

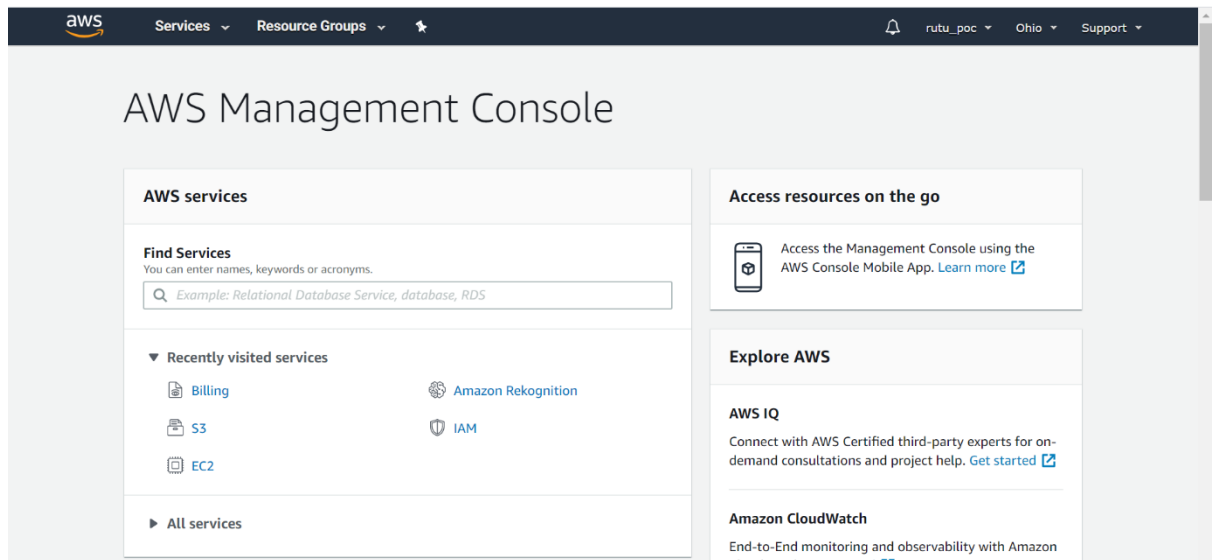
Name: Patel Rutu Manish

University: VIT, Vellore

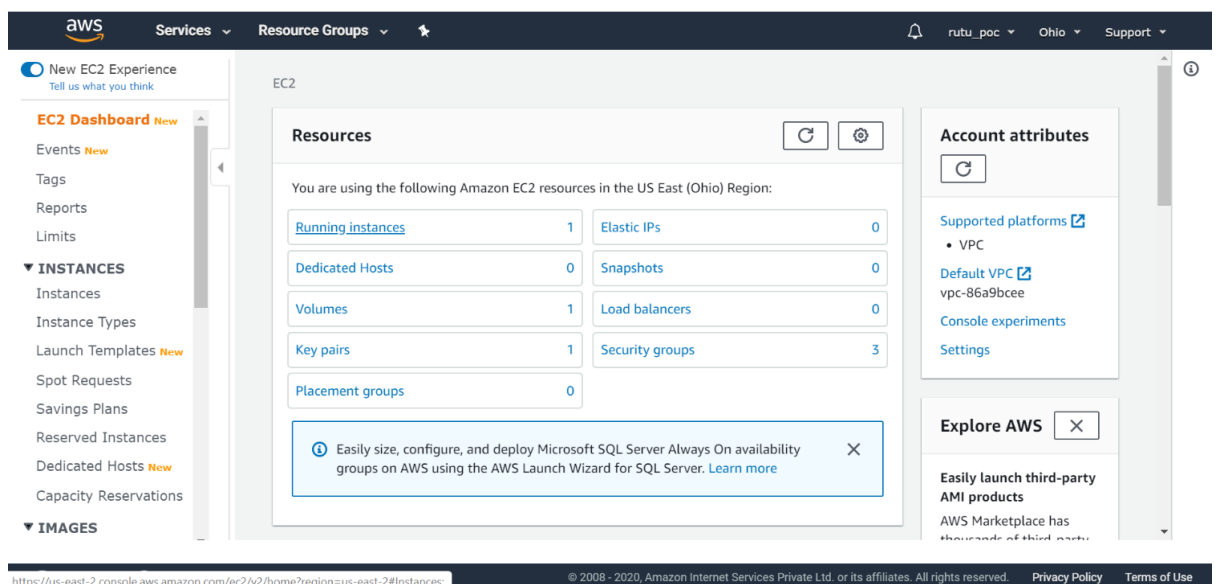
Course: MTech Computer Science

Screenshots for dashboards

1. AWS Login screen with username



2. EC2 Dashboard



3. S3 Dashboard

The screenshot shows the Amazon S3 console. The left sidebar contains the 'Amazon S3' header and a list of links: 'Buckets', 'Batch operations', 'Access analyzer for S3', 'Block public access (account settings)', and 'Feature spotlight' with a '2' badge. The main content area is titled 'Amazon S3' and shows 'Buckets (1)'. Above the bucket list are buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'. A search bar with the placeholder 'Find bucket by name' is present. Below the search bar is a table with one bucket:

	Name	Region	Access	Bucket created
<input type="radio"/>	myimagerecognitionapp	US East (Ohio) us-east-2	Objects can be public	2020-04-01T13:30:53.000Z

The footer includes 'Feedback', 'English (US)', and copyright information: '© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use'.

4. Rekognition Dashboard

The screenshot shows the Amazon Rekognition console. The left sidebar has the 'Amazon Rekognition' header and links for 'Custom Labels' (with a 'New' badge), 'Use Custom Labels', 'Demos', 'Object and scene detection', 'Image moderation', 'Facial analysis', 'Celebrity recognition', 'Face comparison', 'Text in image', 'Video Demos', 'Video analysis', and 'Metrics'. The main content area features a large hero section with the title 'Amazon Rekognition' and the subtitle 'Deep learning-based visual analysis service'. It includes the text 'Search, verify, and organize millions of images and videos' and buttons for 'Try Demo' and 'Download SDKs'. Below the hero section are three