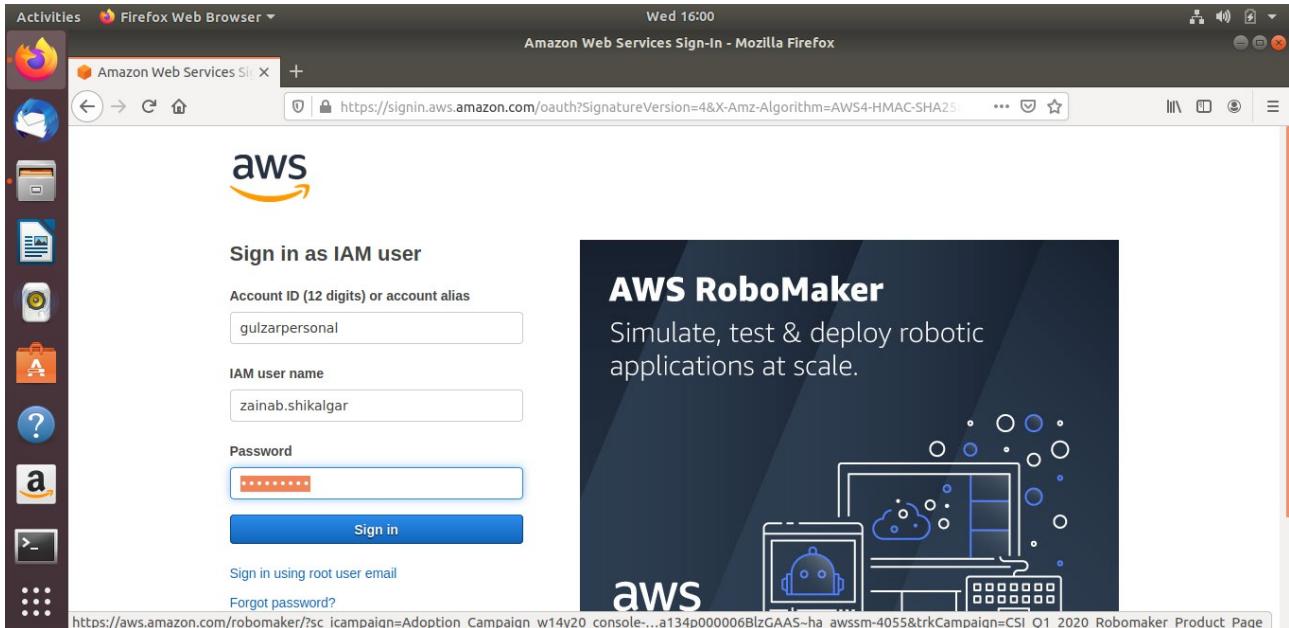


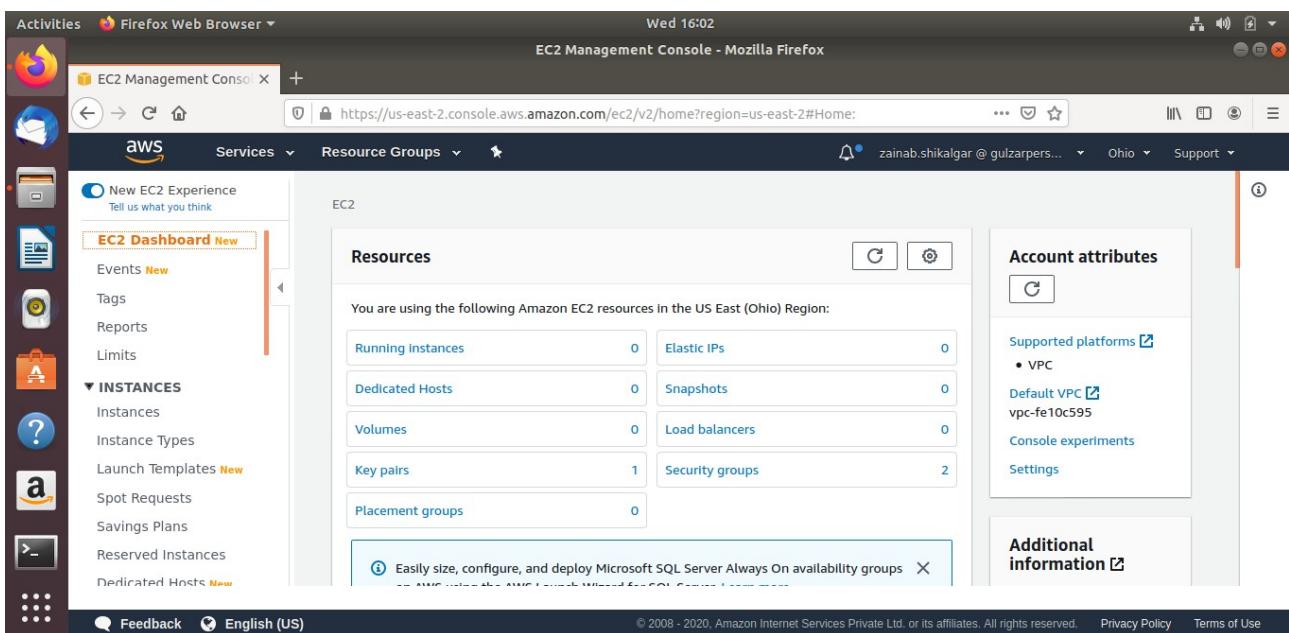
# Screenshots For AWS Project

- Screenshots needed for Dashboards

## 1. AWS Login screen with username



## 2. EC2 Dashboard



### 3. S3 Dashboard

The screenshot shows the AWS S3 Management Console interface. On the left, there's a sidebar titled "Amazon S3" with options like "Buckets", "Batch operations", "Access analyzer for S3", "Block public access (account settings)", and "Feature spotlight". The main area is titled "Amazon S3" and shows a table for "Buckets (1)". The table has columns for Name, Region, Access, and Bucket created. One row is listed: "aws-webinar-zainab" in US East (Ohio) region, with "Objects can be public" access and created on "2020-03-29T15:44:58.000Z". There are buttons for "Copy ARN", "Empty", "Delete", and "Create bucket". At the bottom, there's a footer with links for "Feedback", "English (US)", "Privacy Policy", and "Terms of Use".

### 4. Rekognition Dashboard

The screenshot shows the AWS Rekognition Console homepage. On the left, there's a sidebar titled "Amazon Rekognition" with options like "Custom Labels", "Demos", "Image moderation", "Facial analysis", "Celebrity recognition", "Face comparison", "Text in image", and "Video Demos". The main area has a dark blue background with a network graph pattern. It features the "Amazon Rekognition" logo and the text "Deep learning-based visual analysis service" and "Search, verify, and organize millions of images and videos". There are buttons for "Try Demo" and "Download SDKs". At the bottom, there are three icons: a stack of images, a circuit board, and a puzzle piece. The footer includes links for "Feedback", "English (US)", "Privacy Policy", and "Terms of Use".

- Screenshots needed for EC2

## 1. Choosing an AMI

The screenshot shows the AWS Management Console Launch Instance wizard. The user is on Step 1: Choose an Amazon Machine Image (AMI). The interface includes a sidebar with icons for various services like CloudWatch Metrics, Lambda, and S3. The main content area displays a search bar and a list of AMIs under 'Quick Start'. One item is