

Dashboards



Root user sign in ⓘ

Email: chaitanyapawar51@gmail.com

Password

[Forgot password?](#)

[Sign in](#)

[Sign in to a different account](#)

[Create a new AWS account](#)

aws RE:INFORCE

June 30 – July 1, 2020 • Houston, TX

Two days and hundreds of sessions focused on cloud security, identity, and compliance.

[Register Now](#)

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 have entered into a separate agreement with Amazon Web Services or an AWS Value Added Reseller to purchase these products and services. The AWS Customer Agreement was updated on March 31, 2017. For more information about these updates, see [Recent Changes](#).

1.AWS login Screen with username

The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, 'Services', 'Resource Groups', and user information (Chaitanya Sunil Pawar, Ohio, Support). Below this is a blue banner announcing the 'New EC2 Experience'. The left-hand navigation pane is expanded to 'INSTANCES', showing options like 'Instances', 'Instance Types', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', 'Dedicated Hosts', and 'Capacity Reservations'. The main content area is titled 'EC2' and displays a 'Resources' section with a table of EC2 resources in the US East (Ohio) Region. The table lists: Running instances (1), Elastic IPs (0), Dedicated Hosts (0), Snapshots (0), Volumes (1), Load balancers (0), Key pairs (2), Security groups (10), and Placement groups (0). Below the table is a 'Launch instance' button. On the right, there's an 'Account attributes' panel showing supported platforms (VPC) and default VPC (vpc-938055f8). Below that is an 'Explore AWS' panel with a link to 'Optimize your EC2 cost and performance with Spot Instances'.

Resources	
Running instances	1
Elastic IPs	0
Dedicated Hosts	0
Snapshots	0
Volumes	1
Load balancers	0
Key pairs	2
Security groups	10
Placement groups	0

2.EC2 dashboard

The screenshot shows the Amazon S3 console interface. On the left, there's a navigation pane with 'Buckets' selected. The main area displays 'Buckets (1)' with a search bar and a table containing one bucket: 'aws-web-lax' in the 'us-east-1' region, created on '2020-04-02T05:06:42.000Z'. Action buttons like 'Copy ARN', 'Empty', 'Delete', and 'Create bucket' are visible at the top right of the bucket list.

3.S3 dashboard

The screenshot displays the Amazon Rekognition dashboard. The header includes the AWS logo and navigation links. The main content area features a large banner with the text 'Amazon Rekognition' and 'Deep learning-based visual analysis service'. Below the banner, there are three columns of information:

- Easily Integrate Powerful Visual Analysis into Your App:** Explains how Rekognition simplifies building visual analysis applications.
- Continuously Learning:** States that Rekognition uses deep learning technology to analyze billions of images and videos daily, improving its capabilities over time.
- Integrated with AWS Services:** Highlights the seamless integration with Amazon S3 and AWS Lambda for scalable and reliable visual analysis.

 The left sidebar lists various features like 'Custom Labels', 'Demos', and 'Metrics'.

4.Rekognition dashboard

EC2

aws

Services

Resource Groups

Chaitanya Sunil PawarOhioSupport

1. Choose AMI2. Choose Instance Type3. Configure Instance4. Add Storage5. Add Tags6. Configure Security Group7. Review

Step 1: Choose an Amazon Machine Image (AMI)

Cancel and Exit

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

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: ebsVirtualization type: hvmENA Enabled: Yes

64-bit (x86)64-bit (Arm)

Select

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: ebsVirtualization type: hvmENA Enabled: Yes

64-bit (x86)

Select

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: ebsVirtualization type: hvmENA Enabled: Yes

64-bit (x86)64-bit (Arm)

Select

FeedbackEnglish (US)

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

1.Choosing an AMI

aws

Services

Resource Groups

Chaitanya Sunil PawarOhioSupport

1. Choose AMI2. Choose Instance Type3. Configure Instance4. Add Storage5. Add Tags6. Configure Security Group7. 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 typesCurrent generationShow/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

Cancel

Previous

Review and Launch

Next: Configure Instance Details

FeedbackEnglish (US)

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

2.Choosing an instance type

Services

Resource Groups

Chaitanya Sunil Pawar

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-0f54682056aaa4c20	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](#)

3.Adding Storage

Services

Resource Groups

Chaitanya Sunil Pawar

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:

