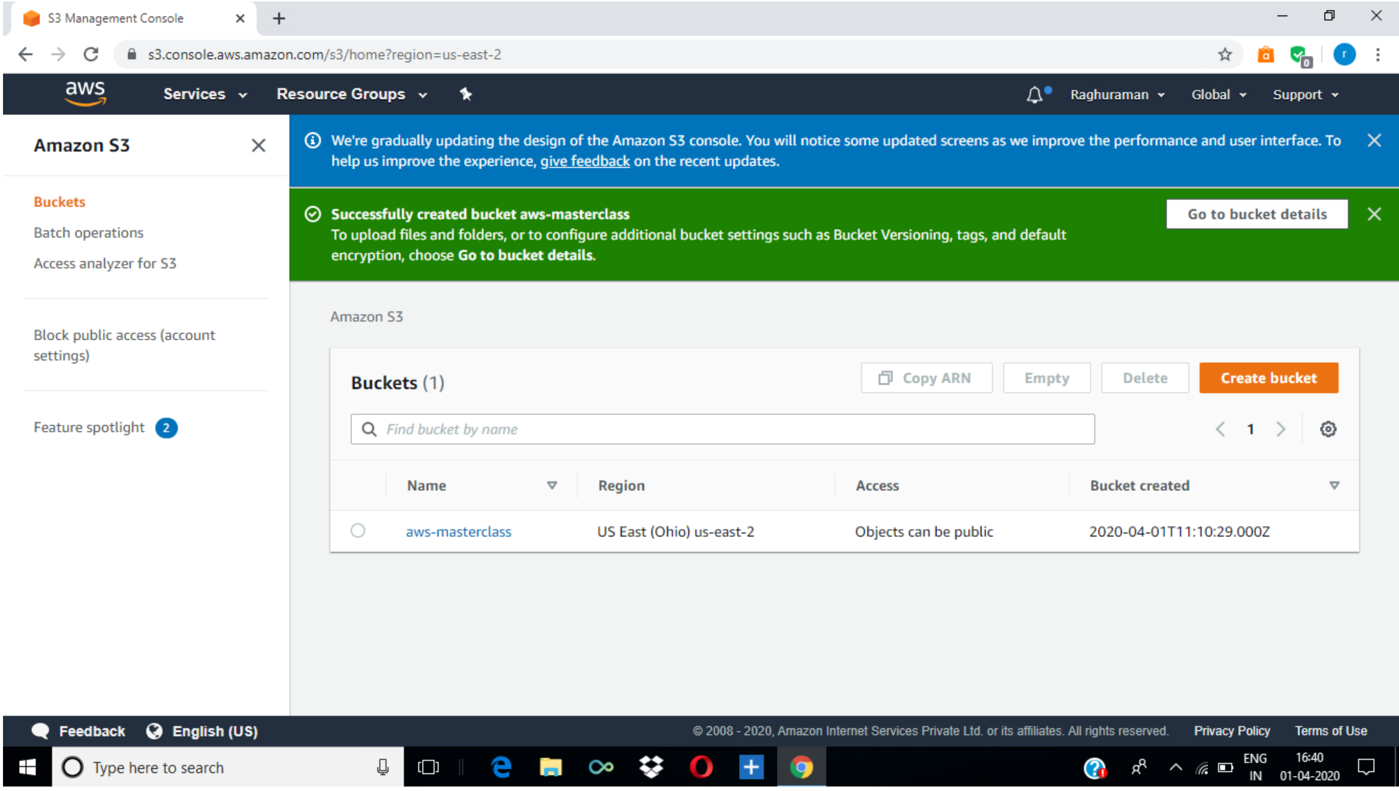
7. Logged in EC2 black screen



```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Mon Mar 30 10:04:35 2020 from 157.46.124.189

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-31-18 ~]$
```

S3

1. Creating a bucket



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.

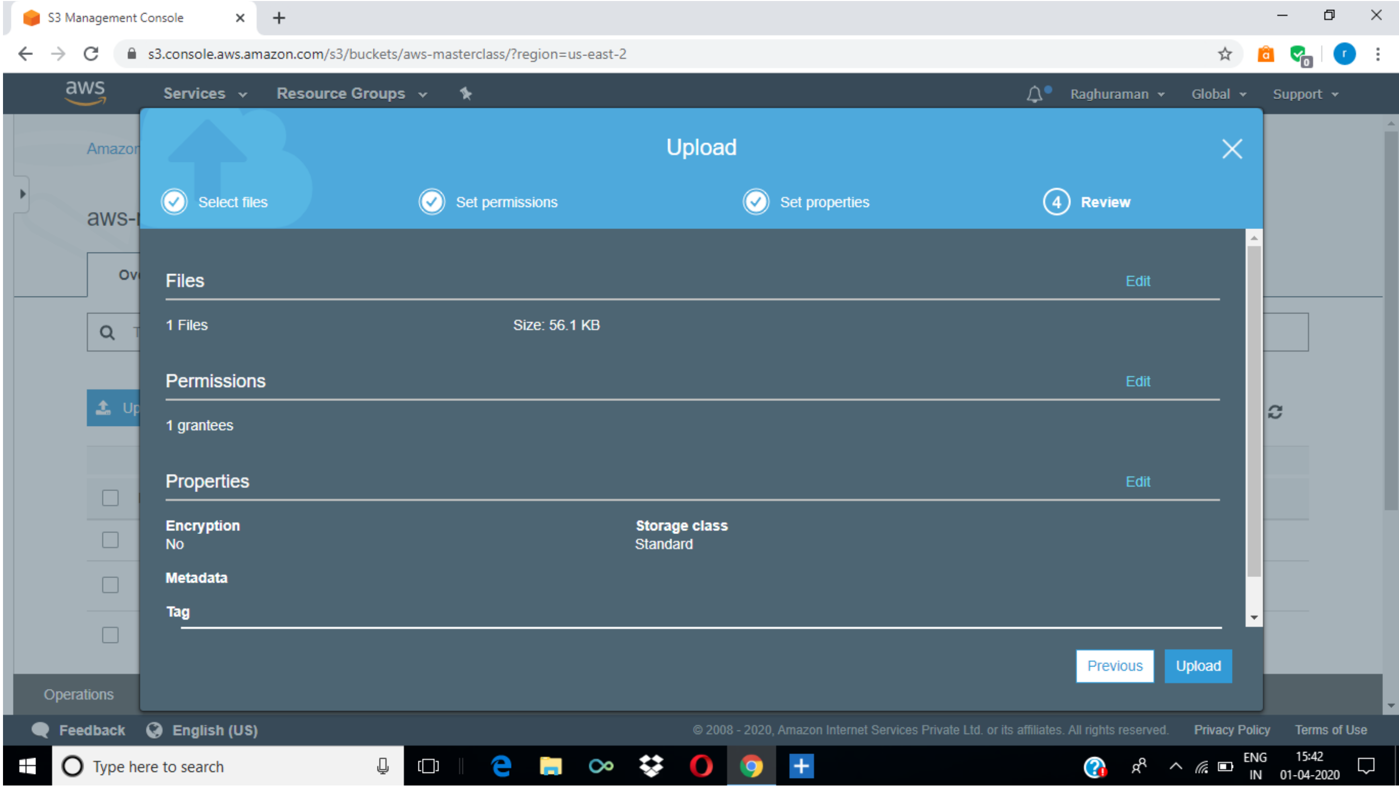
Successfully created bucket aws-masterclass
To upload files and folders, or to configure additional bucket settings such as Bucket Versioning, tags, and default encryption, choose **Go to bucket details**.