highlighted: 'Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0e01ce4ee18447327 (64-bit x86) / ami-03201f374ab66a26e (64-bit Arm)'. Below it, a description states: '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.' There are also options for 'Root device type: ebs' and 'Virtualization type: hvm'. At the bottom right, there are 'Select' and 'Cancel and Exit' buttons.

## 2. Choosing an Instance Type

The screenshot shows the AWS Management Console Launch Instance wizard. The user is on Step 2: Choose an Instance Type. The interface includes a sidebar with icons for various services like CloudWatch Metrics, Lambda, and S3. The main content area displays a table of instance types. The table has columns for Family, Type, vCPUs, Memory (GiB), Instance Storage (GB), EBS-Optimized Available, Network Performance, and IPv6 Support. Two rows are visible: one for t2.nano (General purpose) and one for t2.micro (General purpose, Free tier eligible). The t2.micro row is currently selected. At the bottom right, there are 'Cancel', 'Previous', 'Review and Launch' (highlighted in blue), and 'Next: Configure Instance Details' buttons.

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	1	1	EBS only	-	Low to Moderate	Yes

### 3. Adding Storage

The screenshot shows the AWS Launch Instance wizard Step 4: Add Storage page. The URL is <https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard>. The page title is "Launch instance wizard | EC2 Management Console - Mozilla Firefox". The navigation bar includes "Services", "Resource Groups", and "AWS". The top menu has items like "1. Choose AMI", "2. Choose Instance Type", "3. Configure Instance", "4. Add Storage" (which is selected), "5. Add Tags", "6. Configure Security Group", and "7. Review". A sidebar on the left lists various AWS services. The main content area shows a table for adding storage volumes:

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

Below the table, a note says: "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 limits." At the bottom are "Cancel", "Previous", "Review and Launch" (which is highlighted in blue), and "Next: Add Tags".