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

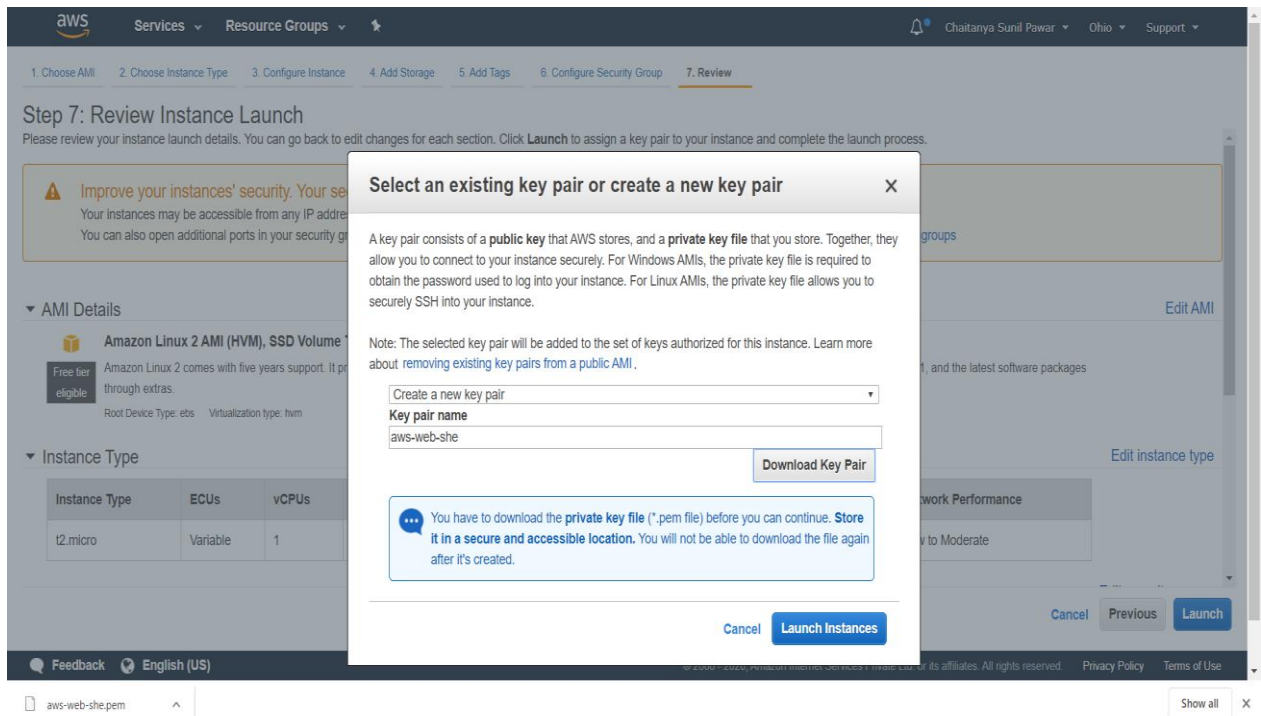
Review and Launch

Feedback

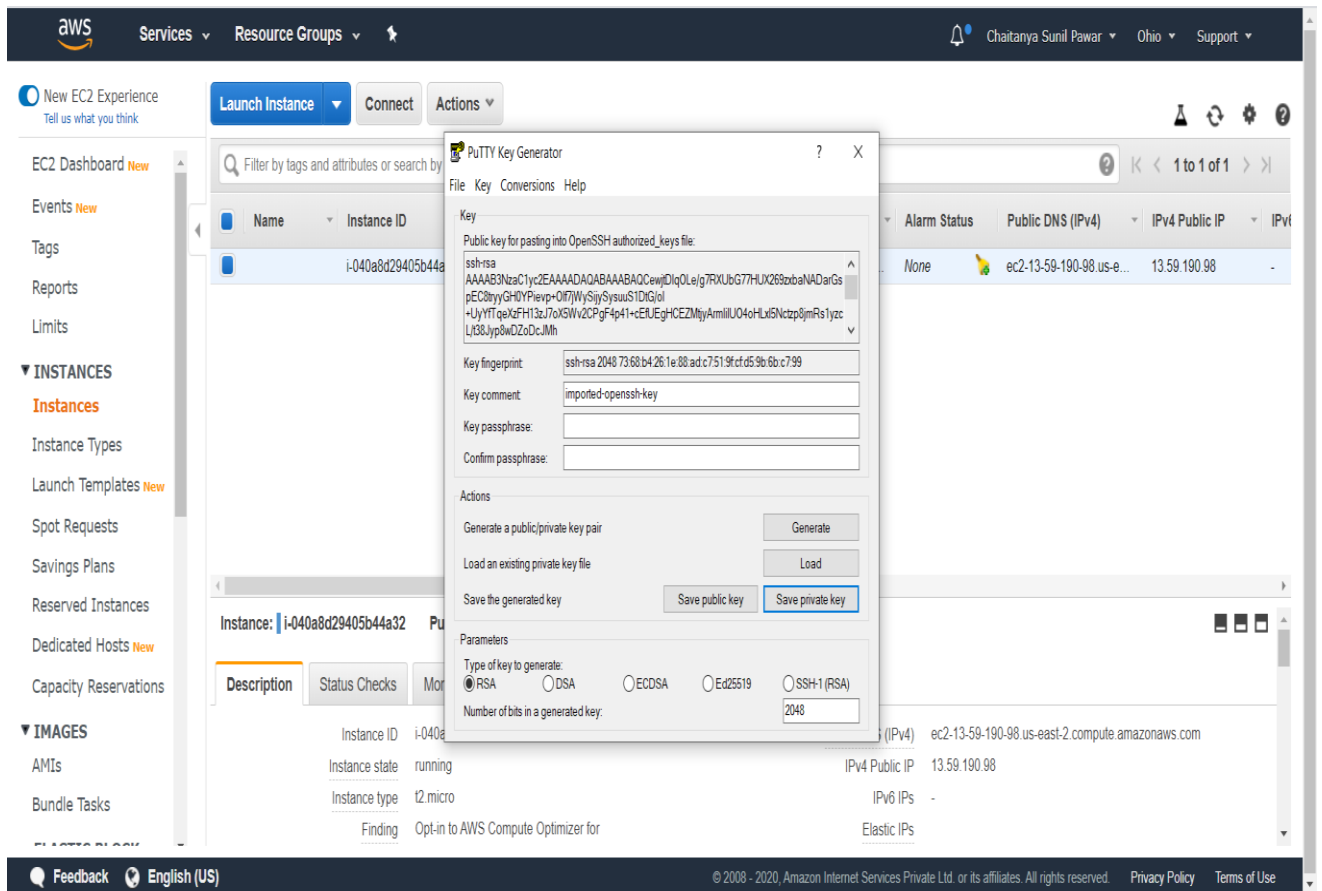
English (US)