columns with icons and text: 'Easily Integrate Powerful Visual Analysis into Your' (with a stack of layers icon), 'Continuously Learning' (with a neural network icon) and 'Amazon Rekognition is designed to use', and 'Integrated with AWS Services' (with a puzzle pieces icon). The footer includes 'Feedback', 'English (US)', and copyright information: '© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use'.

Screenshots for EC2

1. Choosing an AMI

aws

Services

Resource Groups

rutu_poc

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 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

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: ebs Virtualization type: hvm ENA 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.

64-bit (x86)

Select

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

rutu_poc

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

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

Cancel

Previous

Review and Launch

Next: Configure Instance Details

Feedback

English (US)

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

3. Adding Storage

aws

Services

Resource Groups

★

🔔

rutu_poc

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	<input checked="" type="checkbox"/>	Not Encrypt

Add New Volume

Free tier eligible customers can get up to 30 GiB 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](#)

4. Configuring Security Group

aws

Services

Resource Groups

★

🔔

rutu_poc

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

Security group name	Security group ID	Description	VPC ID
launch-wizard-2	sg-03a7a9fe68ac9664a	launch-wizard-2 created 2020-03-31T21:47:56.888+05:30	vpc-86a9bcee
Owner	Inbound rules count	Outbound rules count	
131269167822	2 Permission entries	1 Permission entry	