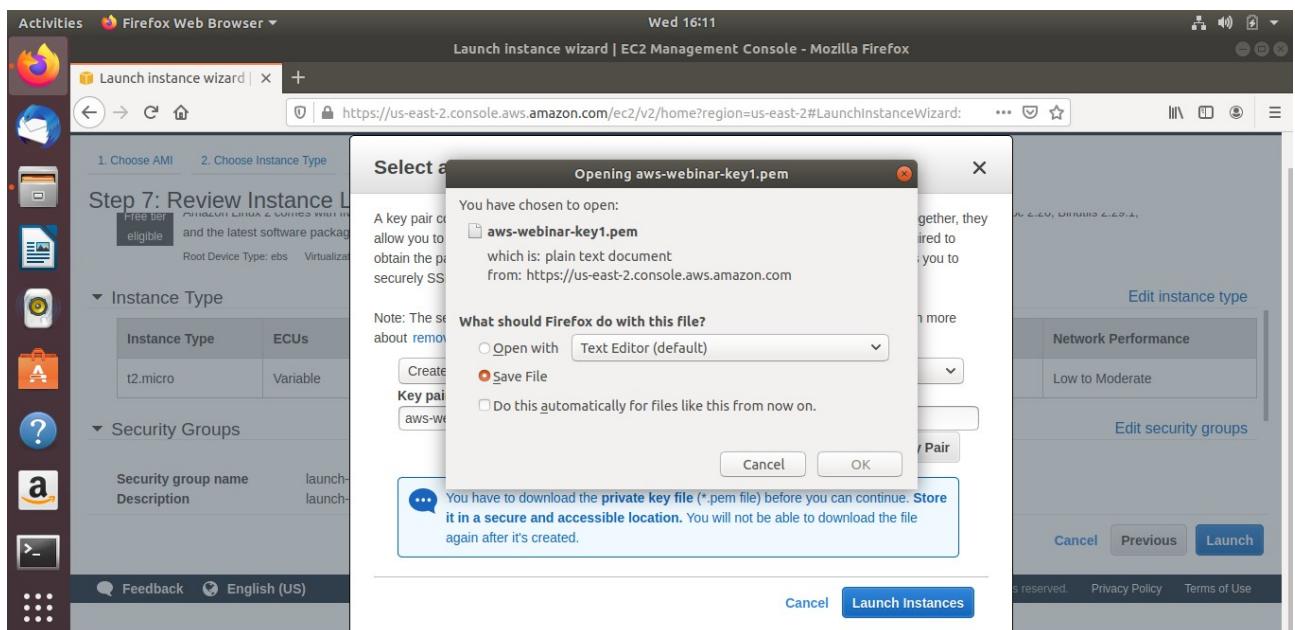
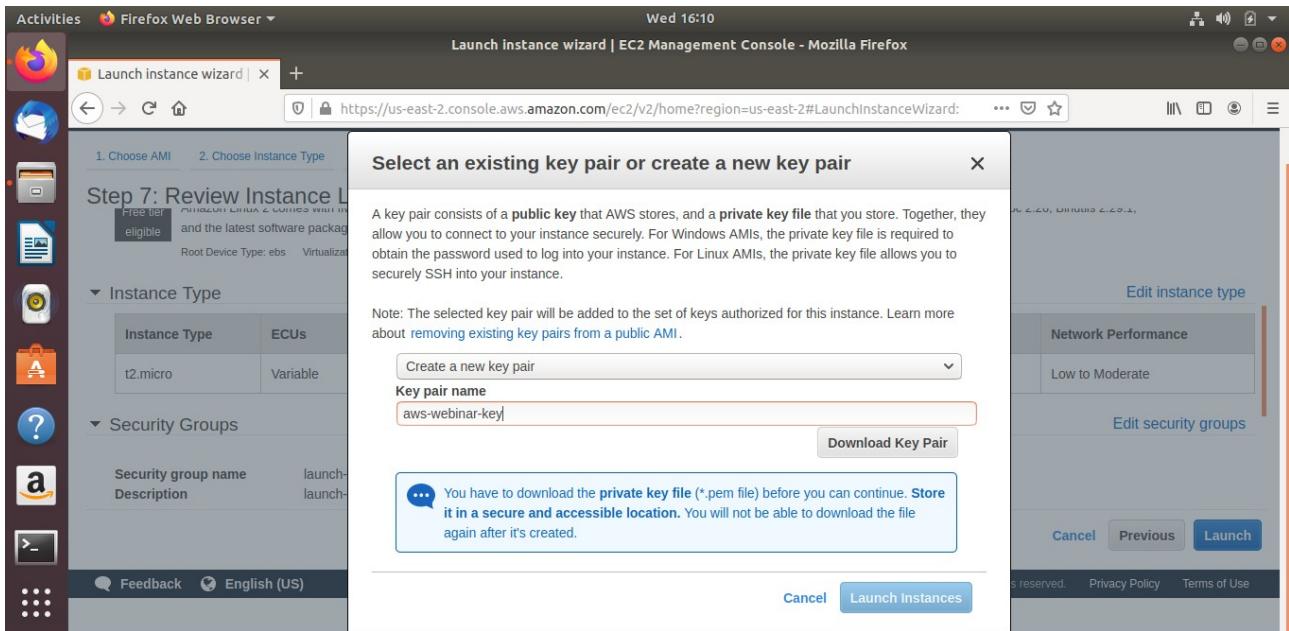
### 4. Configuring Security Group