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

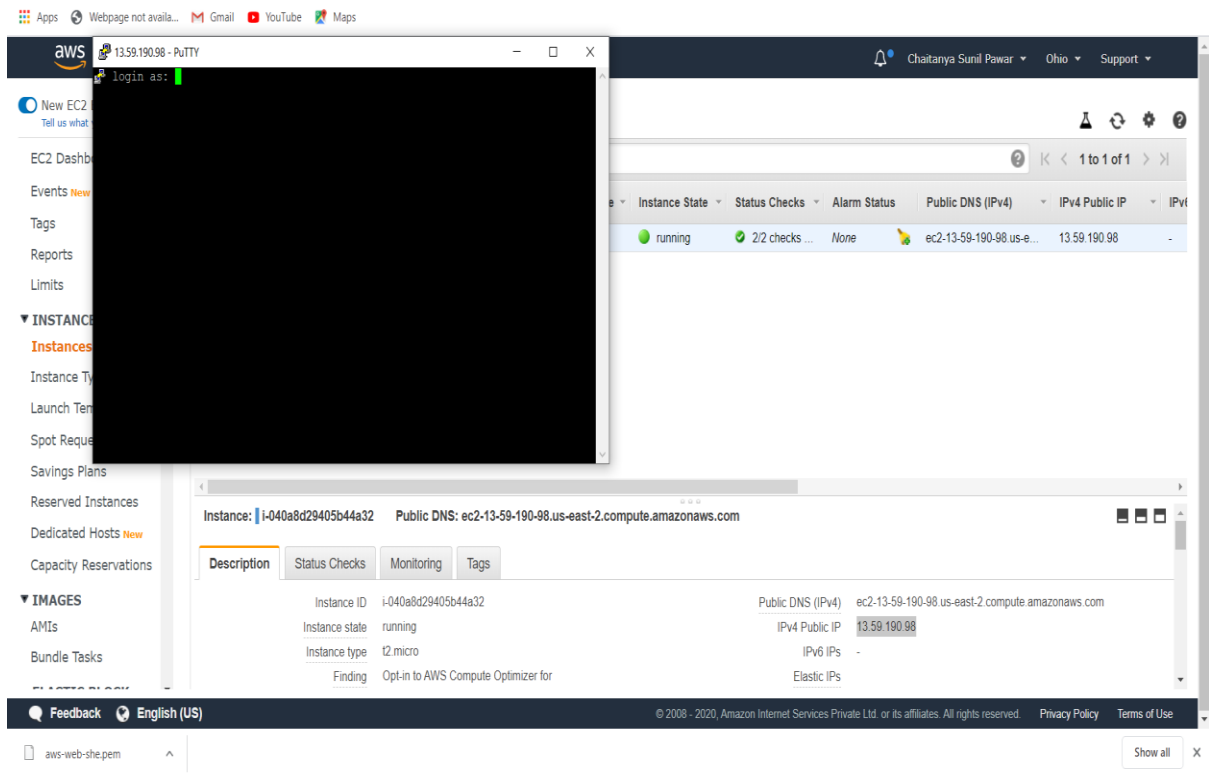
4.configur security group



5.key pair download

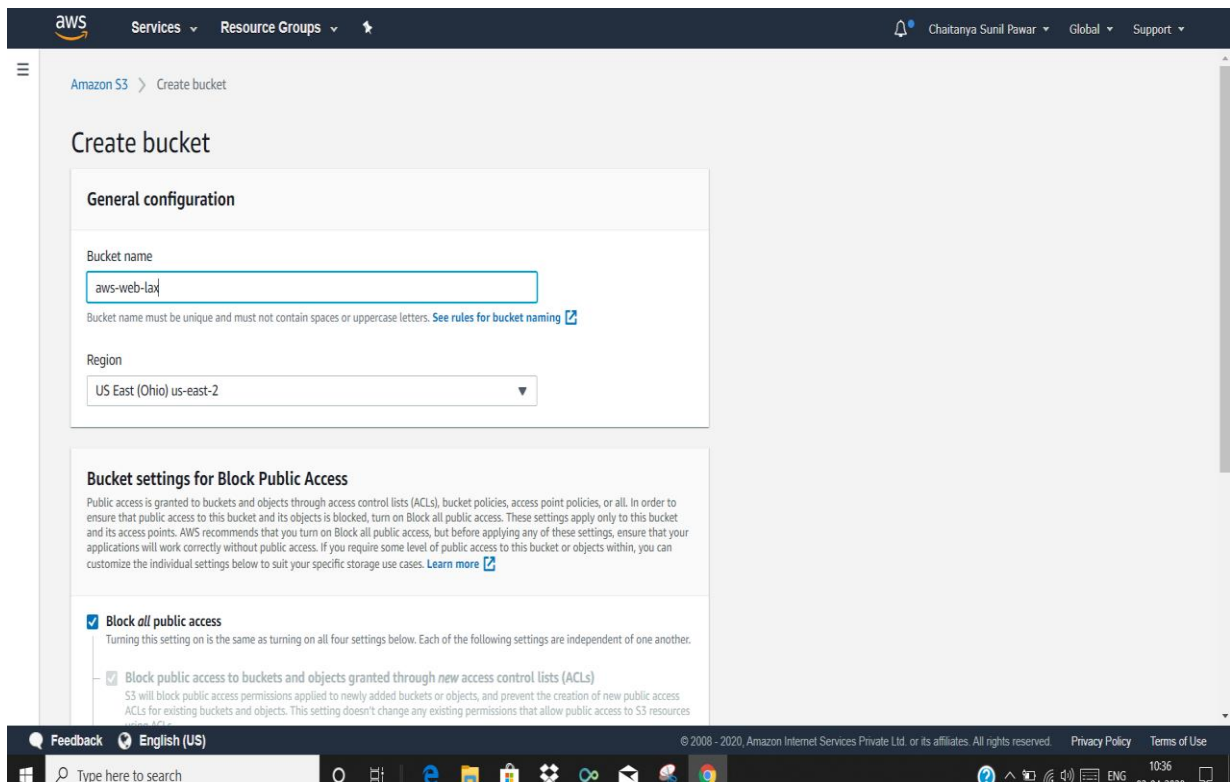


6.PUTTYgen conversion from pem to ppk

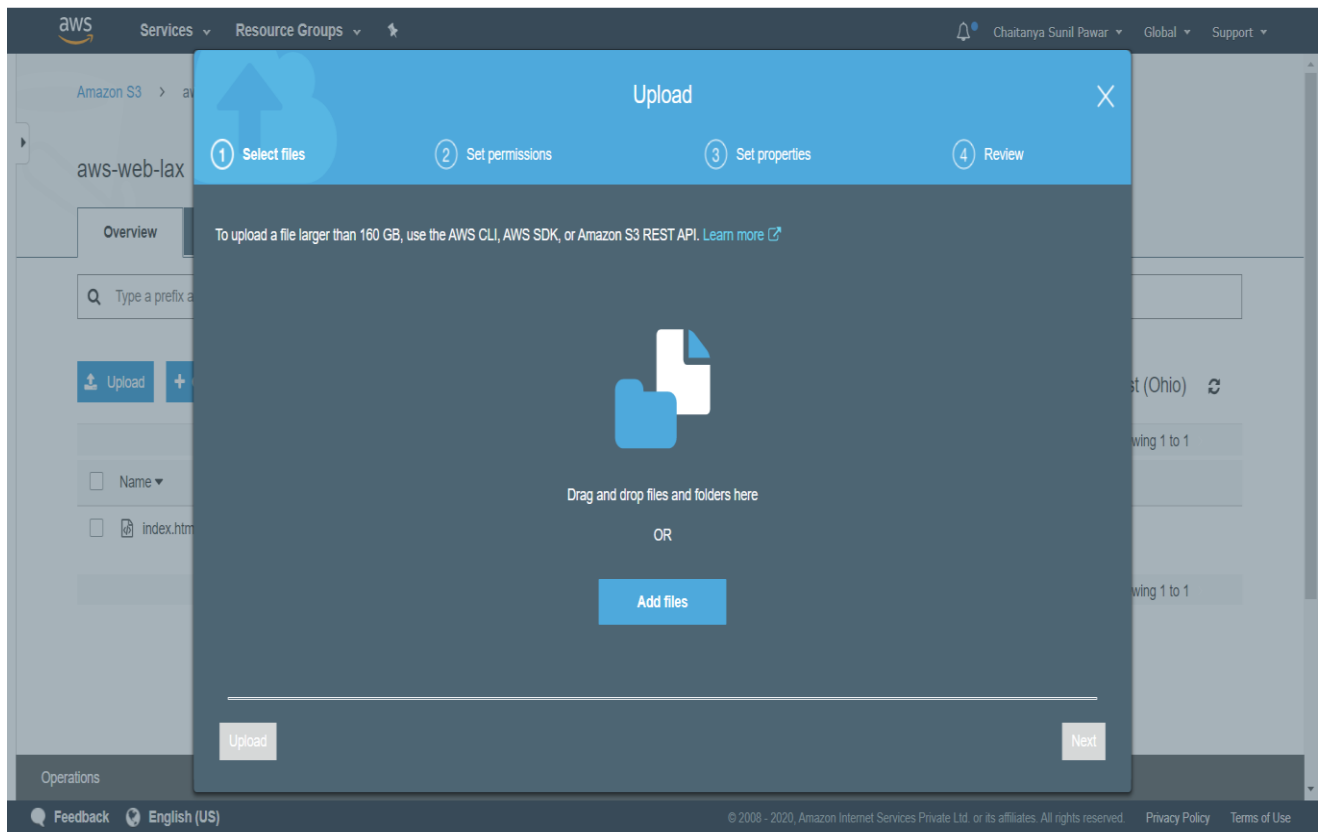


7. Logged in EC2 black screen