Inbound rules

Outbound rules

Tags

Inbound rules

Edit inbound rules

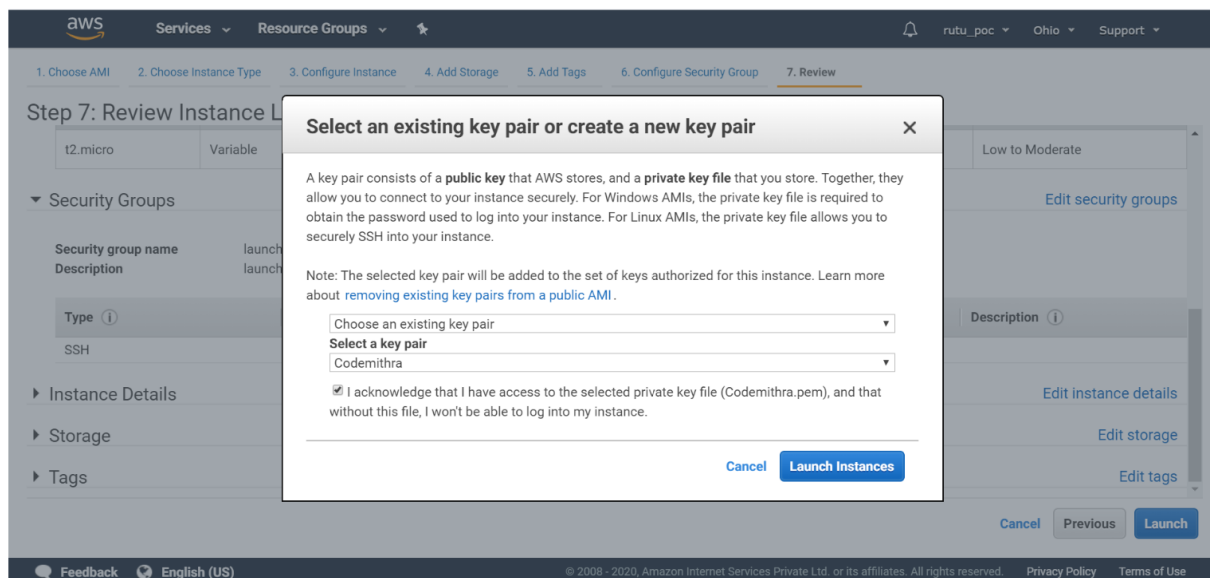
Type	Protocol	Port range	Source	Description - optional
HTTP	TCP	80	::/0	-
SSH	TCP	22	0.0.0.0/0	-

Feedback

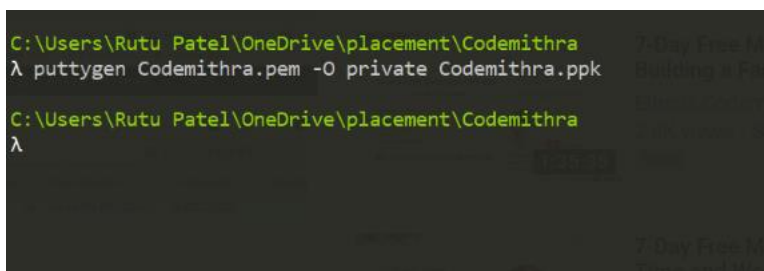
English (US)

