

# SCREENSHOTS FOR STEPS IN AWS WEB HOSTING AND USING S3 SERVICES

**Name:** V. Lakshmi Sriya

**Email:** lakshmisriya.varada@vitap.ac.in

**AWS Services used:**

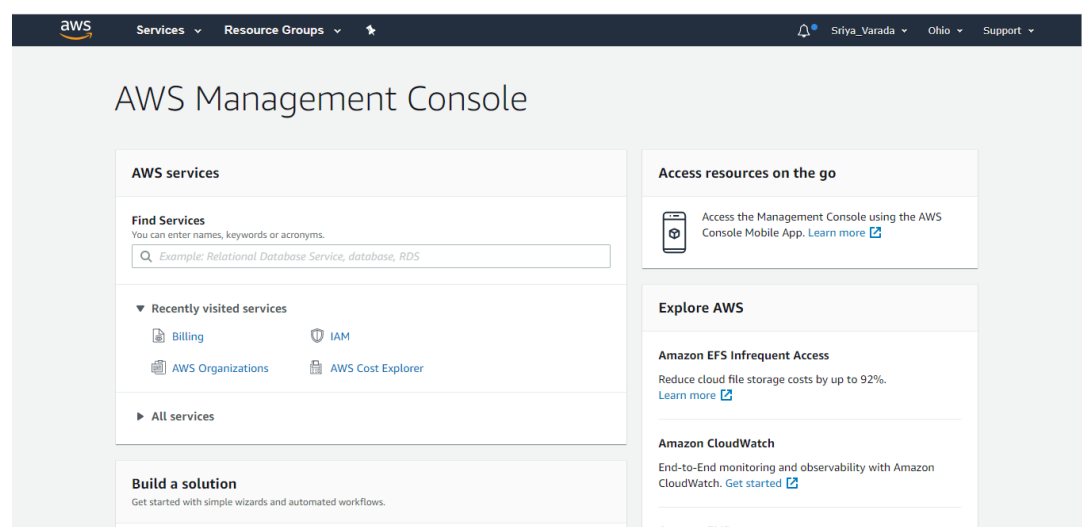
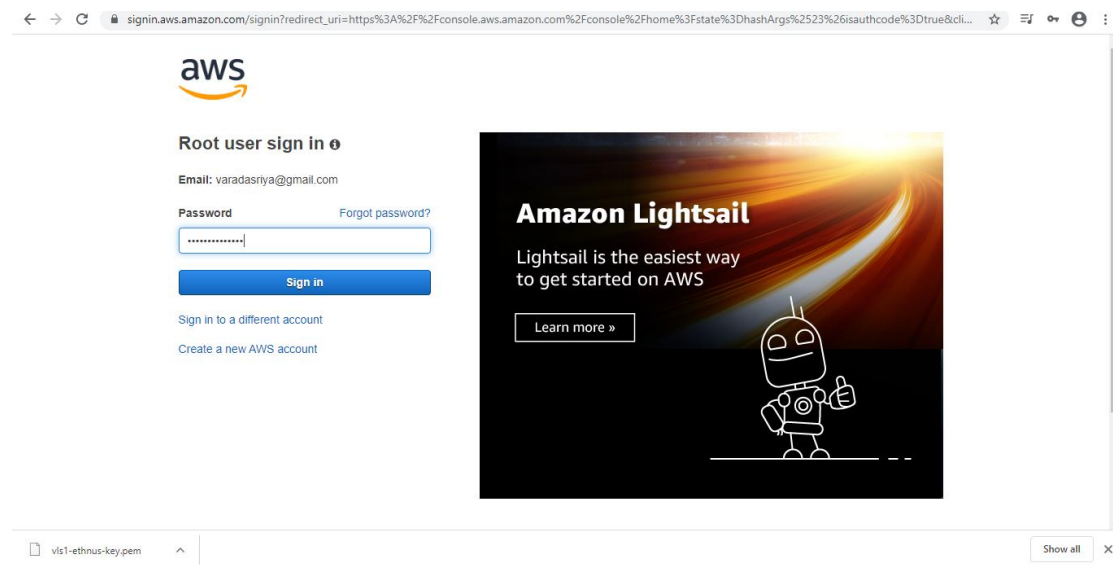
1.EC2

2.S3

3.Amazon Rekognition

**Screenshots needed for Dashboards**

1.AWS Login screen with username



## 2. EC2 Dashboard

The screenshot displays the AWS Management Console for the EC2 service. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and user information (Sriya\_Varada, Ohio, Support). A left-hand sidebar lists navigation options like 'New EC2 Experience', 'EC2 Dashboard', 'Events', 'Tags', 'Reports', 'Limits', 'INSTANCES', and 'IMAGES'. The main content area is titled 'EC2' and features a blue banner announcing the new console design. Below the banner, the 'Resources' section shows usage for the US East (Ohio) Region: Running instances (0), Elastic IPs (0), Dedicated Hosts (0), Snapshots (0), Volumes (0), Load balancers (0), Key pairs (0), Security groups (1), and Placement groups (0). A right-hand sidebar contains 'Account attributes' (Supported platforms, Default VPC, Console experiments, Settings) and 'Explore AWS' (Save with AMD EPYC-Powered F7 instances). A bottom footer bar includes 'Feedback', 'English (US)', and copyright information.