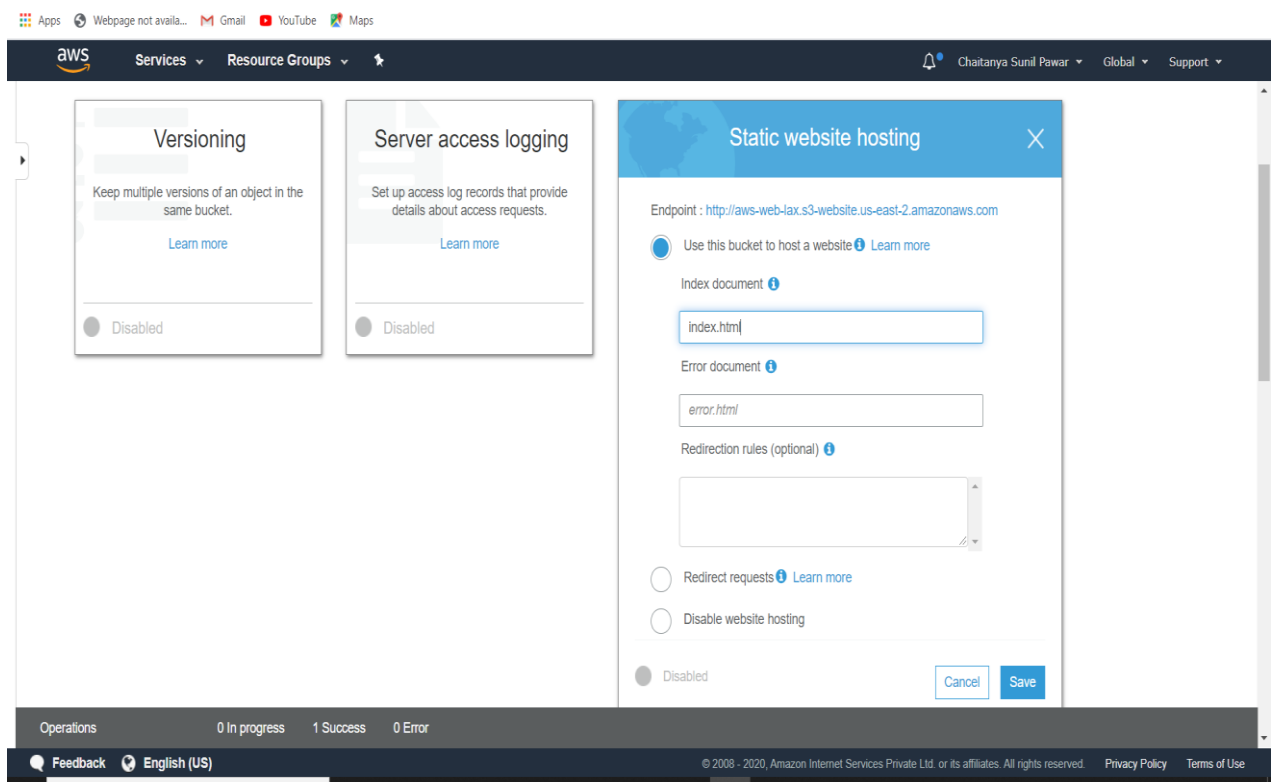
S3



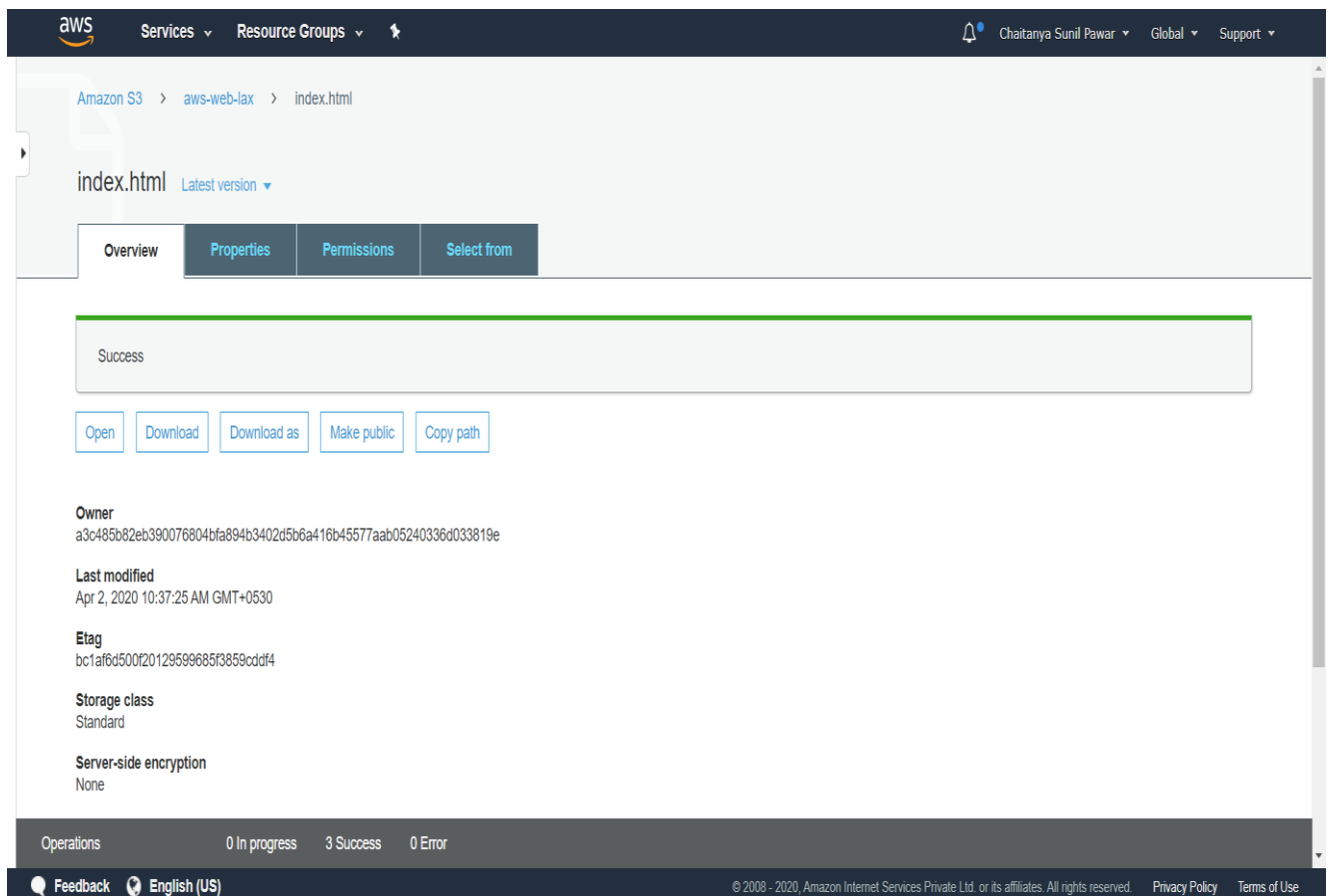
1. Creating a bucket



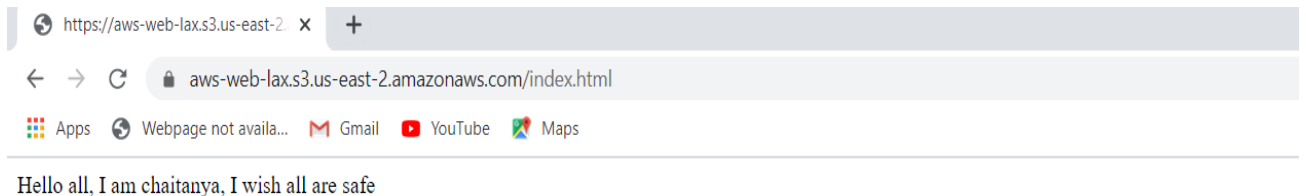
2. Uploading a Object



3. Enabling Static website



4. Making the object Public



5. Checking S3 link in browser

Rekognition

aws

Services

Resource Groups

Chaitanya Sunil Pawar

Ohio

Support

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

Additional Resources

Getting started guide

Facial analysis

Get a complete analysis of facial attributes, including 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.