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

5. Key Pair Download



6. PuTTYgen conversion from pem to ppk

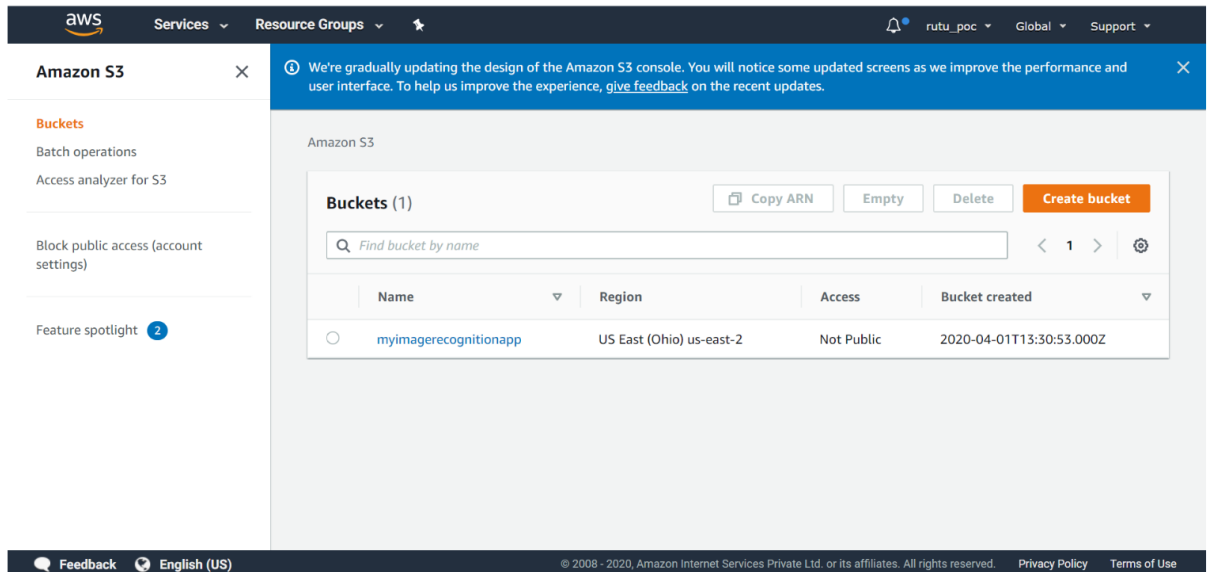


7. Logged in EC2 black screen

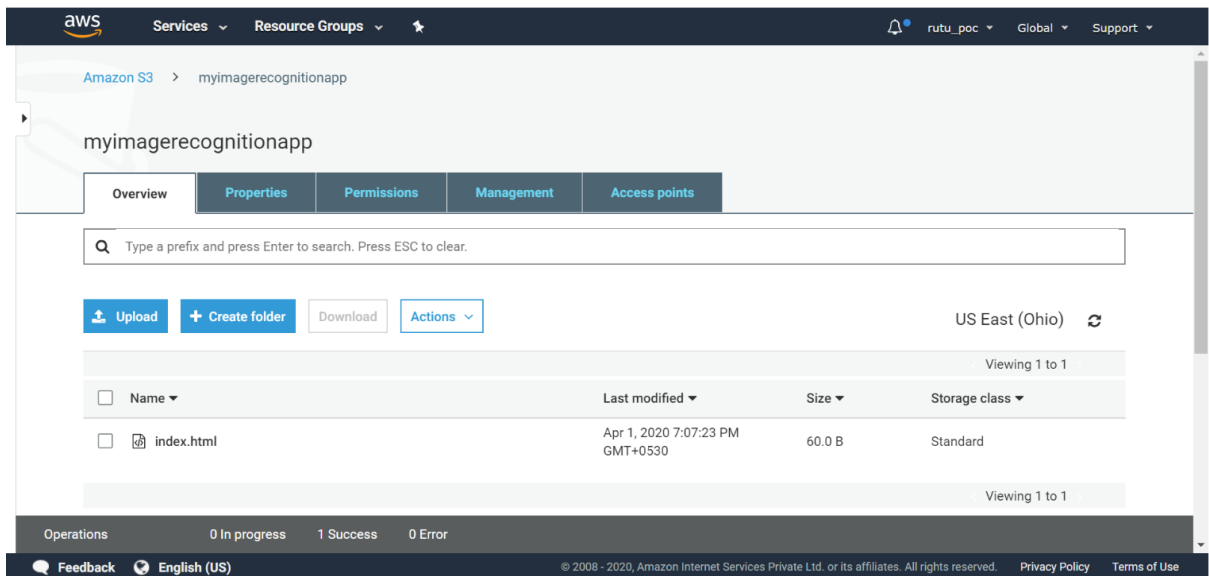


Screenshots for S3

1. Creating a bucket



2. Uploading an Object



3. Enabling Static Website

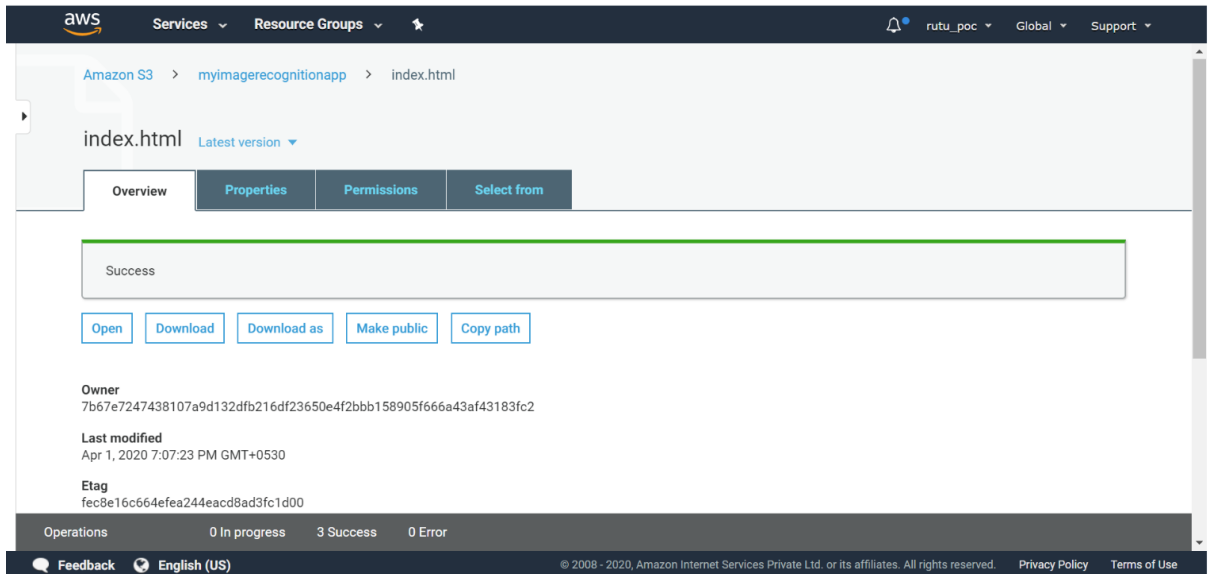
The screenshot shows the 'Static website hosting' configuration page in the AWS S3 console. The endpoint is `http://myimagerecognitionapp.s3-website.us-east-2.amazonaws.com`. The 'Index document' is `index.html` and the 'Error document' is `error.html`. The 'Redirect requests' option is selected. A 'Disabled' status box is visible on the right. The bottom of the console shows the footer with '© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.'

4. Making the Object Public

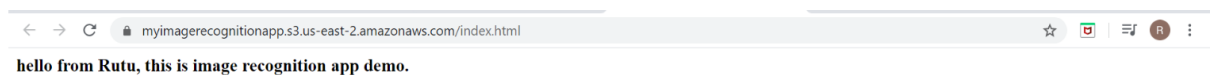
Making bucket public

The screenshot shows the 'Block public access (bucket settings)' page in the AWS S3 console. The page explains that public access is granted through ACLs, bucket policies, or access point policies. The 'Block all public access' checkbox is checked. Below it, four sub-settings are listed, all of which are unchecked: '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)', 'Block public access to buckets and objects granted through new public bucket or access point policies', and 'Block public and cross-account access to buckets and objects through any public bucket or access point policies'. The bottom of the console shows the footer with '© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.'