Amazon S3

Buckets (1)

	Name	Region	Access	Bucket created
○	aws-masterclass	US East (Ohio) us-east-2	Objects can be public	2020-04-01T11:10:29.000Z

2. Uploading an Object



Files

1 Files Size: 56.1 KB

Permissions

1 grantees

Properties

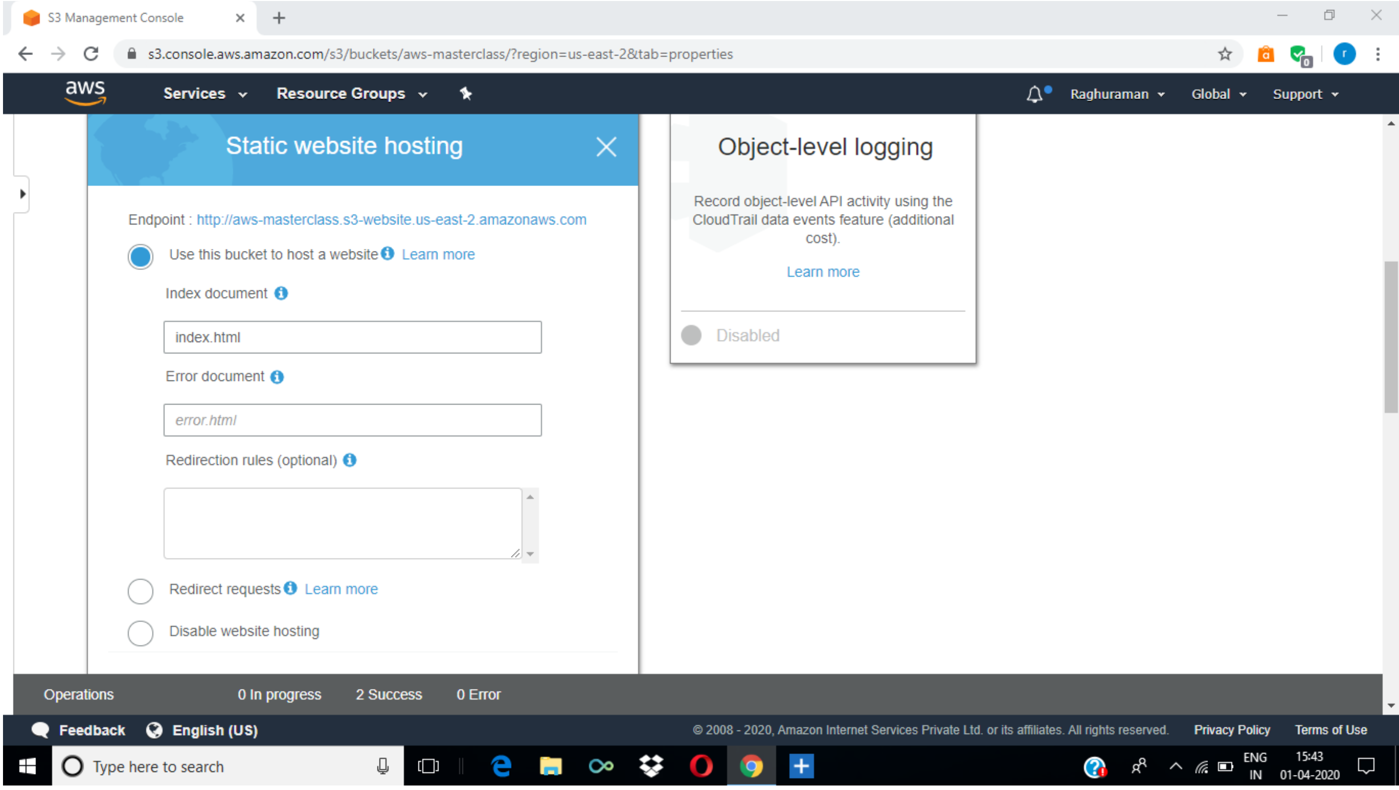
Encryption: No

Storage class: Standard

Metadata

Tag

3. Enabling Static Website



S3 Management Console
s3.console.aws.amazon.com/s3/buckets/aws-masterclass/?region=us-east-2&tab=properties
aws
Services
Resource Groups
Raghuraman
Global
Support
Learn more
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
ENG IN
15:43
01-04-2020

Static website hosting

Endpoint : http://aws-masterclass.s3-website.us-east-2.amazonaws.com

● Use this bucket to host a website

Index document: index.html

Error document: error.html

Redirection rules (optional)

○ Redirect requests

○ Disable website hosting

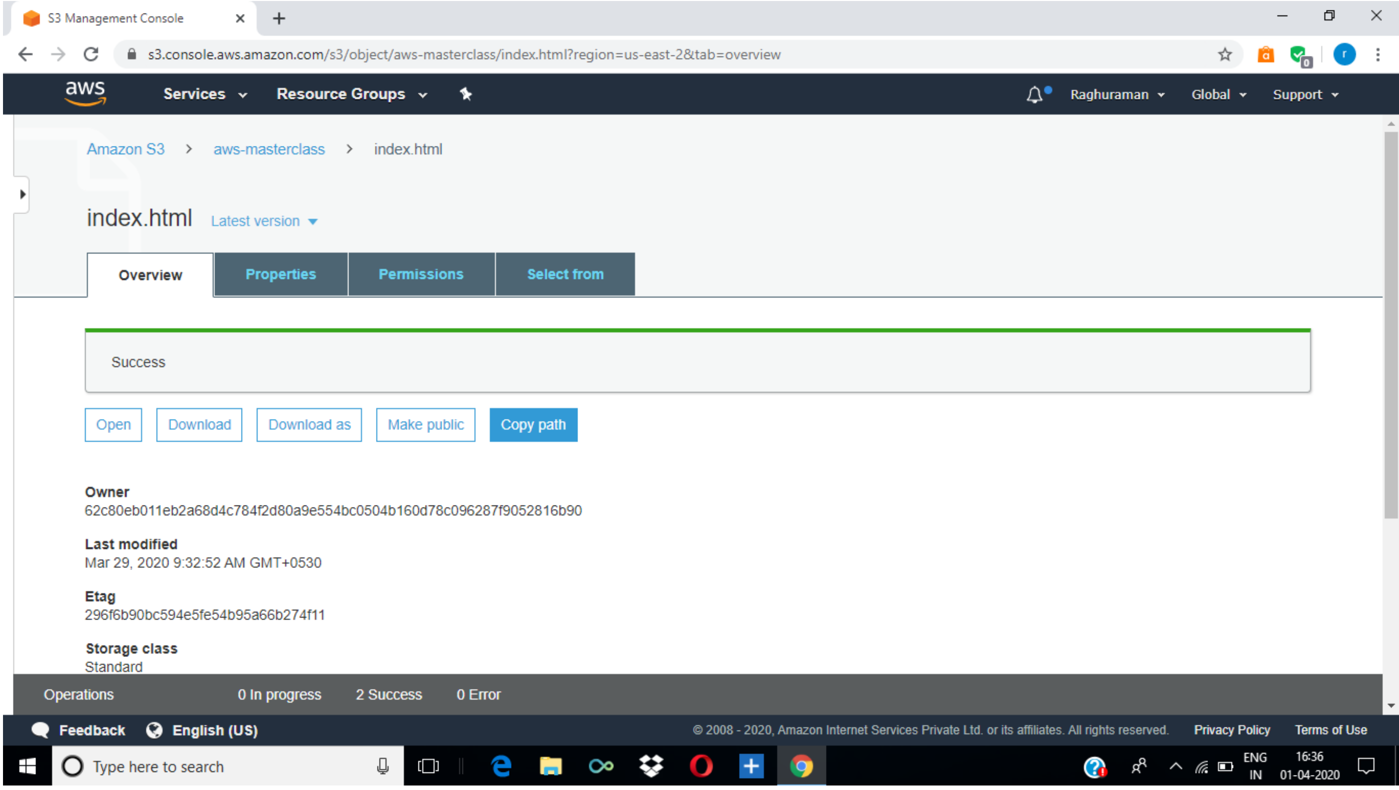
Object-level logging

Record object-level API activity using the CloudTrail data events feature (additional cost).