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](#)

1.Face Detect

aws

Services

Resource Groups

Chaitanya Sunil Pawar

Ohio

Support

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

Additional Resources



Getting started guide

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		99.8 %
	≠	
	≠	

2.Face Compare

aws

Services

Resource Groups

Chaitanya Sunil Pawar

Ohio

Support

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

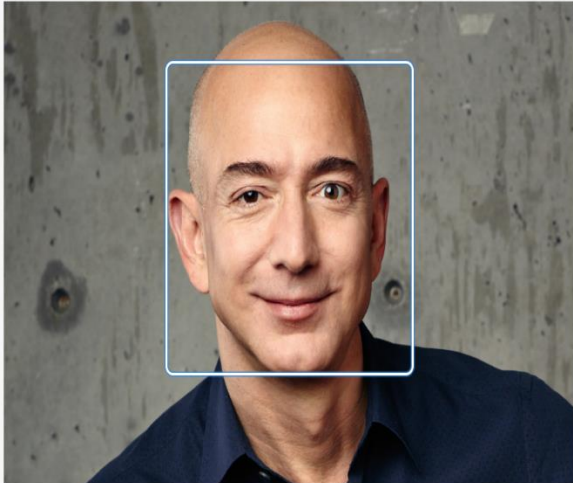
Metrics

Additional Resources



Getting started guide

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.

Done with the demo?

[Learn more](#)

▼ Results



Jeff Bezos

[Learn More](#)

Match confidence100 %

► Request

► Response

3.Celebrity rekognition

aws

Services

Resource Groups

Chaitanya Sunil Pawar

Ohio

Support

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

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 5MB. Your image isn't stored.

Done with the demo?

[Learn more](#)

▼ Results

US English only

IT'S

MONDAY

but keep

Smiling

► Request

► Response

4.Text in Image

EC2& S3

The screenshot displays the AWS Management Console interface. On the left, the navigation menu includes sections for EC2 Dashboard, INSTANCES, IMAGES, and ELASTIC BLOCK STORE. The main content area shows the details of an EC2 instance with ID `i-040a8d29405b44a32`. The instance is running on a `t2.micro` instance type. A terminal window is open, showing the execution of `composer require aws/aws-sdk-php`. The output indicates that the package and its dependencies were successfully installed.

```
Setting up swap space version 1, size = 1024 MiB (107373728 bytes)
no label, UUID=452b88c3-lac0-485c-8c40-694465c12c37
[ec2-user@ip-172-31-18-37 face]$ sudo /sbin/swapoff /var/swap.1
swapoff: /var/swap.1: insecure permissions 0644, 0600 suggested.
[ec2-user@ip-172-31-18-37 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-172-31-18-37 face]$
```

```
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-172-31-18-37 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, 13.9193 s, 77.1 MB/s
[ec2-user@ip-172-31-18-37 face]$ sudo /sbin/mkswap /var/swap.1
mkswap: /var/swap.1: insecure permissions 0644, 0600 suggested.
Setting up swap space version 1, size = 1024 MiB (107373728 bytes)
no label, UUID=452b88c3-lac0-485c-8c40-694465c12c37
[ec2-user@ip-172-31-18-37 face]$ sudo /sbin/swapoff /var/swap.1
swapoff: /var/swap.1: insecure permissions 0644, 0600 suggested.
[ec2-user@ip-172-31-18-37 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-172-31-18-37 face]$
```

1.Installing aws-sdk

aws Services Resource Groups

Chaitanya Sunil Pawar Ohio Support

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

Unlabeled

ec2-user@ip-172-31-18-37:/var/www/html/face

```
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-172-31-18-37 ~]$ sudo yum install php
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                               | 2.4 kB      00:00
Resolving Dependencies
--> Running transaction check
---> Package php.x86_64 0:5.4.16-46.amzn2.0.2 will be installed
--> Processing Dependency: php-cli(x86-64) = 5.4.16-46.amzn2.0.2 for package: php-5.4.16-46.amzn2.0.2.x86_64
--> Processing Dependency: php-common(x86-64) = 5.4.16-46.amzn2.0.2 for package: php-5.4.16-46.amzn2.0.2.x86_64
--> Processing Dependency: httpd-mmn = 20120211x8664 for package: php-5.4.16-46.amzn2.0.2.x86_64
--> Processing Dependency: httpd for package: php-5.4.16-46.amzn2.0.2.x86_64
--> Running transaction check
```

Instance: i-040a8d29405b44a32 Public DNS: ec2-13-59-190-98.us-east-2.compute.amazonaws.com

Description Status Checks Monitoring Tags

Instance ID i-040a8d29405b44a32 Public DNS (IPv4) ec2-13-59-190-98.us-east-2.compute.amazonaws.com

Instance state running IPv4 Public IP 13.59.190.98

Instance type t2.micro IPv6 IPs -

Finding Opt-in to AWS Compute Optimizer for recommendations. Elastic IPs

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 10:52 02-04-2020

```
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-172-31-18-37 ~]$ sudo yum install php
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                               | 2.4 kB      00:00
Resolving Dependencies
--> Running transaction check
---> Package php.x86_64 0:5.4.16-46.amzn2.0.2 will be installed
--> Processing Dependency: php-cli(x86-64) = 5.4.16-46.amzn2.0.2 for package: php-5.4.16-46.amzn2.0.2.x86_64
--> Processing Dependency: php-common(x86-64) = 5.4.16-46.amzn2.0.2 for package: php-5.4.16-46.amzn2.0.2.x86_64
--> Processing Dependency: httpd-mmn = 20120211x8664 for package: php-5.4.16-46.amzn2.0.2.x86_64
--> Processing Dependency: httpd for package: php-5.4.16-46.amzn2.0.2.x86_64
--> 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
```