Making object public

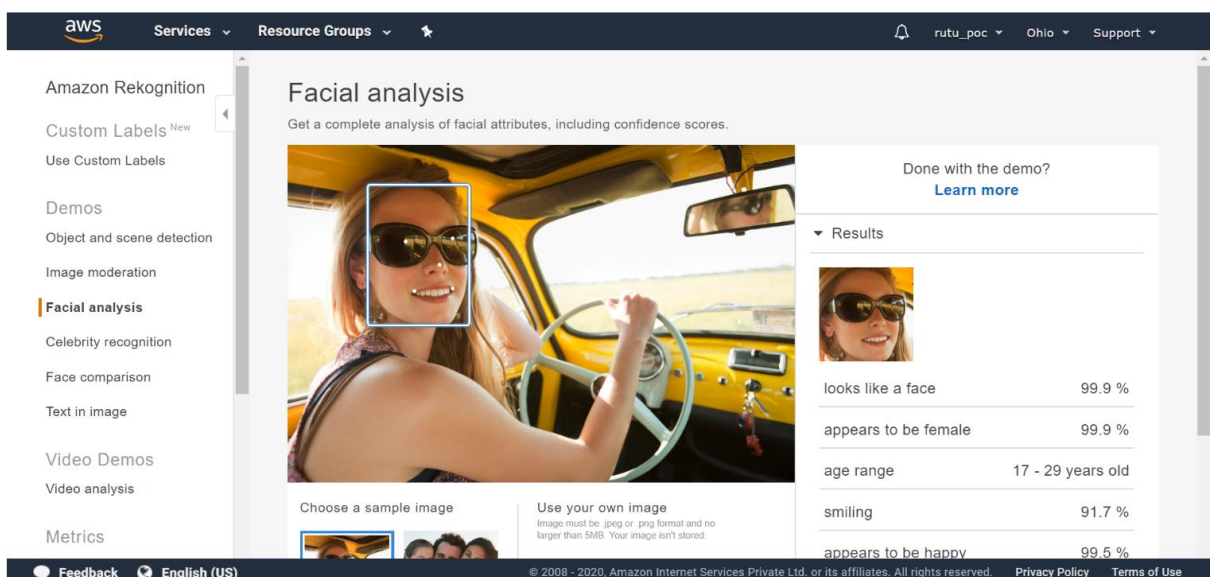


5. Checking the S3 link on the browser



Screenshots needed for Rekognition

1. Face Detect



2. Face Compare

The screenshot shows the AWS Face Comparison demo interface. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and user information. The left sidebar lists various Amazon Rekognition services, with 'Face comparison' highlighted. The main content area is titled 'Face comparison' and includes a description: 'Compare faces to see how closely they match based on a similarity percentage.' Below this, there are two image upload sections: 'Reference face' and 'Comparison faces'. The 'Reference face' section shows a sample image of a young girl. The 'Comparison faces' section shows a sample image of three young girls. Below these sections are two buttons: 'Choose a sample image' and 'Choose a sample image'. To the right, there is a 'Results' section. It shows a comparison of the two faces with an equals sign and a similarity percentage of 99.8%. Below this, there is a comparison with a different face, showing a not-equals sign. At the bottom, there is a footer with 'Feedback', 'English (US)', and copyright information.

aws Services Resource Groups rutu_poc 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

Face comparison

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

Reference face

Comparison faces

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

▼ Results

Similarity 99.8 %

Choose a sample image

Choose a sample image

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

3. Celebrity Recognition

The screenshot shows the AWS Celebrity Recognition demo interface. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and user information. The left sidebar lists various Amazon Rekognition services, with 'Celebrity recognition' highlighted. The main content area is titled 'Celebrity recognition' and includes a description: 'Rekognition automatically recognizes celebrities in images and provides confidence scores.' Below this, there is a large image of Jeff Bezos with a blue bounding box around his face. Below the image are two buttons: 'Choose a sample image' and 'Use your own image'. To the right, there is a 'Results' section. It shows a comparison of the image with a celebrity, with a match confidence of 100%. Below this, there is a 'Request' and 'Response' section. At the bottom, there is a footer with 'Feedback', 'English (US)', and copyright information.

aws Services Resource Groups rutu_poc 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