**Resources**

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

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

**Account attributes**

- Supported platforms
  - VPC
- Default VPC
  - vpc-60f5210b
- Console experiments
- Settings

**Explore AWS**

Save with AMD EPYC-Powered F7 instances

## 3.S3 Dashboard

The screenshot displays the AWS Management Console for the S3 service. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and user information (Sriya\_Varada, Global, Support). A left-hand sidebar lists navigation options like 'Amazon S3', 'Buckets', 'Batch operations', 'Access analyzer for S3', 'Block public access (account settings)', and 'Feature spotlight'. The main content area is titled 'Amazon S3' and features a blue banner announcing the new console design. Below the banner, the 'Buckets (0)' section shows a search bar and a table with columns: Name, Region, Access, and Bucket created. The table is empty, and a message states 'No buckets. You don't have any buckets.' with a 'Create bucket' button.

**Buckets (0)**

Copy ARN Empty Delete Create bucket

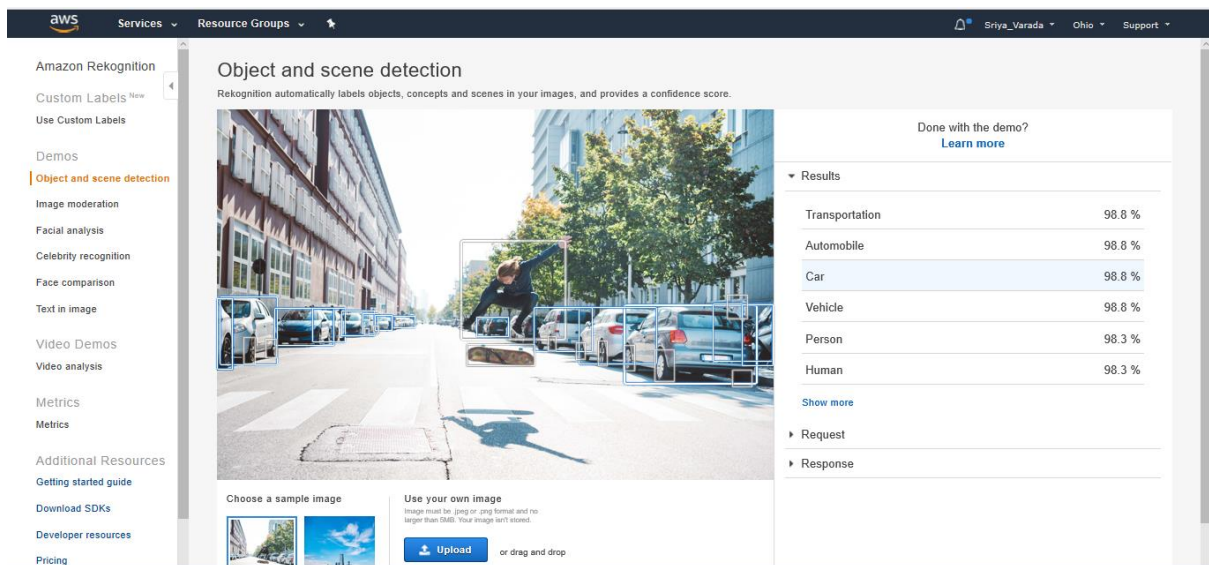
Find bucket by name

Name	Region	Access	Bucket created
No buckets			

You don't have any buckets.

Create bucket

## 4. Rekognition Dashboard



**Object and scene detection**  
Rekognition automatically labels objects, concepts and scenes in your images, and provides a confidence score.

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

**Results**

Transportation	98.8 %
Automobile	98.8 %
Car	98.8 %
Vehicle	98.8 %
Person	98.3 %
Human	98.3 %

[Show more](#)