The screenshot shows the AWS Launch Instance wizard Step 6: Configure Security Group page. The URL is <https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard>. The page title is "Launch instance wizard | EC2 Management Console - Mozilla Firefox". The navigation bar includes "Services", "Resource Groups", and "AWS". The top menu has items like "1. Choose AMI", "2. Choose Instance Type", "3. Configure Instance", "4. Add Storage", "5. Add Tags", "6. Configure Security Group" (which is selected), and "7. Review". A sidebar on the left lists various AWS services. The main content area shows a table for configuring security group rules:

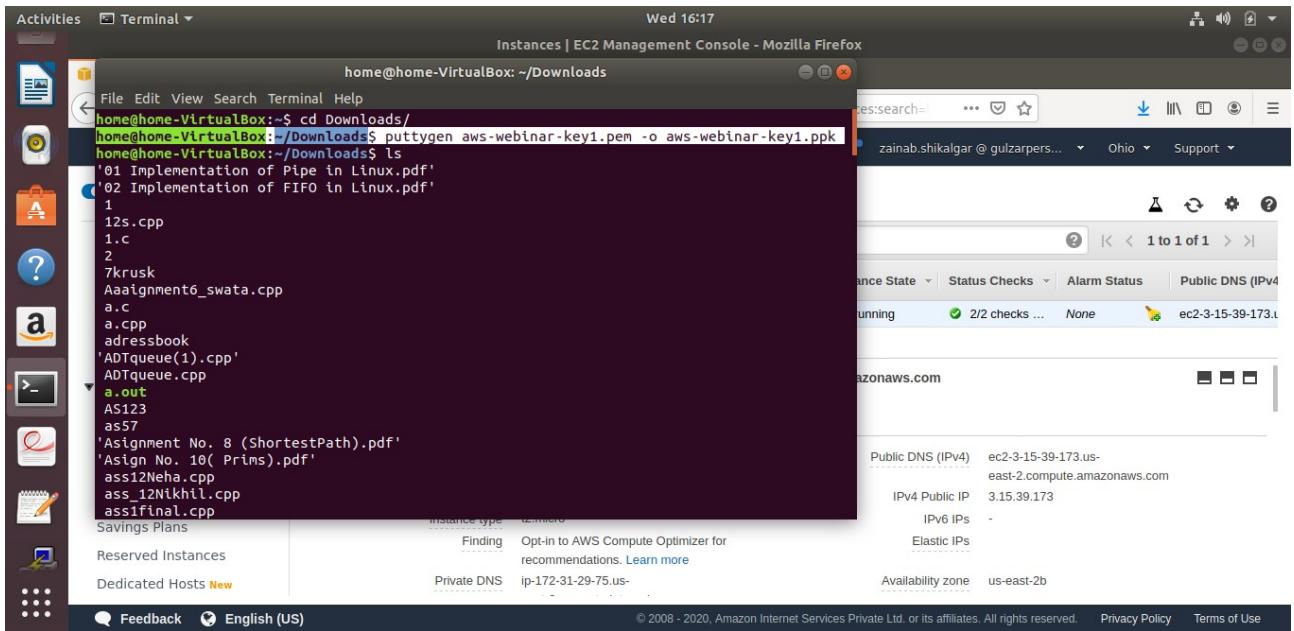
Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom	0.0.0.0/0

Below the table, a note says: "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." At the bottom are "Cancel", "Previous", "Review and Launch" (which is highlighted in blue), and "Next: Add Tags".