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

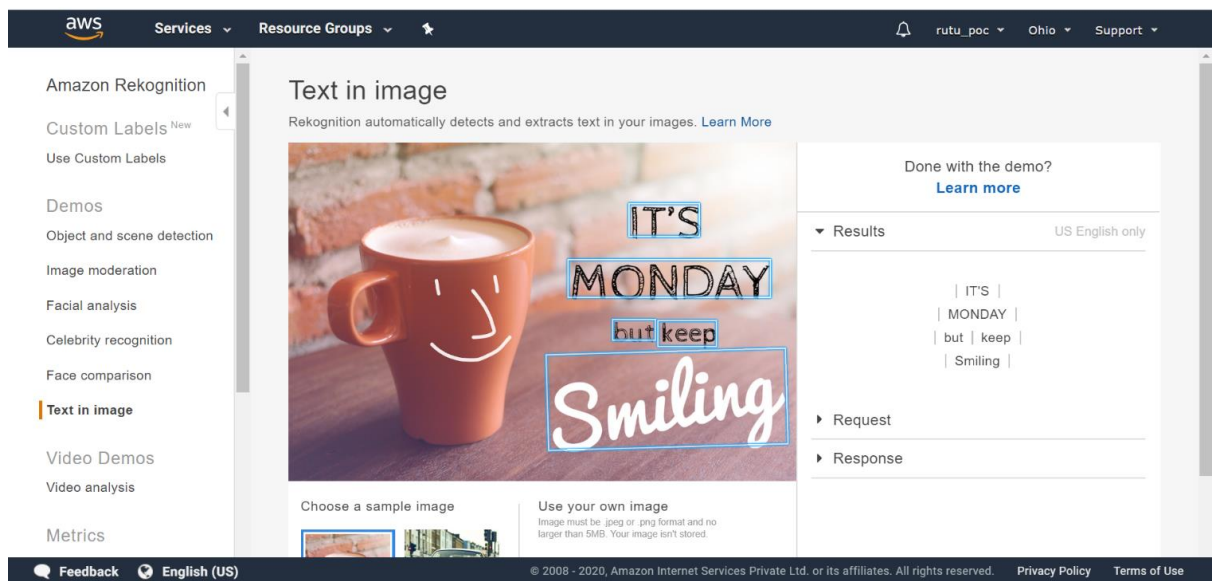
Match confidence 100 %

► 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



Scenshots needed for EC2 and S3

1. Installing aws-sdk

```
[ec2-user@ip-172-31-24-49 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
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
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-172-31-24-49 face]$
```

2. Installing php

```
=====
Install 1 Package (+3 Dependent packages)

Total download size: 4.7 M
Installed size: 17 M
Is this ok [y/d/N]: y
Downloading packages:
(1/4): libzip010-compat-0.10.1-9.amzn2.0.5.x86_64.rpm      | 30 kB  00:00
(2/4): php-5.4.16-46.amzn2.0.2.x86_64.rpm                | 1.4 MB  00:00
(3/4): php-common-5.4.16-46.amzn2.0.2.x86_64.rpm         | 563 kB  00:00
(4/4): php-cli-5.4.16-46.amzn2.0.2.x86_64.rpm            | 2.8 MB  00:00
-----
Total                                                    20 MB/s | 4.7 MB  00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : libzip010-compat-0.10.1-9.amzn2.0.5.x86_64 1/4
  Installing : php-common-5.4.16-46.amzn2.0.2.x86_64      2/4
  Installing : php-cli-5.4.16-46.amzn2.0.2.x86_64         3/4
  Installing : php-5.4.16-46.amzn2.0.2.x86_64            4/4
  Verifying   : php-5.4.16-46.amzn2.0.2.x86_64            1/4
  Verifying   : libzip010-compat-0.10.1-9.amzn2.0.5.x86_64 2/4
  Verifying   : php-cli-5.4.16-46.amzn2.0.2.x86_64       3/4
  Verifying   : php-common-5.4.16-46.amzn2.0.2.x86_64    4/4

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

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

Complete!
[ec2-user@ip-172-31-24-49 ~]$ curl -sS https://getcomposer.org/installer | php
All settings correct for using Composer
Downloading...

Composer (version 1.10.1) successfully installed to: /home/ec2-user/composer.phar
Use it: php composer.phar

[ec2-user@ip-172-31-24-49 ~]$
```