► Request  
► Response

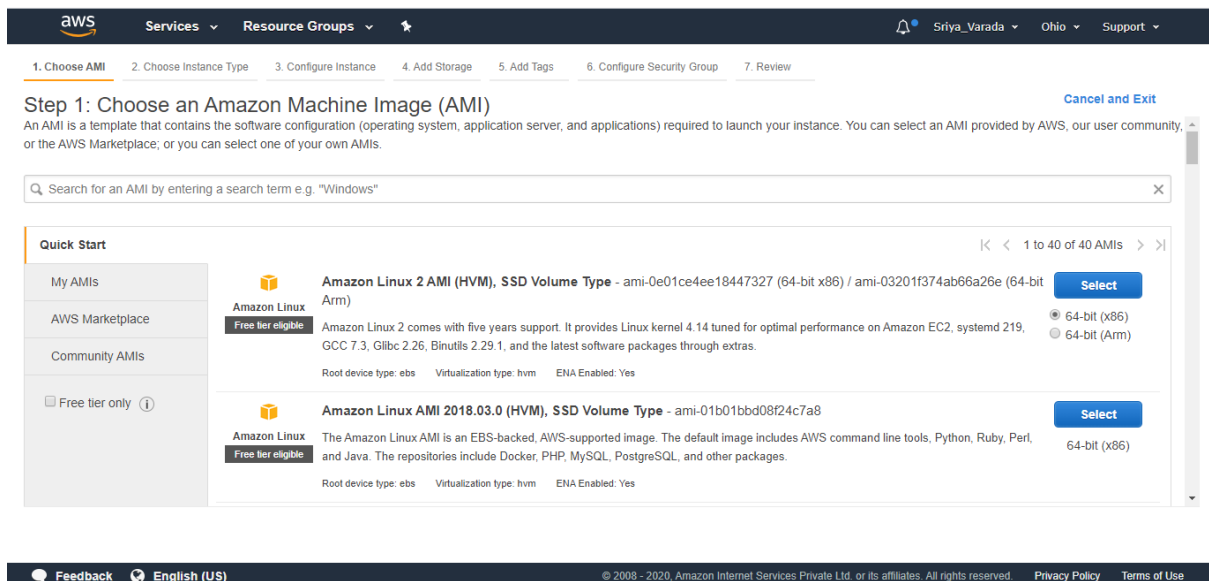
Choose a sample image

Use your own image  
Image must be .jpg or .png format and no larger than 5MB. Your image isn't loaded.

[Upload](#) or drag and drop

## Screenshots needed for EC2

### 1.Choosing an AMI



1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. 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.

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) [Select](#)

**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 [Select](#)

**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

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

## 2. Choosing an Instance Type

aws

Services

Resource Groups

Sriya\_Varada

Ohio

Support

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

Step 2: Choose an Instance Type

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

CancelPreviousReview and LaunchNext: Configure Instance Details

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

## 3. Adding Storage

aws

Services

Resource Groups

Sriya\_Varada

Ohio

Support

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

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.

CancelPreviousReview and LaunchNext: Add Tags

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

## 4. Configuring Security Group

aws

Services

Resource Groups

Sriya\_Varada

Ohio

Support

1. Choose AMI2. Choose Instance Type3. Configure Instance4. Add Storage5. Add Tags6. Configure Security Group7. 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](#)