● Disabled

Operations 0 In progress 2 Success 0 Error

4. Making the Object Public



Success

Owner
62c80eb011eb2a68d4c784f2d80a9e554bc0504b160d78c096287f9052816b90

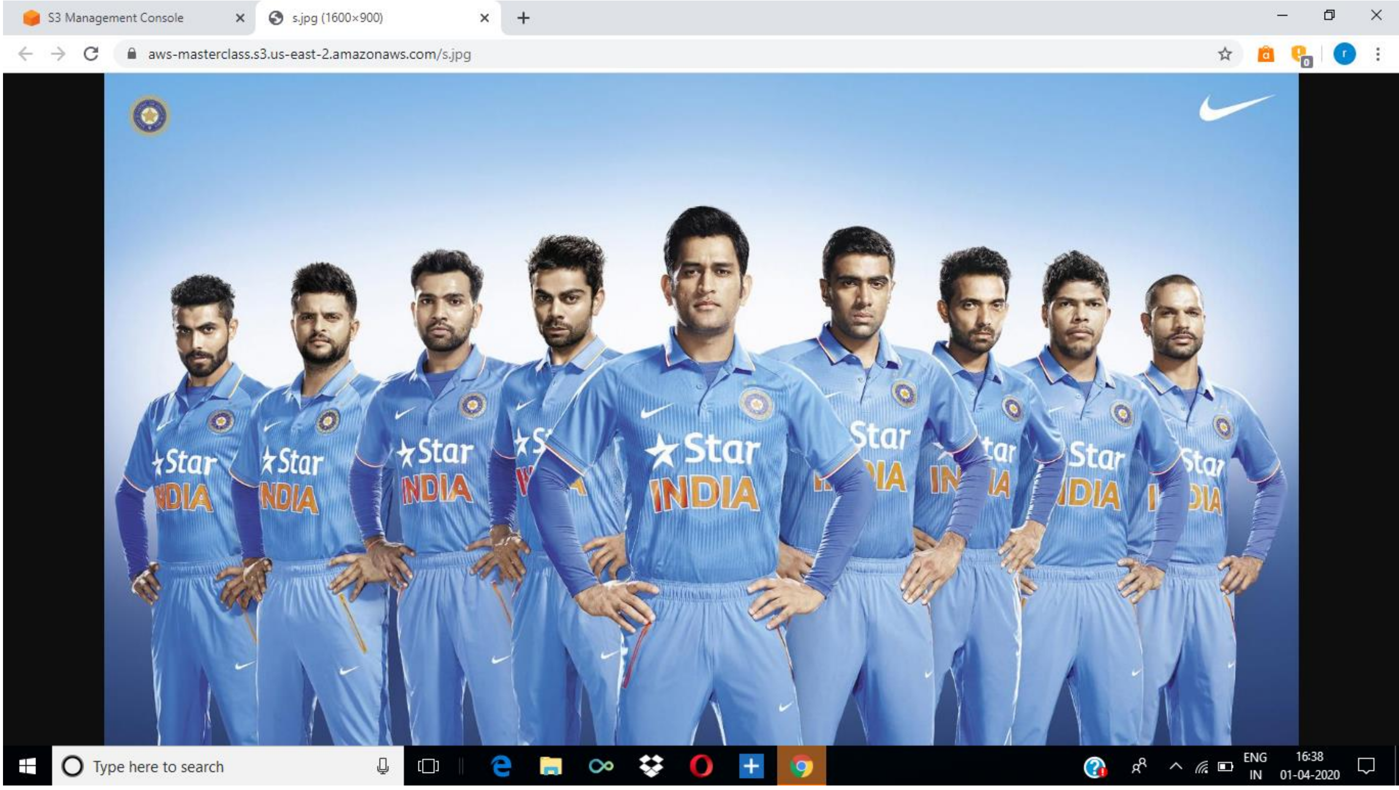
Last modified
Mar 29, 2020 9:32:52 AM GMT+0530

Etag
296f6b90bc594e5fe54b95a66b274f11

Storage class
Standard

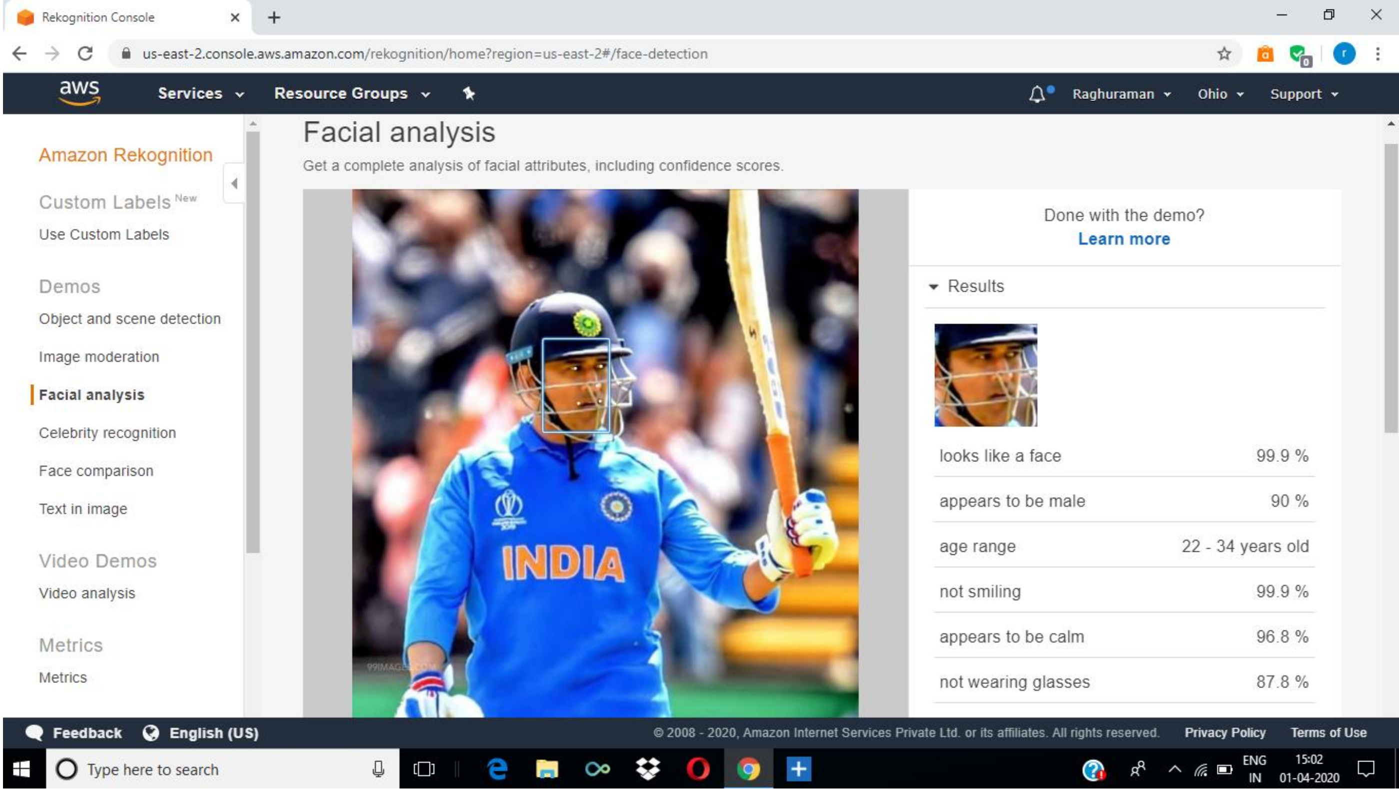
Operations 0 In progress 2 Success 0 Error