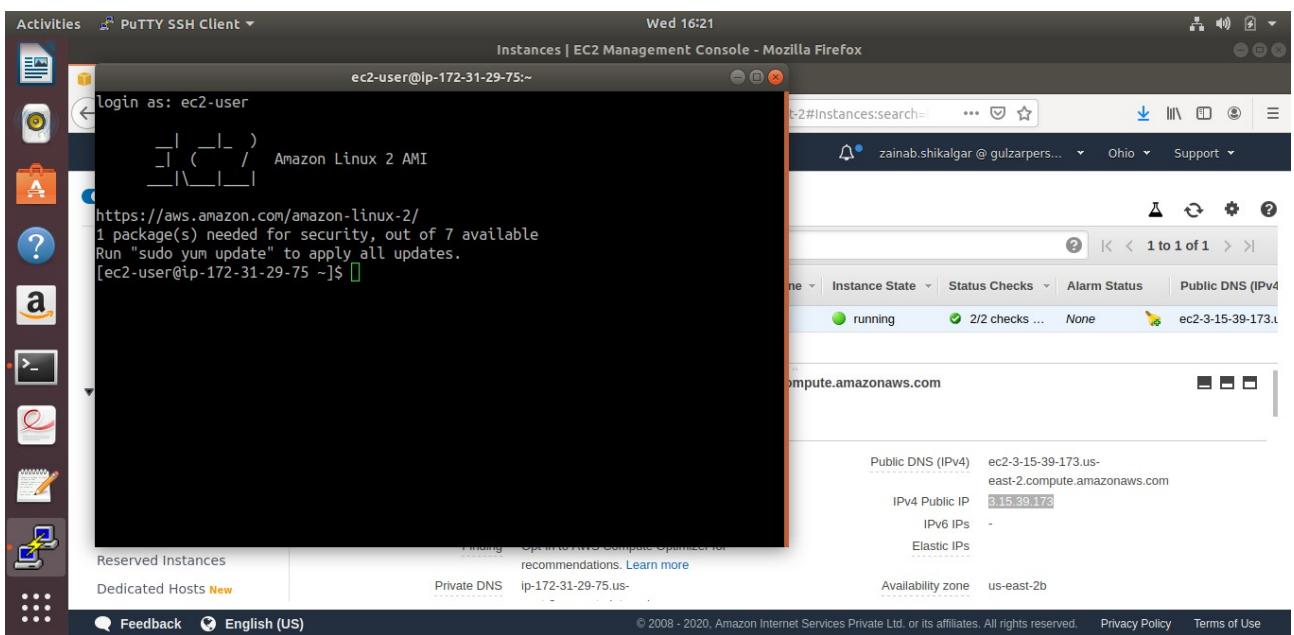
## 5. Key Pair Download



## 6. PuTTYgen conversion from pem to ppk



## 7. Logged in EC2 black screen



- Screenshots needed for S3

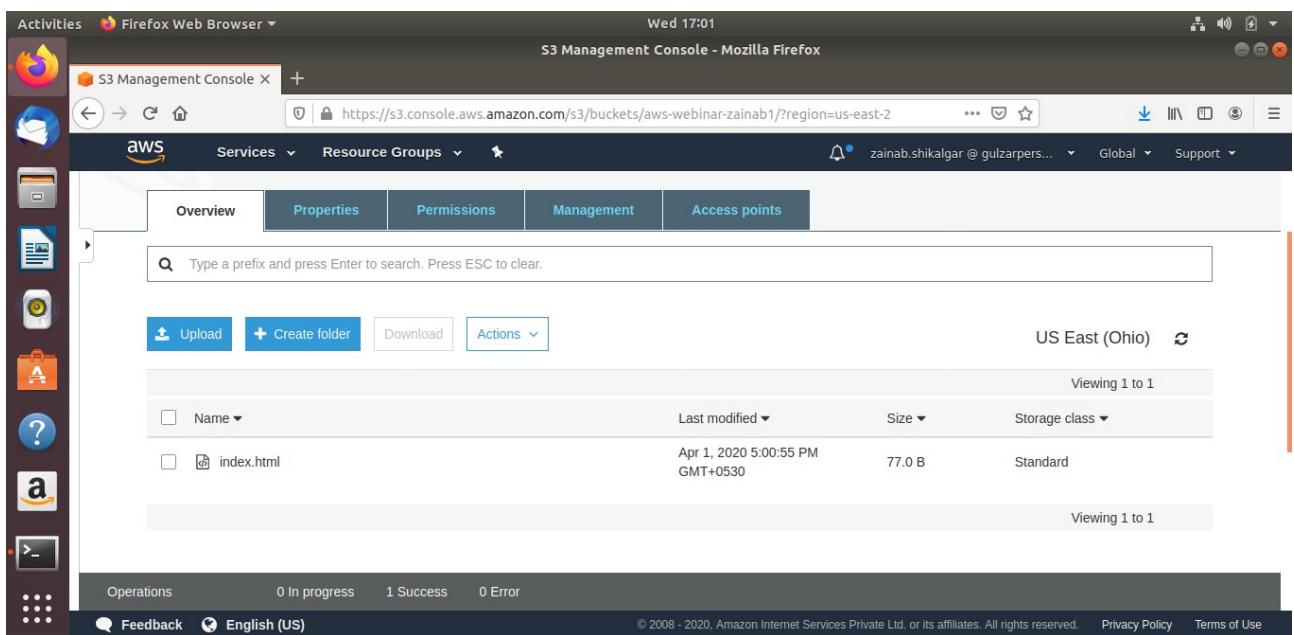
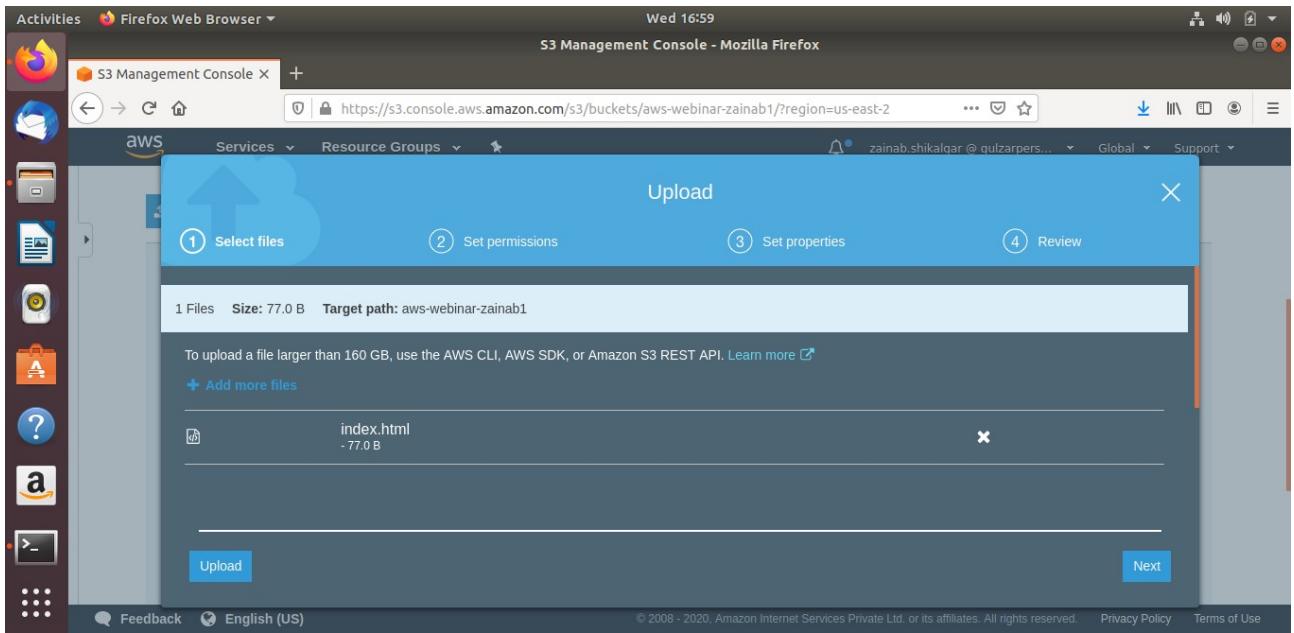
## 1. Creating a bucket