## 5. Key Pair Download

aws

Services

Resource Groups

Sriya\_Varada

Ohio

Support

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

### Step 7: Review Instance Launch

Instance Type

ECUs

t2.micro

Variable

Security Groups

Security group name

launch-wizard-1

Description

launch-wizard-1

Type

SSH

Protocol

TCP

Instance Details

Storage

Tags

Network Performance

Low to Moderate

Edit security groups

Description

Edit instance details

Edit storage

Edit tags

Cancel

Previous

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**

vis-ethnus-key

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

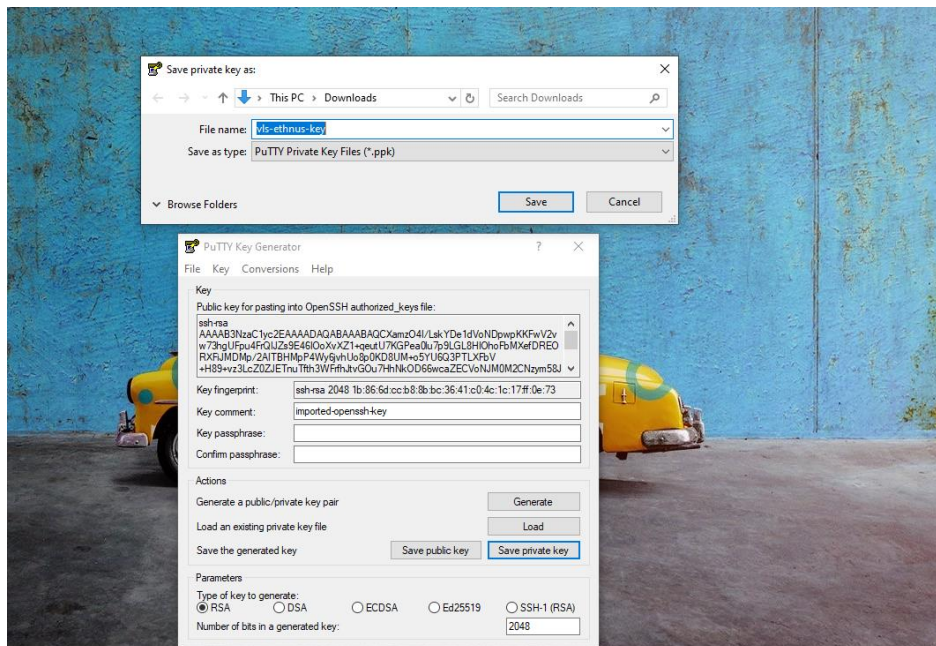
Launch Instances

Feedback

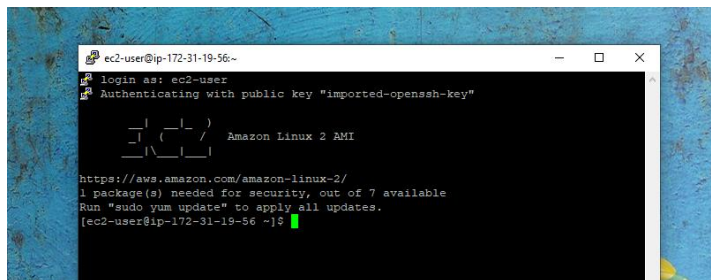
English (US)

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

## 6. PuTTYgen conversion from pem to ppk



## 7. Logged in EC2 black screen



The image shows the AWS Management Console interface. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and user information. The left sidebar shows the 'EC2 Dashboard' and various navigation links. The main content area is titled 'Security Groups (1/1)' and shows a table of security groups. The table has columns for 'Security group ID', 'Security group name', 'VPC ID', 'Description', and 'Owner'. The first row is highlighted, showing 'sg-07566942503804241' for 'launch-wizard-1' in VPC 'vpc-60f5210b'. Below the table, there is a detailed view of the selected security group, showing its 'Type', 'Protocol', 'Port range', 'Source', and 'Description - optional'.

Type	Protocol	Port range	Source	Description - optional
HTTP	TCP	80	0.0.0.0/0	-
HTTP	TCP	80	:::0	-
SSH	TCP	22	0.0.0.0/0	-

# Screenshots needed for S3

## 1. Creating a bucket

The screenshot shows the 'General configuration' step of the AWS S3 'Create bucket' wizard. The 'Bucket name' field is populated with 'vls-bitbucket'. Below it, a note states: 'Bucket name must be unique and must not contain spaces or uppercase letters. See rules for bucket naming'. The 'Region' dropdown is set to 'US East (Ohio) us-east-2'. The 'Bucket settings for Block Public Access' section is expanded, showing the 'Block all public access' checkbox checked. Below this, two sub-settings are listed: 'Block public access to buckets and objects granted through new access control lists (ACLs)' and 'Block public access to buckets and objects granted through any access control lists (ACLs)', both of which are also checked. The footer of the wizard includes 'Feedback', 'English (US)', and copyright information for 2008-2020 Amazon Internet Services Private Ltd.

aws Services Resource Groups

General configuration

Bucket name

vls-bitbucket

Bucket name must be unique and must not contain spaces or uppercase letters. See rules for bucket naming

Region

US East (Ohio) us-east-2

Bucket settings for Block Public Access

Public 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 and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

☒ Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☒ Block public access to buckets and objects granted through new access control lists (ACLs)

S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

☒ Block public access to buckets and objects granted through any access control lists (ACLs)

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

## 2. Uploading an Object

The screenshot shows the 'Upload' dialog box in the AWS S3 console. The dialog has a blue header with the title 'Upload' and a close button. Below the header, there are four steps: 1. Select files, 2. Set permissions, 3. Set properties, and 4. Review. The 'Select files' step is active. It shows a list of files with columns for 'Files', 'Size', and 'Target path'. One file is listed: 'index.html' with a size of '35.0 B' and a target path of 'vls-bitbucket'. Below the list, there is a note: 'To upload a file larger than 160 GB, use the AWS CLI, AWS SDK, or Amazon S3 REST API. Learn more'. There is a link to 'Add more files'. At the bottom of the dialog, there are two buttons: 'Upload' and 'Next'. The footer of the dialog includes 'Feedback', 'English (US)', and copyright information for 2008-2020 Amazon Internet Services Private Ltd.

aws Services Resource Groups

Upload

1 Select files 2 Set permissions 3 Set properties 4 Review

1 Files Size: 35.0 B Target path: vls-bitbucket

To upload a file larger than 160 GB, use the AWS CLI, AWS SDK, or Amazon S3 REST API. Learn more