2.Installing php

```

cd /var/www/html
sudo mkdir 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/swap on /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-web-lax';
$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;
}

```

3.index.php file code


```

no label, UUID=452b88c3-1ac0-485c-8c40-694465c12c37
[ec2-user@ip-172-31-18-37 face]$ sudo /sbin/swapoff /var/swap.1
swapoff: /var/swap.1: insecure permissions 0644, 0600 suggested.
[ec2-user@ip-172-31-18-37 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-172-31-18-37 face]$ sudo wget https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
--2020-04-02 05:21:21-- https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
Resolving i.pinimg.com (i.pinimg.com)... 23.197.108.230, 2a04:4e42:3b::84
Connecting to i.pinimg.com (i.pinimg.com)|23.197.108.230|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 215551 (210K) [image/jpeg]
Saving to: 'b97ea33b5842c7894b804923c6c05580.jpg'

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

2020-04-02 05:21:21 (5.61 MB/s) - 'b97ea33b5842c7894b804923c6c05580.jpg' saved [215551/215551]

[ec2-user@ip-172-31-18-37 face]$ ls
b97ea33b5842c7894b804923c6c05580.jpg composer.json composer.lock vendor
[ec2-user@ip-172-31-18-37 face]$ sudo mv b97ea33b5842c7894b804923c6c05580.jpg sample.jpg
[ec2-user@ip-172-31-18-37 face]$ ls
composer.json composer.lock sample.jpg vendor
[ec2-user@ip-172-31-18-37 face]$ sudo vim index.php
[ec2-user@ip-172-31-18-37 face]$ sudo php index.php
You must specify a non-null value for the Body or SourceFile parameters.
[ec2-user@ip-172-31-18-37 face]$ sudo vim index.php
[ec2-user@ip-172-31-18-37 face]$ sudo php index.php
PHP Parse error: syntax error, unexpected '<' in /var/www/html/face/index.php on line 55
[ec2-user@ip-172-31-18-37 face]$ sudo php index.php
PHP Parse error: syntax error, unexpected '<' in /var/www/html/face/index.php on line 55
[ec2-user@ip-172-31-18-37 face]$ sudo vim index.php
[ec2-user@ip-172-31-18-37 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-web-lax.s3.us-east-2.amazonaws.com/sample.jpg[ec2-user@ip-172-31-18-37 face]$