5. Checking the S3 link on the Browser



Rekognition

1. Face Detect



Facial analysis

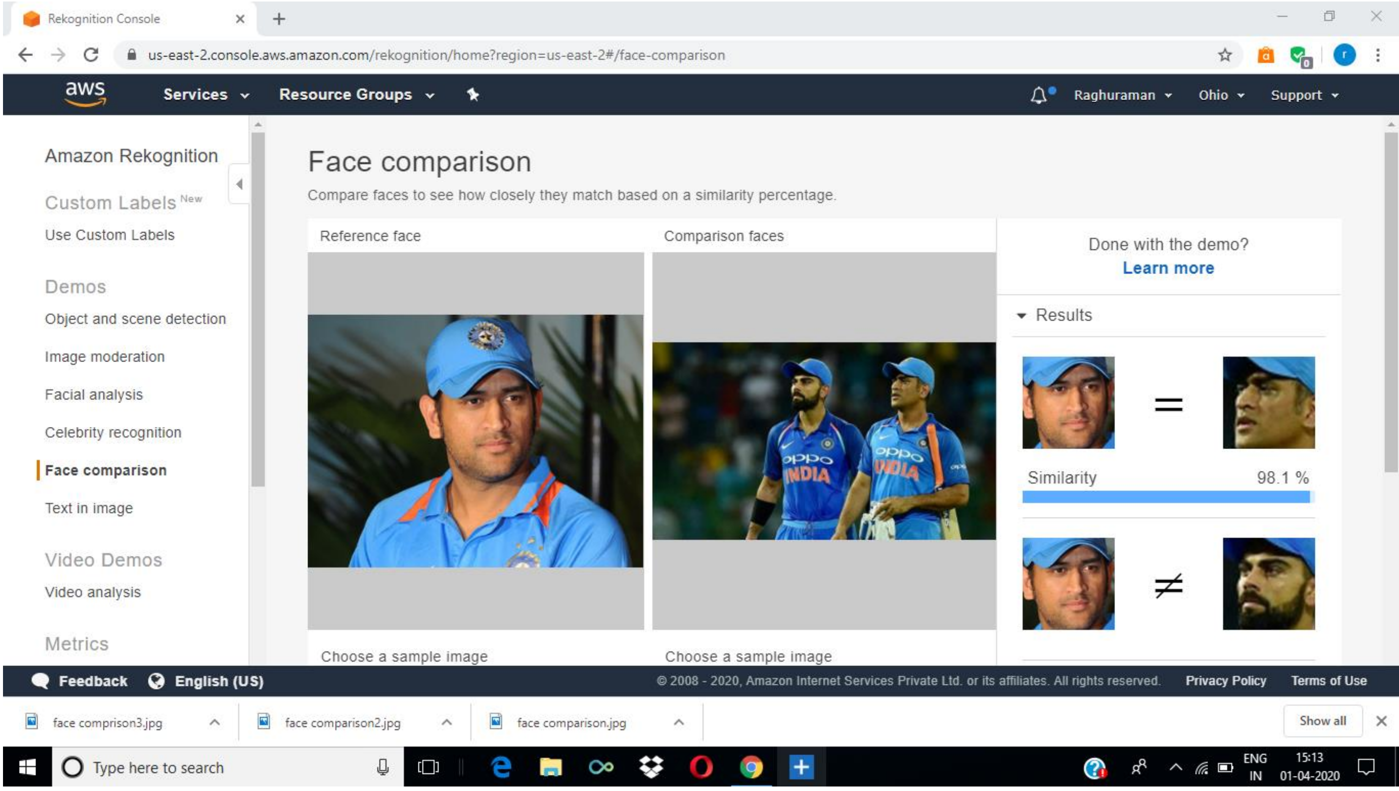
Get a complete analysis of facial attributes, including confidence scores.

Done with the demo?
Learn more

Results

looks like a face	99.9 %
appears to be male	90 %
age range	22 - 34 years old
not smiling	99.9 %
appears to be calm	96.8 %
not wearing glasses	87.8 %