The screenshot shows the 'Create bucket' page in the AWS S3 Management Console. The 'Bucket name' field contains 'aws-webinar-zainab1'. The 'Region' dropdown is set to 'US East (Ohio) us-east-2'. A note below the region selection states: 'Bucket access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket'.

The screenshot shows the 'Amazon S3' dashboard after a bucket has been created. A green success message at the top states: 'Successfully created bucket aws-webinar-zainab1. To upload files and folders, or to configure additional bucket settings such as Bucket Versioning, tags, and default encryption, choose Go to bucket details.' Below this, the 'Buckets' section shows two buckets: 'aws-webinar-zainab' and 'aws-webinar-zainab1'. The table provides details for each bucket:

Name	Region	Access	Bucket created
aws-webinar-zainab	US East (Ohio) us-east-2	Objects can be public	2020-03-29T15:44:58.000Z
aws-webinar-zainab1	US East (Ohio) us-east-2	Not Public	2020-04-01T11:22:08.000Z

## 2. Uploading an Object



### 3. Enabling Static Website

The screenshot shows the AWS S3 Management Console in Mozilla Firefox. The main panel displays the 'Static website hosting' configuration for a bucket named 'aws-webinar-zainab1'. It includes fields for 'Index document' (set to 'index.html') and 'Error document' (set to 'error.html'). A sidebar on the right provides information about 'Object-level logging', stating that it records object-level API activity using CloudTrail data events. The status is set to 'Disabled'.

### 4. Making the Object Public

The screenshot shows the 'Block public access (bucket settings)' page in the AWS S3 Management Console. A success message box indicates 'Public access settings updated successfully'. Below, under 'Block all public access', the setting is currently 'Off'. There are three sub-options: '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)', and 'Block public access to buckets and objects granted through new public bucket or access point policies'. Each of these sub-options also has its own 'Off' setting.

## 5. Checking the S3 link on the browser

The screenshot shows the AWS S3 Management Console in Mozilla Firefox. The object details are as follows:

- Owner:** 8850e717ef6ee6388f7704200dec7a3313fc3ff2acddd6d4a81c3a8903c30271
- Last modified:** Apr 1, 2020 5:00:55 PM GMT+0530
- Etag:** 97f069191ad825f34e920b11bd9c2b26
- Storage class:** Standard
- Server-side encryption:** None
- Size:** 77.0 B
- Key:** index.html
- Object URL:** <https://aws-webinar-zainab1.s3.us-east-2.amazonaws.com/index.html>

At the bottom, the status bar shows: Operations 0 In progress 1 Success 0 Error.

The screenshot shows the AWS S3 Management Console in Mozilla Firefox. The page content is:

Hii i am Zainab from Pune. Learning AWS Service on EthnusCodemithra Webinar.