```

```

HTTP request sent, awaiting response... 200 OK
Length: 215551 (210K) [image/jpeg]
Saving to: 'b97ea33b5842c7894b804923c6c05580.jpg'

100%[=====]

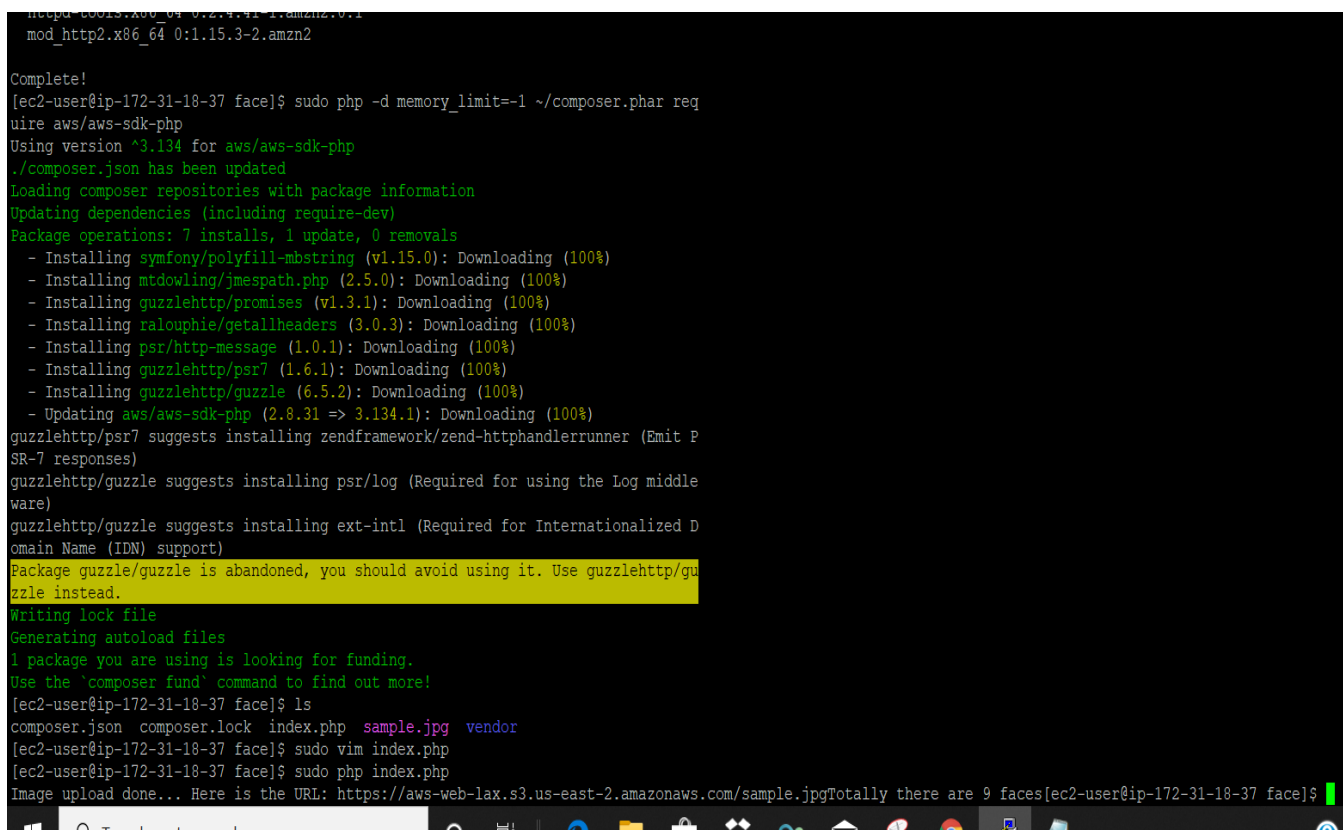
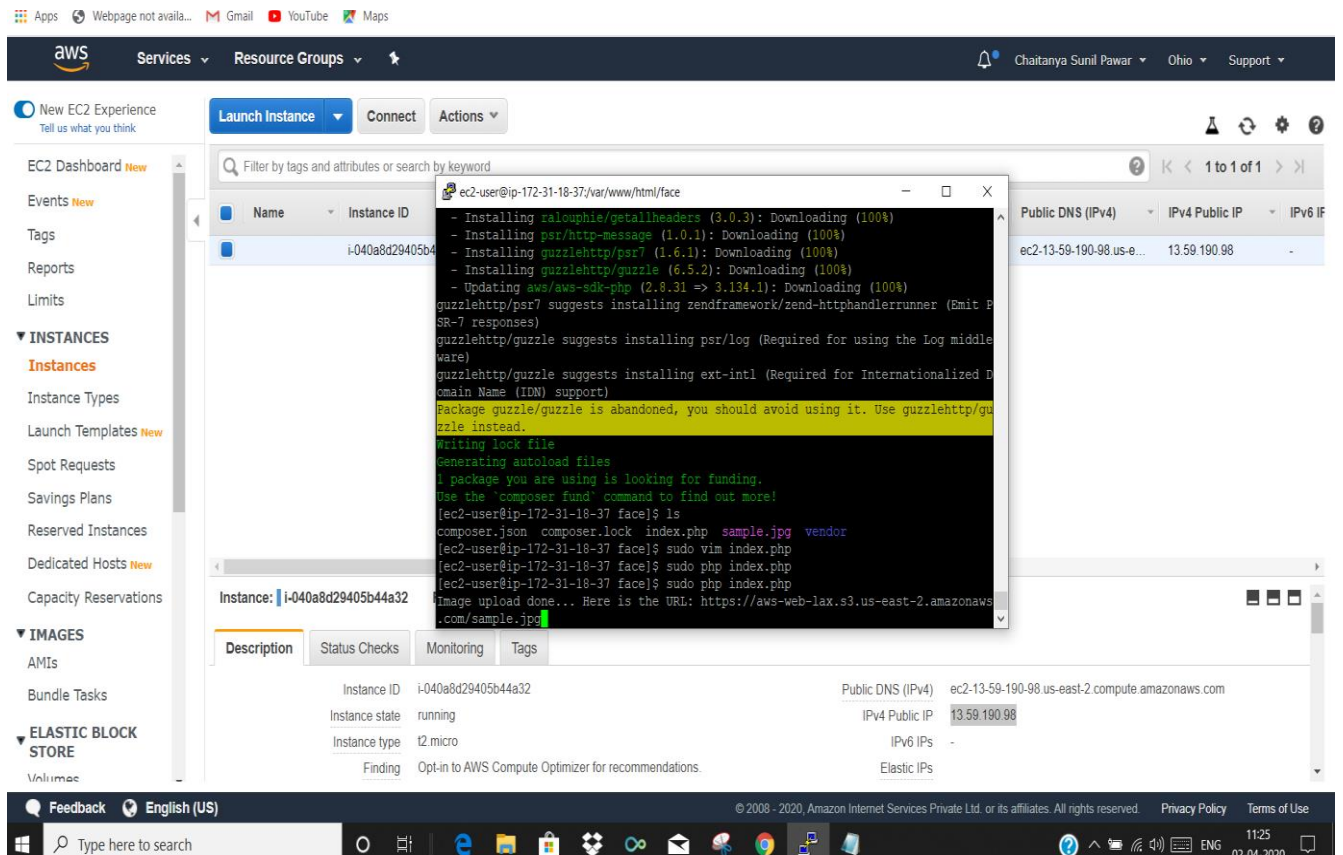
2020-04-02 05:21:21 (5.61 MB/s) - 'b97ea33b5842c7894b804923c6c05580.jpg' saved [215551/215551]

[ec2-user@ip-172-31-18-37 face]$ ls
b97ea33b5842c7894b804923c6c05580.jpg composer.json composer.lock vendor
[ec2-user@ip-172-31-18-37 face]$ sudo mv b97ea33b5842c7894b804923c6c05580.jpg sample.jpg
[ec2-user@ip-172-31-18-37 face]$ ls
composer.json composer.lock sample.jpg vendor
[ec2-user@ip-172-31-18-37 face]$ sudo vim index.php
[ec2-user@ip-172-31-18-37 face]$ sudo php index.php
You must specify a non-null value for the Body or SourceFile parameters.
[ec2-user@ip-172-31-18-37 face]$ sudo vim index.php
[ec2-user@ip-172-31-18-37 face]$ sudo php index.php
PHP Parse error: syntax error, unexpected '<' in /var/www/html/face/index.php on line 55
[ec2-user@ip-172-31-18-37 face]$ sudo php index.php
PHP Parse error: syntax error, unexpected '<' in /var/www/html/face/index.php on line 55
[ec2-user@ip-172-31-18-37 face]$ sudo vim index.php
[ec2-user@ip-172-31-18-37 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-web-lax.s3.us-east-2.amazonaws.com/sample.jpg[ec2-user@ip-172-31-18-37 face]$

```

4.Upload success screenshot

Screenshot needed for Ec2 & Rekognition



1.Face detection success screenshot