2. Face Compare



Face comparison

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

Reference face

Comparison faces

Choose a sample image

Choose a sample image

Done with the demo?
Learn more

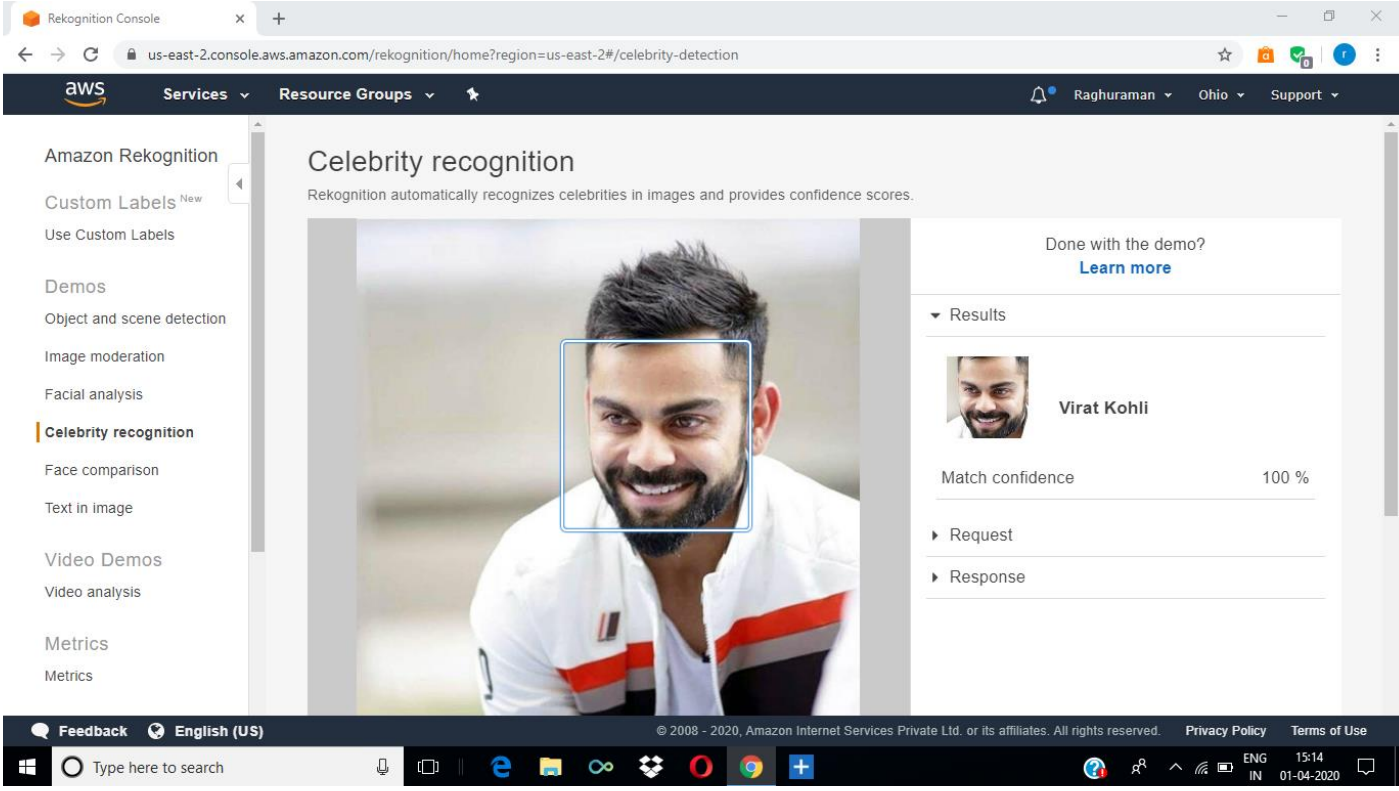
Results

=

Similarity 98.1 %

≠

3. Celebrity Recognition



Celebrity recognition

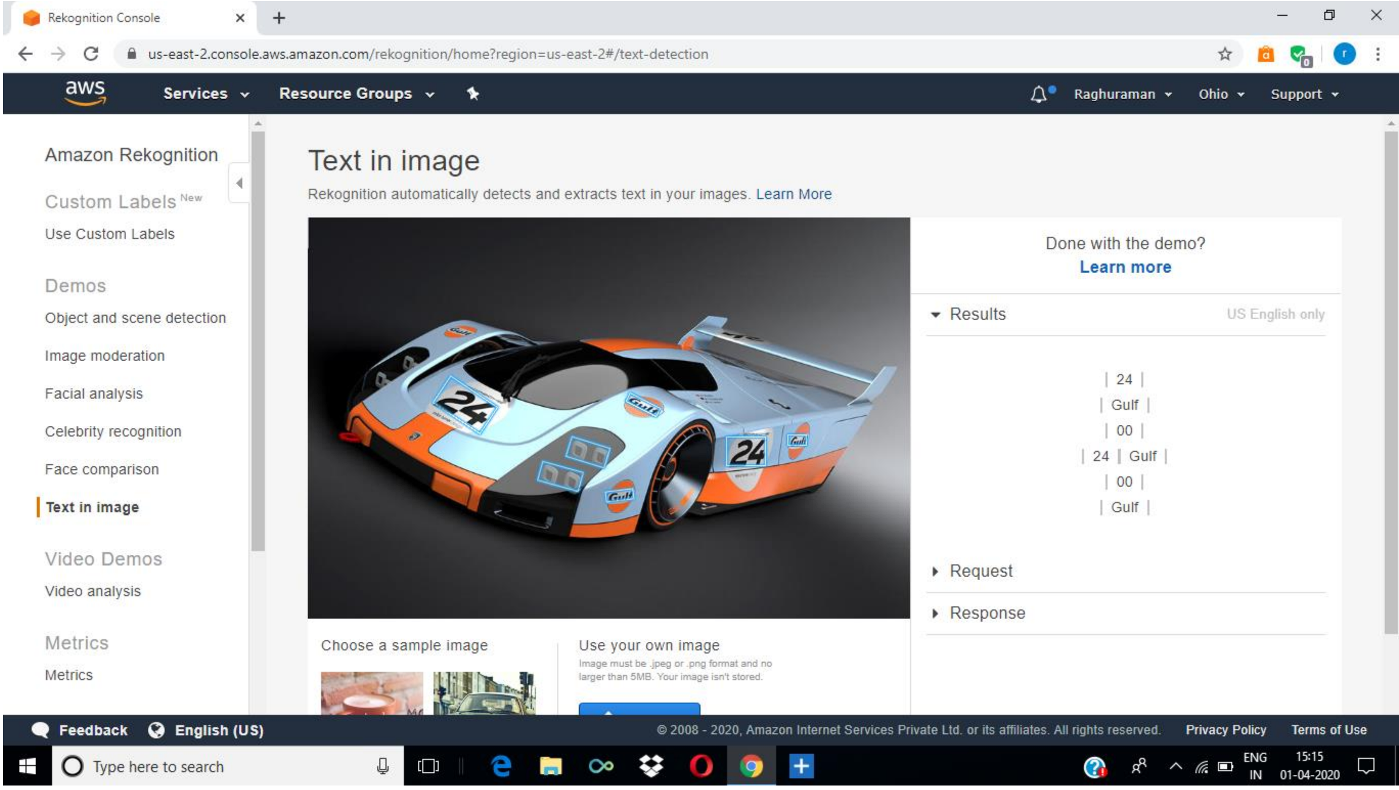
Rekognition automatically recognizes celebrities in images and provides confidence scores.

Results

Virat Kohli

Match confidence 100 %

4. Text in Image



Text in image

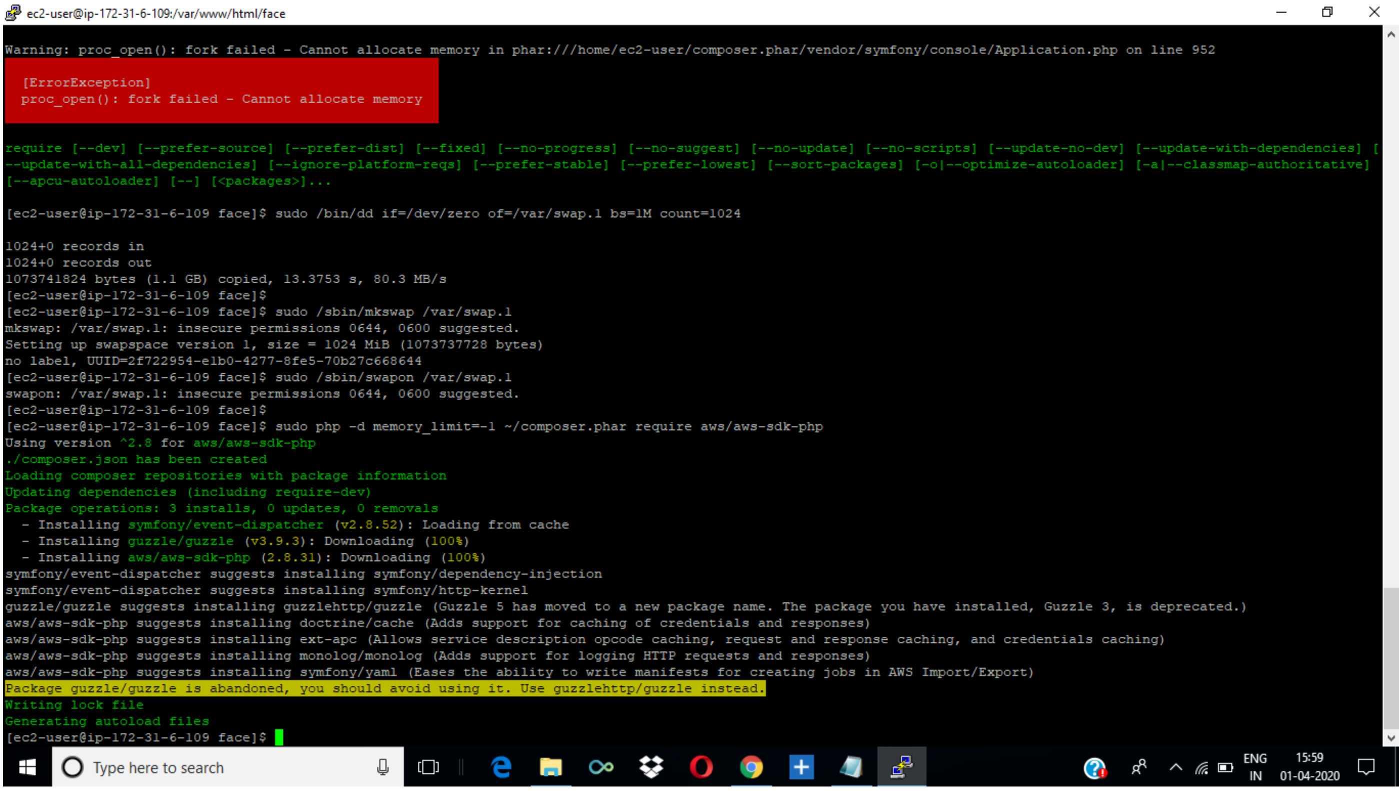
Rekognition automatically detects and extracts text in your images. Learn More