- Screenshots needed for Rekognition

## 1. Face Detect

The screenshot shows the AWS Rekognition Console interface. On the left, there's a sidebar with various services like Demos, Object and scene detection, Image moderation, Facial analysis, etc. The main content area is titled "Object and scene detection" and displays a street scene with several cars and a person. A bounding box highlights the person, with the label "Person" above it. Below the image, a list of detected objects and their confidence scores is shown:

Category	Confidence (%)
Transportation	98.8 %
Vehicle	98.8 %
Automobile	98.8 %
Car	98.8 %
Person	98.3 %

## 2. Face Compare

The screenshot shows the AWS Rekognition Console interface. The sidebar is identical to the previous one. The main content area is titled "Face comparison" and shows a reference face (a girl smiling) and two comparison faces (two girls looking at a phone). Below the images, a similarity percentage is displayed:

Comparison Face	Similarity (%)
Two girls looking at a phone	99.8 %

### 3. Celebrity Recognition

Activities Firefox Web Browser ▾ Thu 23:50 Rekognition Console - Mozilla Firefox

Rekognition Console Instances | EC2 Manager 3.15.39.173/face/index.php Telegram bot link - zaina

aws Services Resource Groups

Demos Object and scene detection Image moderation Facial analysis **Celebrity recognition** Face comparison Text in image Video Demos Video analysis Metrics Metrics

Use Custom Labels

Celebrity recognition

Rekognition automatically recognizes celebrities in images and provides confidence scores.

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

Results

Andy Jassy

Match confidence 100 %

Request Response

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

### 4. Text in Image

Activities Firefox Web Browser ▾ Thu 23:51 Rekognition Console - Mozilla Firefox

Rekognition Console Instances | EC2 Manager 3.15.39.173/face/index.php Telegram bot link - zaina

aws Services Resource Groups

Demos Object and scene detection Image moderation Facial analysis **Celebrity recognition** Face comparison **Text in image** Video Demos Video analysis Metrics Metrics

Use Custom Labels

Text in image

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

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

Results US English only

| IT'S |  
| MONDAY |  
| but | keep |  
| Smiling |

Request Response

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

## • Screenshots needed for EC2 & S3

### 1. Installing aws-sdk

```

Wed 21:17
*index.php
~/Downloads

ec2-user@ip-172-31-29-75:/var/www/html/face
httpd-tools.x86_64 0:2.4.41-1.amzn2.0.1
mod_http2.x86_64 0:1.15.3-2.amzn2

Complete!
[ec2-user@ip-172-31-29-75 face]$ sudo php -d memory_limit=-1 ~/composer.phar req
ure aws/aws-sdk-php
Using version ^3.134 for aws/aws-sdk-php
./composer.json has been created
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 8 installs, 0 updates, 0 removals
- Installing symfony/polyfill-mbstring (v1.15.0): Downloading (100%
- Installing mtwdowling/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%
- Installing aws/aws-sdk-php (3.134.0): Downloading (100%
guzzlehttp/psr7 suggests installing zendframework/zend-httpHandlerrunner (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
[ec2-user@ip-172-31-29-75 face]$ [REDACTED]

```

PHP ▾ Tab Width: 8 ▾ Ln 15, Col 68 ▾ INS

```

Wed 21:17
*index.php
~/Downloads

ec2-user@ip-172-31-29-75:/var/www/html/face
Package operations: 8 installs, 0 updates, 0 removals
- Installing symfony/polyfill-mbstring (v1.15.0): Downloading (100%
- Installing mtwdowling/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%
- Installing aws/aws-sdk-php (3.134.0): Downloading (100%
guzzlehttp/psr7 suggests installing zendframework/zend-httpHandlerrunner (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
[ec2-user@ip-172-31-29-75 face]$ [REDACTED]

```

PHP ▾ Tab Width: 8 ▾ Ln 15, Col 68 ▾ INS

## 2. Installing php

A screenshot of a Linux desktop environment. On the left is a dock with various icons. In the center is a terminal window titled 'PUTTY SSH Client' with the command history shown. To the right of the terminal is a code editor window titled 'index 2.php' containing some PHP code.

```
1 Apr 01 10:54:13 ip-172-31-29-75.us-east-2.compute.internal systemd[1]: Starti...
2 Apr 01 10:54:13 ip-172-31-29-75.us-east-2.compute.internal systemd[1]: Starte...
3 Hint: Some lines were ellipsized, use -l to show in full.
4 [ec2-user@ip-172-31-29-75 ~]$ sudo vim /var/www/html/index.html
5 [ec2-user@ip-172-31-29-75 ~]$ ls
6 [ec2-user@ip-172-31-29-75 ~]$ sudo yum install php
7 Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
8 amzn2-core | 2.4 kB     00:00
10 Resolving Dependencies
11 --> Running transaction check
12 ---> Package php.x86_64 0:5.4.16-46.amzn2.0.2 will be installed
13 ---> Processing Dependency: php-cli(x86-64) = 5.4.16-46.amzn2.0.2 for package: ph
14 p-5.4.16-46.amzn2.0.2.x86_64
15 ---> Processing Dependency: php-common(x86-64) = 5.4.16-46.amzn2.0.2 for package:
16 php-5.4.16-46.amzn2.0.2.x86_64
17 ---> Running transaction check
18 ---> Package php-cli.x86_64 0:5.4.16-46.amzn2.0.2 will be installed
19 ---> Package php-common.x86_64 0:5.4.16-46.amzn2.0.2 will be installed
20 ---> Processing Dependency: libzip.so.2()(64bit) for package: php-common-5.4.16-4
21 6.amzn2.0.2.x86_64
22 ---> Running transaction check
23 ---> Package libzip010-compat.x86_64 0:0.10.1-9.amzn2.0.5 will be installed
24 ---> Finished Dependency Resolution
25
26 sudo yum remove php*
27 sudo yum remove httpd*
28 sudo yum clean all
29 sudo yum upgrade -v
```