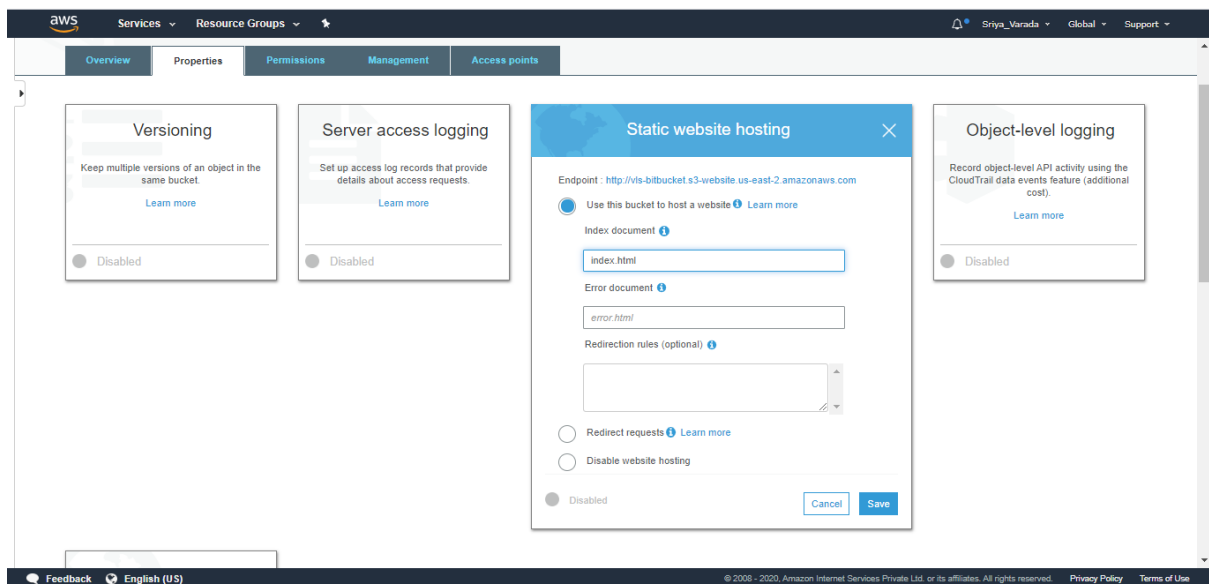
+ Add more files

index.html - 35.0 B

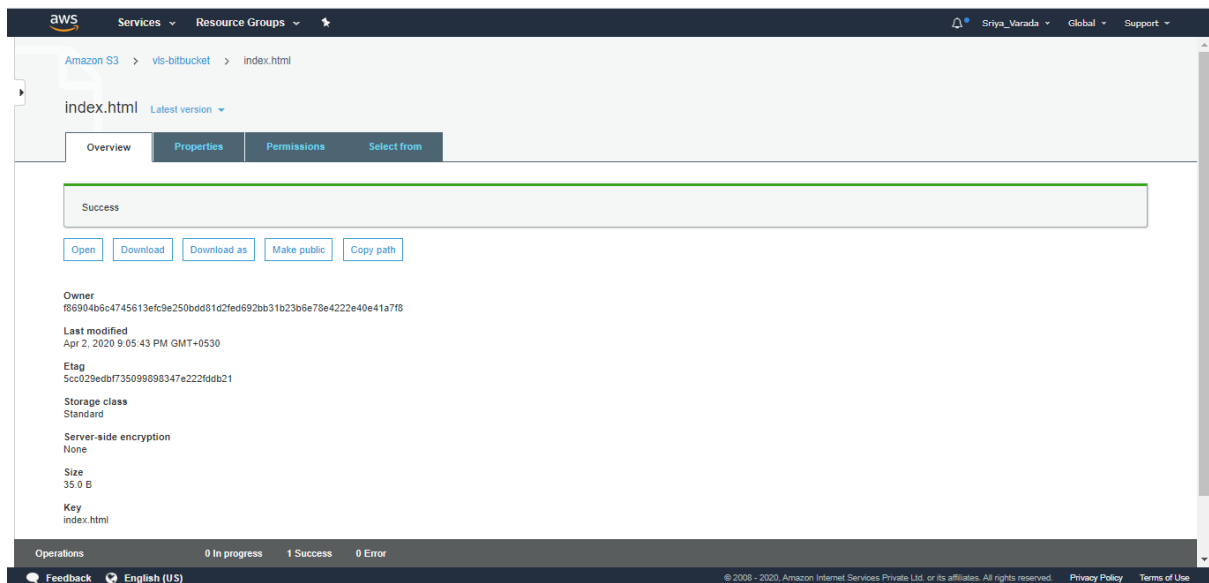
Upload Next

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

### 3. Enabling Static Website

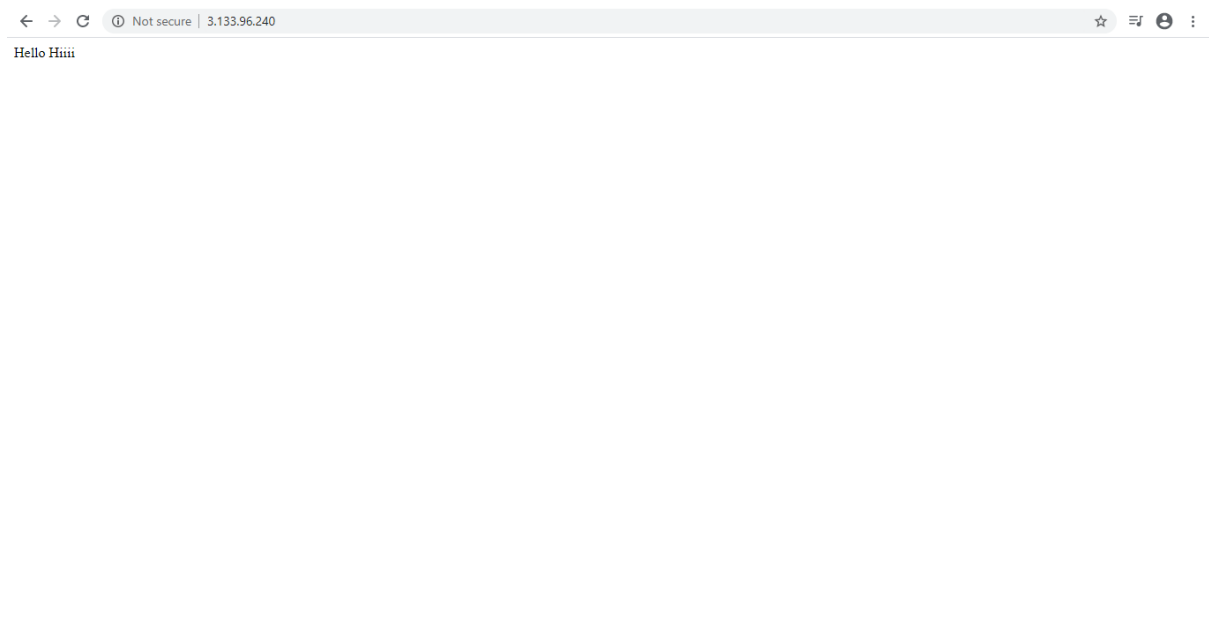


### 4. Making the Object Public



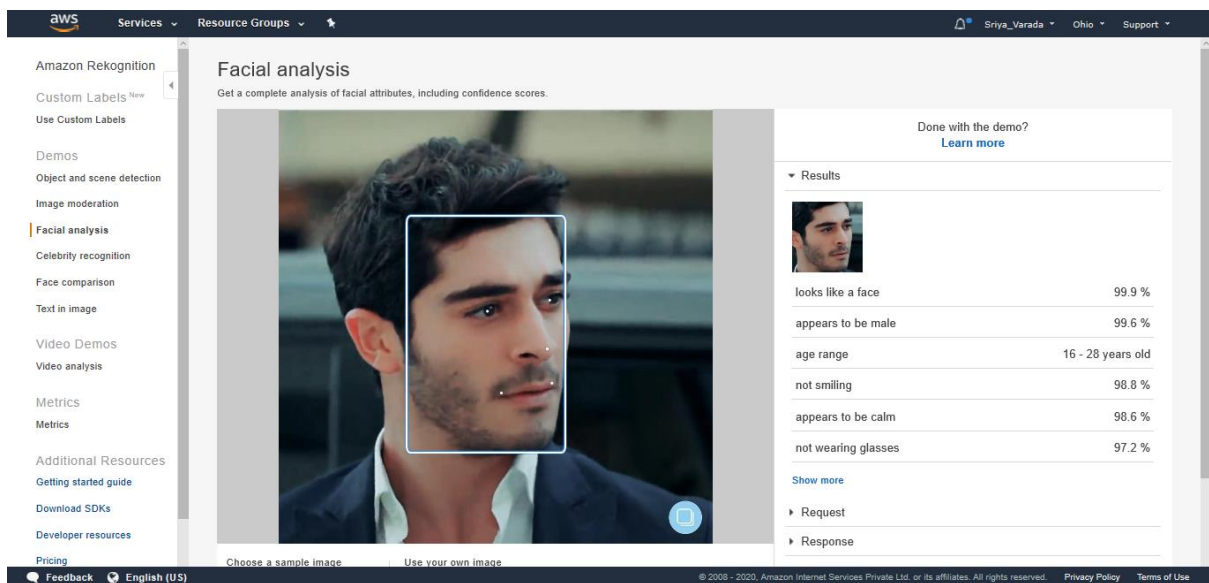


## 5. Checking the S3 link on the browser



## Screenshots needed for Rekognition

### 1.Face Detect



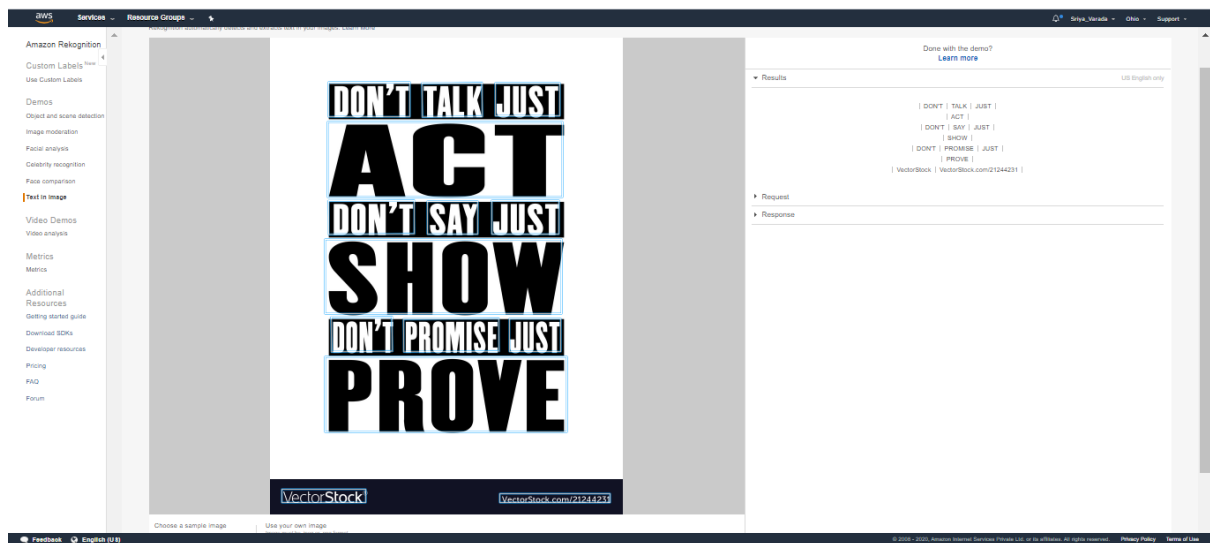
## 2. Face Compare

The screenshot shows the AWS Face Compare demo interface. On the left is a navigation menu with options like Amazon Rekognition, Custom Labels, Demos, and various services. The main area is split into two columns: 'Reference face' and 'Comparison face'. The 'Reference face' column shows a close-up of a man's face. The 'Comparison face' column shows a group photo of several people, with a red banner at the bottom that says 'EPISODE - 31'. Below these images are sections for 'Choose a sample image' and 'Use your own image', each with an 'Upload' button and a 'Use image URL' field. On the right side, there is a 'Results' section titled 'Done with this demo? Learn more'. It displays a list of similarity results, with the top result showing a similarity score of 98.1% between the reference face and a face in the comparison image.