Results US English only

| 24 |
| Gulf |
| 00 |
| 24 | Gulf |
| 00 |
| Gulf |

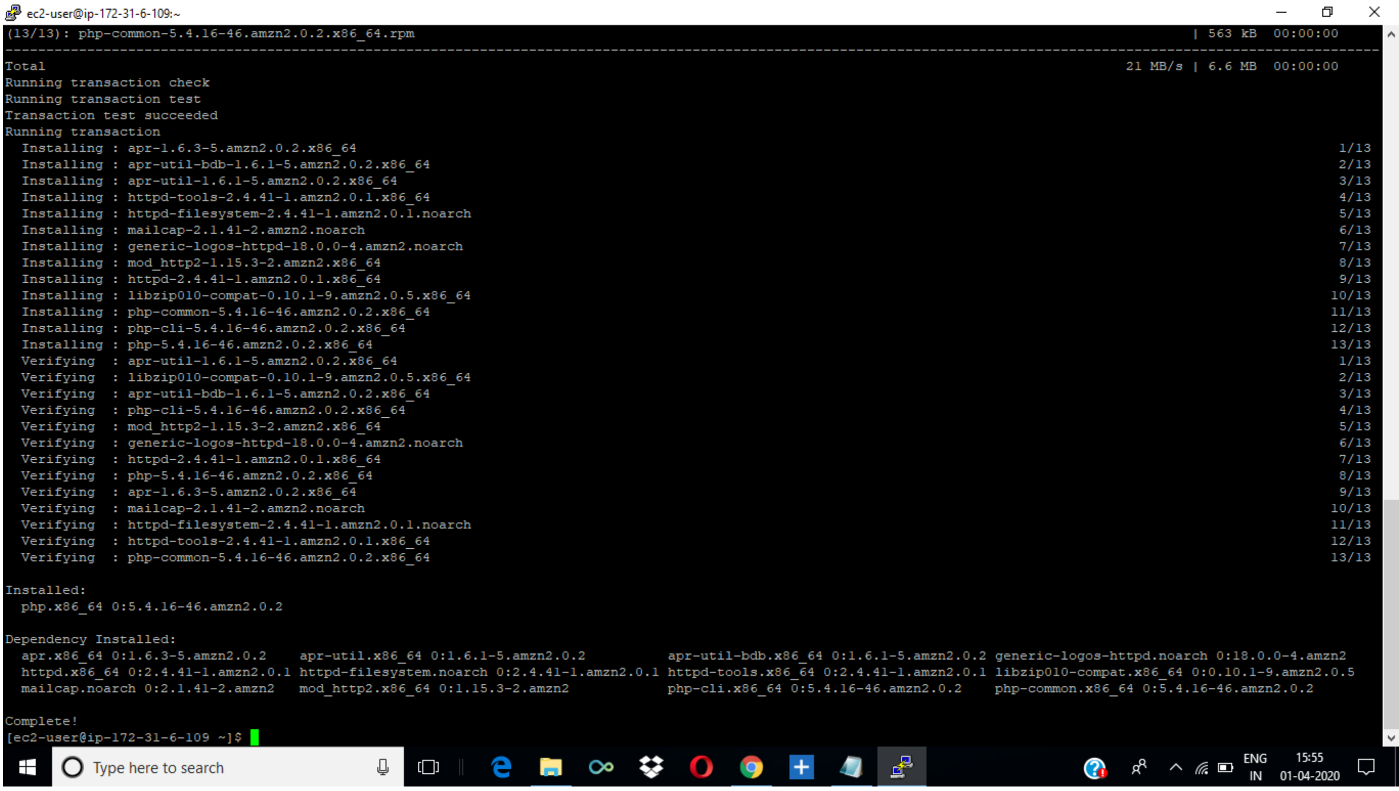
EC2 and S3

1. Installing aws-sdk



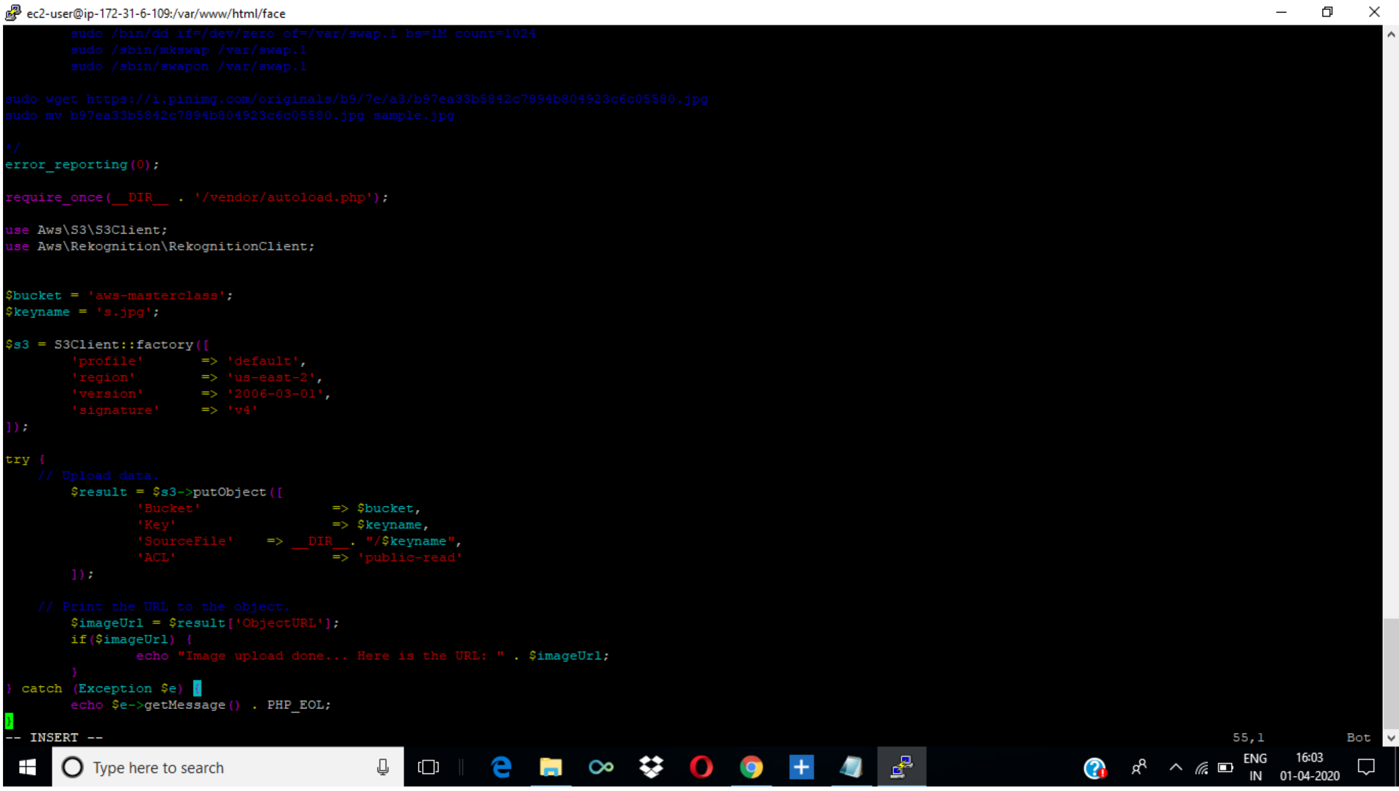
```
Warning: proc_open(): fork failed - Cannot allocate memory in phar:///home/ec2-user/composer.phar/vendor/symfony/console/Application.php on line 952

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

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

[ec2-user@ip-172-31-6-109 face]$ sudo /bin/dd if=/dev/zero of=/var/swap.1 bs=1M count=1024

1024+0 records in
1024+0 records out
1073741824 bytes (1.1 GB) copied, 13.3753 s, 80.3 MB/s
[ec2-user@ip-172-31-6-109 face]$
[ec2-user@ip-172-31-6-109 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=2f722954-e1b0-4277-8fe5-70b27c668644
[ec2-user@ip-172-31-6-109 face]$ sudo /sbin/swapon /var/swap.1
swapon: /var/swap.1: insecure permissions 0644, 0600 suggested.
[ec2-user@ip-172-31-6-109 face]$
[ec2-user@ip-172-31-6-109 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): 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-172-31-6-109 face]$
```