A screenshot of a Linux desktop environment, similar to the one above. It shows a terminal window and a code editor window. The terminal window displays the output of a 'yum' command, showing the installation progress of PHP and its dependencies.

```
Total          23 MB/s | 4.7 MB  00:00
1 Running transaction check
2 Running transaction test
3 Transaction test succeeded
4 Running transaction
5   Installing : libzip010-compat-0.10.1-9.amzn2.0.5.x86_64           1/4
6   Installing : php-common-5.4.16-46.amzn2.0.2.x86_64                 2/4
7   Installing : php-cli-5.4.16-46.amzn2.0.2.x86_64                   3/4
8   Installing : php-5.4.16-46.amzn2.0.2.x86_64                     4/4
9   Verifying  : libzip010-compat-0.10.1-9.amzn2.0.5.x86_64           1/4
10  Verifying : php-5.4.16-46.amzn2.0.2.x86_64                     2/4
11  Verifying : php-cli-5.4.16-46.amzn2.0.2.x86_64                   3/4
12  Verifying : php-common-5.4.16-46.amzn2.0.2.x86_64                 4/4
13
14 Installed:
15   php.x86_64 0:5.4.16-46.amzn2.0.2
16
17 Dependency Installed:
18   libzip010-compat.x86_64 0:0.10.1-9.amzn2.0.5
19   php-cli.x86_64 0:5.4.16-46.amzn2.0.2
20   php-common.x86_64 0:5.4.16-46.amzn2.0.2
21
22 Complete!
23 [ec2-user@ip-172-31-29-75 ~]$ 
24
25 sudo yum remove php*
26 sudo yum remove httpd*
27 sudo yum clean all
28 sudo yum upgrade -v
```

### 3. index.php file code

```
Activities ▾ PutTY SSH Client ▾ Thu 21:59
ec2-user@ip-172-31-29-75:/var/www/html

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

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

$bucket = 'aws-webinar-zainab1';
$keyname = 'sample.jpg';

$s3 = new S3Client([
    '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;
}

-- INSERT --
```

```
Activities ▾ PutTY SSH Client ▾ Thu 21:59
ec2-user@ip-172-31-29-75:/var/www/html

        '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;
}
-- INSERT --
```

## 4. Upload success screenshot

The screenshot shows a terminal window titled "PUTTY SSH Client" with the following session details:

- Activities: PUTTY SSH Client
- Open: ~
- Save: Save
- Tab Width: 8
- Ln 98, Col 1
- INS

The terminal output is as follows:

```
Thu 22:02
index.php
~/Downloads
ec2-user@ip-172-31-29-75:/var/www/html/face
1 package you are using is looking for funding.
Use the `composer fund` command to find out more!
[ec2-user@ip-172-31-29-75 face]$ sudo wget https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
--2020-04-02 16:30:46-- https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
Resolving i.pinimg.com (i.pinimg.com)... 184.86.252.250, 2600:1408:20:aac::1931,
2600:1408:20:a86::1931, ...
Connecting to i.pinimg.com (i.pinimg.com)|184.86.252.250|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 215551 (210K) [image/jpeg]
Saving to: 'b97ea33b5842c7894b804923c6c05580.jpg'

100%[=====] 215,551     ----K/s   in 0.03s
2020-04-02 16:30:46 (6.99 MB/s) - 'b97ea33b5842c7894b804923c6c05580.jpg' saved [215551/215551]
[ec2-user@ip-172-31-29-75 face]$ sudo mv b97ea33b5842c7894b804923c6c05580.jpg sample.jpg
[ec2-user@ip-172-31-29-75 face]$ ls
composer.json composer.lock sample.jpg vendor
[ec2-user@ip-172-31-29-75 face]$ sudo vim index.php
[ec2-user@ip-172-31-29-75 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-webinar-zainab1.s3.us-east-2.amazonaws.com/sample.jpgTotally there are 9 faces[ec2-user@ip-172-31-29-75 face]$
```

- **Screenshots needed for EC2 & Rekognition**

## 1. Face Detect success screenshot