## 3. Celebrity Recognition

The screenshot shows the AWS Celebrity Recognition demo interface. On the left is a navigation menu similar to the one in the previous screenshot. The main area is titled 'Celebrity recognition' and includes the text 'Recognition automatically recognizes celebrities in images and provides confidence scores.' The central image shows a close-up of a man's face with a white bounding box around it. Below the image are sections for 'Choose a sample image' and 'Use your own image', each with an 'Upload' button and a 'Use image URL' field. On the right side, there is a 'Results' section titled 'Done with the demo? Learn more'. It displays a single result for the celebrity 'Burak Deniz' with a 'Match confidence' of 100%.

## 4. Text in Image



## Screenshots needed for EC2 & S3

### 1.Installing aws-sdk

```
ec2-user@ip-172-31-9-63:/var/www/html/face
[ec2-user@ip-172-31-9-63 face]$ free -m
              total        used        free      shared  buff/cache   available
Mem:           983          66          537           0          379          777
Swap:            0           0           0

[ec2-user@ip-172-31-9-63 face]$ sudo /bin/dd if=/dev/zero of=/var/swap.1 bs=1M c
ount=1024
sudo /sbin/mkswap /var/swap1024+0 records in
1024+0 records out
1073741824 bytes (1.1 GB) copied, 13.3761 s, 80.3 MB/s
[ec2-user@ip-172-31-9-63 face]$ sudo /sbin/mkswap /var/swap.1
mkswap: /var/swap.1: insecure permissions 0644, 0600 suggested.
Setting up swappiness Version 1, size = 1024 MiB (1073737728 bytes)
no label, UUID=9c997fbb-263d-462d-8e8c-e4d9e68d7c63
[ec2-user@ip-172-31-9-63 face]$ sudo /sbin/swapon /var/swap.1
swapon: /var/swap.1: insecure permissions 0644, 0600 suggested.
[ec2-user@ip-172-31-9-63 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/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-172-31-9-63 face]$ sudo php -d memory_limit=-1 ~/composer.phar require aws/aws-sdk-php
Using version ^2.8 for aws/aws-sdk-php
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Nothing to install or update
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/guzzle instead.
Generating autoload files
[ec2-user@ip-172-31-9-63 face]$
```