2. Installing php



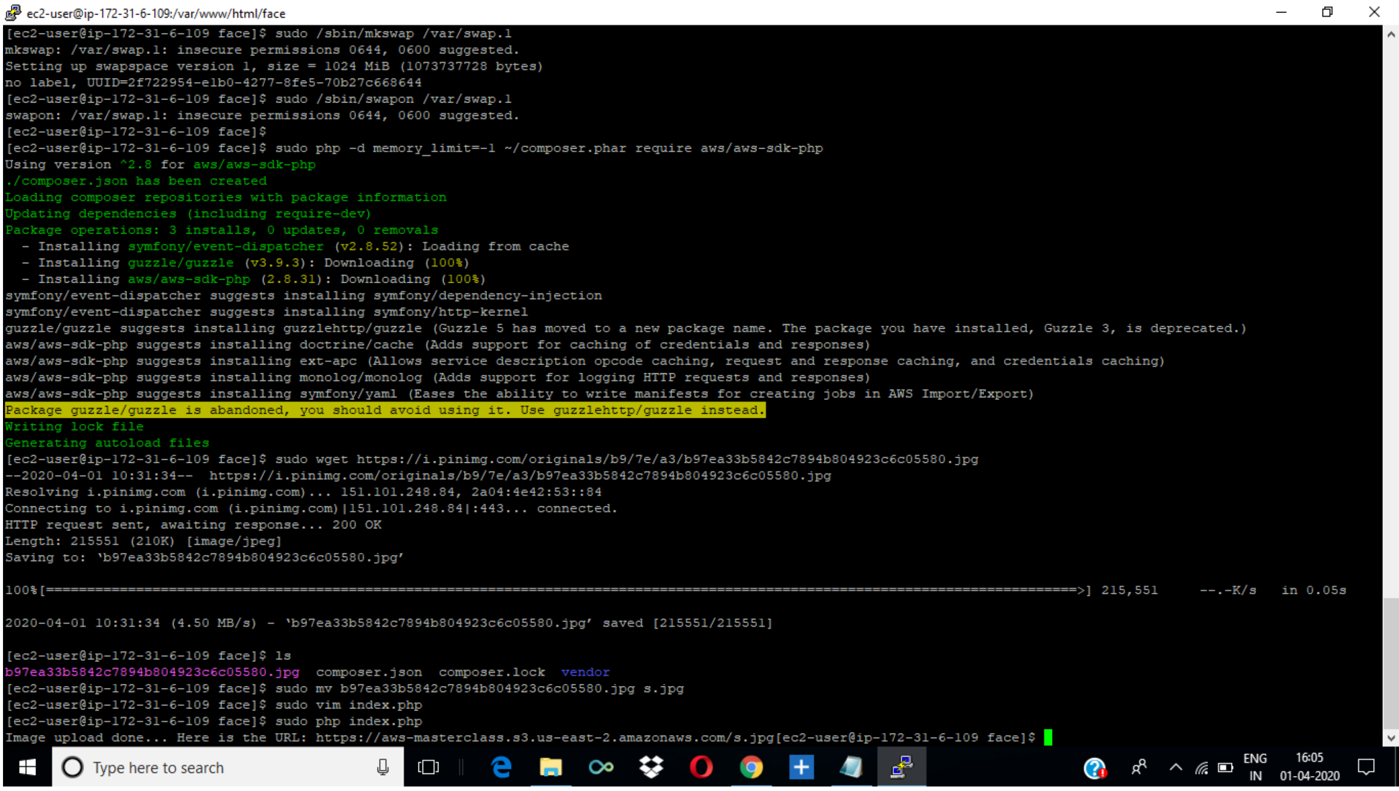
```
(13/13): php-common-5.4.16-46.amzn2.0.2.x86_64.rpm                                                                    | 563 kB  00:00:00
--------------------------------------------------------------------------------------------------------------------------------------------
Total                                                                                                          21 MB/s | 6.6 MB  00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : apr-1.6.3-5.amzn2.0.2.x86_64                                                                                            1/13
  Installing : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64                                                                                   2/13
  Installing : apr-util-1.6.1-5.amzn2.0.2.x86_64                                                                                       3/13
  Installing : httpd-tools-2.4.41-1.amzn2.0.1.x86_64                                                                                   4/13
  Installing : httpd-filesystem-2.4.41-1.amzn2.0.1.noarch                                                                              5/13
  Installing : mailcap-2.1.41-2.amzn2.noarch                                                                                           6/13
  Installing : generic-logos-httpd-18.0.0-4.amzn2.noarch                                                                               7/13
  Installing : mod_http2-1.15.3-2.amzn2.x86_64                                                                                         8/13
  Installing : httpd-2.4.41-1.amzn2.0.1.x86_64                                                                                         9/13
  Installing : libzip010-compat-0.10.1-9.amzn2.0.5.x86_64                                                                             10/13
  Installing : php-common-5.4.16-46.amzn2.0.2.x86_64                                                                                  11/13
  Installing : php-cli-5.4.16-46.amzn2.0.2.x86_64                                                                                     12/13
  Installing : php-5.4.16-46.amzn2.0.2.x86_64                                                                                         13/13
  Verifying  : apr-util-1.6.1-5.amzn2.0.2.x86_64                                                                                       1/13
  Verifying  : libzip010-compat-0.10.1-9.amzn2.0.5.x86_64                                                                              2/13
  Verifying  : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64                                                                                   3/13
  Verifying  : php-cli-5.4.16-46.amzn2.0.2.x86_64                                                                                      4/13
  Verifying  : mod_http2-1.15.3-2.amzn2.x86_64                                                                                         5/13
  Verifying  : generic-logos-httpd-18.0.0-4.amzn2.noarch                                                                               6/13
  Verifying  : httpd-2.4.41-1.amzn2.0.1.x86_64                                                                                         7/13
  Verifying  : php-5.4.16-46.amzn2.0.2.x86_64                                                                                          8/13
  Verifying  : apr-1.6.3-5.amzn2.0.2.x86_64                                                                                            9/13
  Verifying  : mailcap-2.1.41-2.amzn2.noarch                                                                                          10/13
  Verifying  : httpd-filesystem-2.4.41-1.amzn2.0.1.noarch                                                                             11/13
  Verifying  : httpd-tools-2.4.41-1.amzn2.0.1.x86_64                                                                                  12/13
  Verifying  : php-common-5.4.16-46.amzn2.0.2.x86_64                                                                                  13/13

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

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.x86_64 0:2.4.41-1.amzn2.0.1 httpd-filesystem.noarch 0:2.4.41-1.amzn2.0.1 httpd-tools.x86_64 0:2.4.41-1.amzn2.0.1 libzip010-compat.x86_64 0:0.10.1-9.amzn2.0.5
  mailcap.noarch 0:2.1.41-2.amzn2   mod_http2.x86_64 0:1.15.3-2.amzn2            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-172-31-6-109 ~]$
```