3. index.php file code

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

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

$bucket = 'myimagerecognitionapp';
$keyname = 's.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'
    ]);

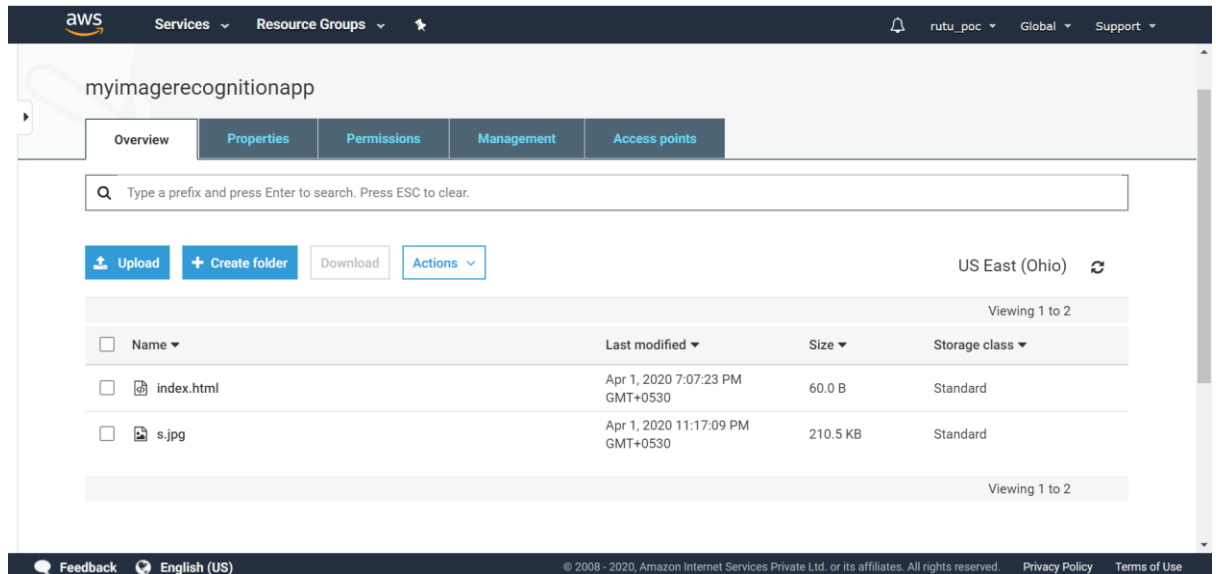
    // Print the URL to the object.
    $imageUrl = $result['ObjectURL'];
    if($imageUrl) {
        echo "Image upload done... Here is the URL: " . $imageUrl;

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

        $result = $rekognition->detectFaces([
            'Attributes' => ['DEFAULT'],
            'Image'      => [
                'S3Object' => [
                    'Bucket' => $bucket,
                    'Name'   => $keyname,
                    'Key'    => $keyname,
                ],
            ],
        ]);
    }
}
```

4. Upload success screenshot

```
[ec2-user@ip-172-31-24-49 face]$ sudo vim index.php
[ec2-user@ip-172-31-24-49 face]$ sudo php index.php
Image upload done... Here is the URL: https://myimagerecognitionapp.s3.us-east-2.amazonaws.com/s.jpg
```



Screenshots needed for EC2 and Rekognition

1. Face Detect success screenshot

```
[ec2-user@ip-172-31-24-49 var]$ cd www/html/face
[ec2-user@ip-172-31-24-49 face]$ sudo php index.php
PHP Catchable fatal error: Argument 1 passed to Aws\Common\Client\AbstractClient::__construct() must be an instance of Aws\Common\Credentials\CredentialsInterface, array given, called in /var/www/html/face/index.php on line 46 and defined in /var/www/html/face/vendor/aws/aws-sdk-php/src/Aws/Common/Client/AbstractClient.php on line 73
[ec2-user@ip-172-31-24-49 face]$
```