## 2. Installing php

```
ec2-user@ip-172-31-9-63:~$ sudo yum install php
Complete!
[ec2-user@ip-172-31-9-63 ~]$ sudo yum install php
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
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: ph
p-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
--> Running transaction check
--> Package php-cli.x86_64 0:5.4.16-46.amzn2.0.2 will be installed
--> Package php-common.x86_64 0:5.4.16-46.amzn2.0.2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package             Arch           Version           Repository        Size
=====
Installing:
php                 x86_64         5.4.16-46.amzn2.0.2   amzn2-core       1.4 M
Installing for dependencies:
php-cli             x86_64         5.4.16-46.amzn2.0.2   amzn2-core       2.8 M
php-common          x86_64         5.4.16-46.amzn2.0.2   amzn2-core       563 k
=====
Transaction Summary
=====
Install 1 Package (+2 Dependent packages)

Total download size: 4.7 M
Installed size: 17 M
Is this ok [y/d/N]: y
Downloading packages:
(1/3): php-5.4.16-46.amzn2.0.2.x86_64.rpm | 1.4 MB 00:00
(2/3): php-cli-5.4.16-46.amzn2.0.2.x86_64.rpm | 2.8 MB 00:00
(3/3): php-common-5.4.16-46.amzn2.0.2.x86_64.rpm | 563 KB 00:00
-----
Total                                     21 MB/s | 4.7 MB 00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : php-common-5.4.16-46.amzn2.0.2.x86_64 1/3
```

## 3. index.php file code

```
ec2-user@ip-172-31-9-63:/var/www/html/face$ cat index.php
$keyname = 'groupshot.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;

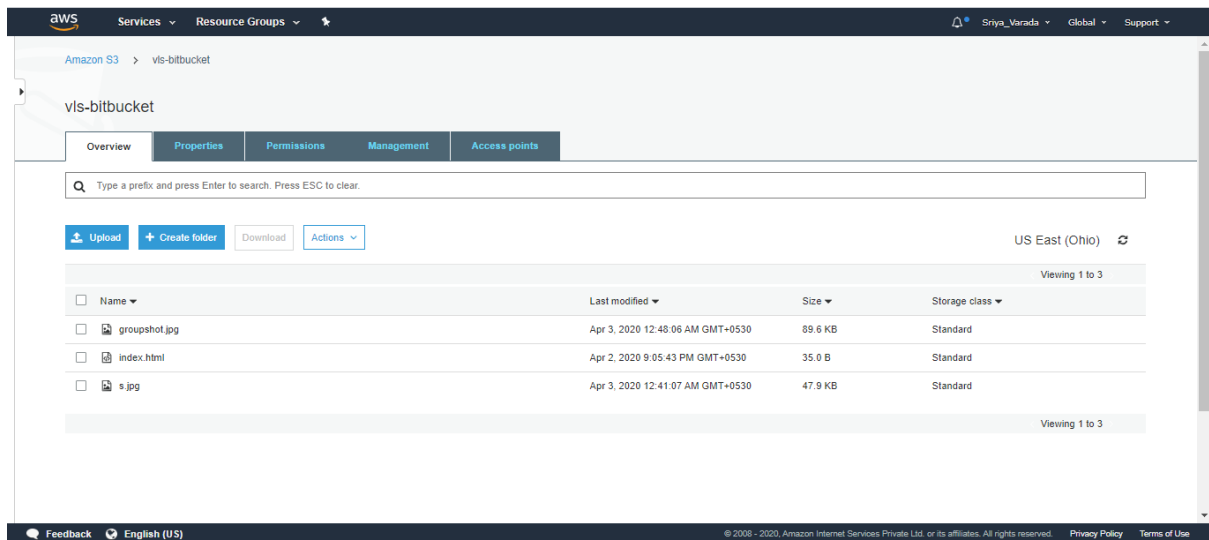
        $rekognition = new RekognitionClient([
            'region' => 'us-east-2',
            'version' => 'latest',
        ]);

        $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;
}
```

## 4. Upload success screenshot

```
ec2-user@ip-172-31-9-63:/var/www/html/face$ ls
composer.json  composer.lock  groupshot.jpg  s.jpg  vendor
ec2-user@ip-172-31-9-63:/var/www/html/face$ sudo vim index.php
ec2-user@ip-172-31-9-63:/var/www/html/face$ sudo php index.php
Image upload done... Here is the URL: https://v1a-btrbucket.s3.us-east-2.amazonaws.com/groupshot.jpgTotally there are 11 facesec2-user@ip-172-31-9-63:/var/www/html/face$
```



## Screenshots needed for EC2 & Rekognition

### 1.Face Detect success screenshot