3. Index.php file code



```
        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-masterclass';
$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'
        ]);

    // 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 --                                                                                    55,1          Bot
```

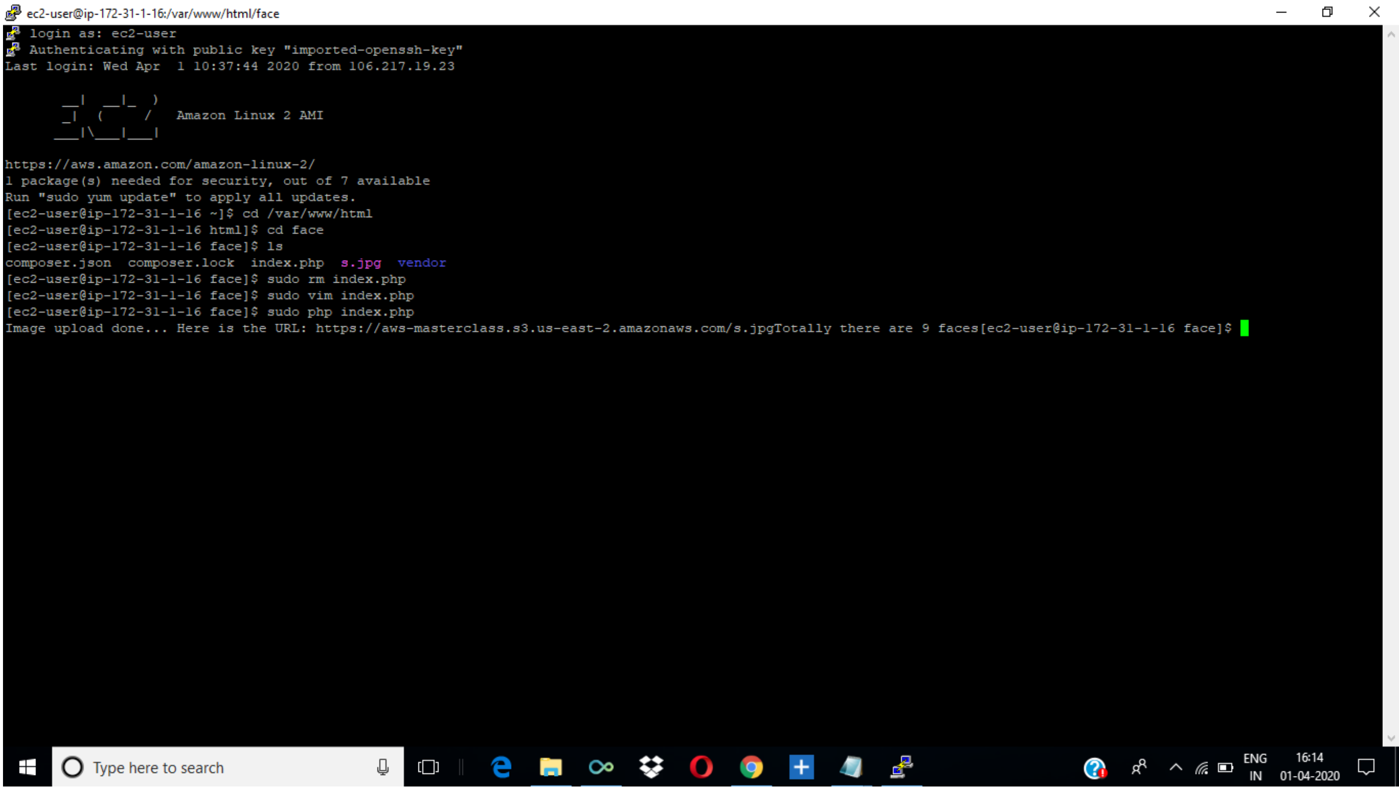
4. Upload Success Screenshot



```
[ec2-user@ip-172-31-6-109 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=2f722954-e1b0-4277-8fe5-70b27c668644
[ec2-user@ip-172-31-6-109 face]$ sudo /sbin/swapon /var/swap.1
swapon: /var/swap.1: insecure permissions 0644, 0600 suggested.
[ec2-user@ip-172-31-6-109 face]$
[ec2-user@ip-172-31-6-109 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): 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-172-31-6-109 face]$ sudo wget https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
--2020-04-01 10:31:34--  https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
Resolving i.pinimg.com (i.pinimg.com)... 151.101.248.84, 2a04:4e42:53::84
Connecting to i.pinimg.com (i.pinimg.com)|151.101.248.84|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 215551 (210K) [image/jpeg]
Saving to: ‘b97ea33b5842c7894b804923c6c05580.jpg’

100%[=================================================================================================================>] 215,551     --.-K/s   in 0.05s

2020-04-01 10:31:34 (4.50 MB/s) - ‘b97ea33b5842c7894b804923c6c05580.jpg’ saved [215551/215551]

[ec2-user@ip-172-31-6-109 face]$ ls
b97ea33b5842c7894b804923c6c05580.jpg  composer.json  composer.lock  vendor
[ec2-user@ip-172-31-6-109 face]$ sudo mv b97ea33b5842c7894b804923c6c05580.jpg s.jpg
[ec2-user@ip-172-31-6-109 face]$ sudo vim index.php
[ec2-user@ip-172-31-6-109 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-masterclass.s3.us-east-2.amazonaws.com/s.jpg[ec2-user@ip-172-31-6-109 face]$
```

EC2 and Rekognition

1. Face Detect Success Screenshot



```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Wed Apr  1 10:37:44 2020 from 106.217.19.23

       __|  __|_  )
       _|  (     /   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-172-31-1-16 ~]$ cd /var/www/html
[ec2-user@ip-172-31-1-16 html]$ cd face
[ec2-user@ip-172-31-1-16 face]$ ls
composer.json  composer.lock  index.php  s.jpg  vendor
[ec2-user@ip-172-31-1-16 face]$ sudo rm index.php
[ec2-user@ip-172-31-1-16 face]$ sudo vim index.php
[ec2-user@ip-172-31-1-16 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-masterclass.s3.us-east-2.amazonaws.com/s.jpgTotally there are 9 faces[ec2-user@ip-172-31-1-16 face]$
```

Thank You…