

FACE DETECTION APP ON AWS

NAME: B.Ramakrishna

MOBILE:9398576582

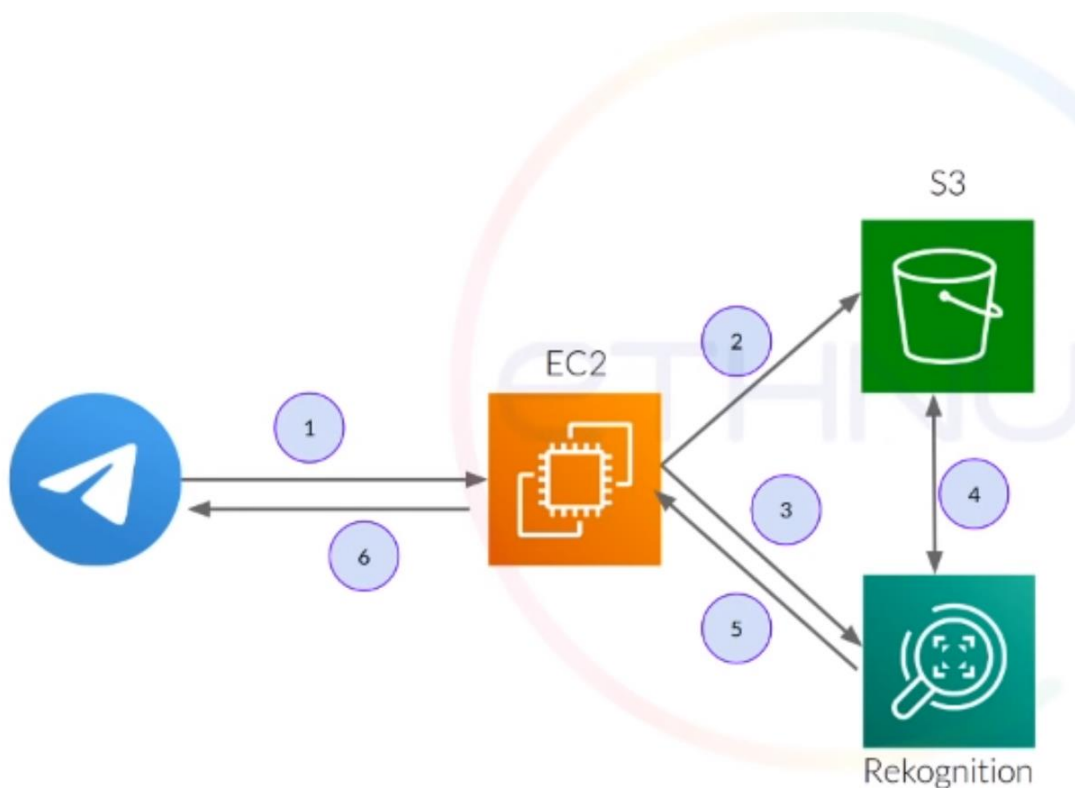
EMAIL: ramakrishna19992107@gmail.com

OBJECTIVE:

The main objective of creating this face detection application is to find the image and describe about the image using the AWS.

DESCRIPTION:

To find the image, we use four components i.e., Telegram Bot, EC2, S3 and Amazon Rekognition. This are the steps that are followed that are given below:

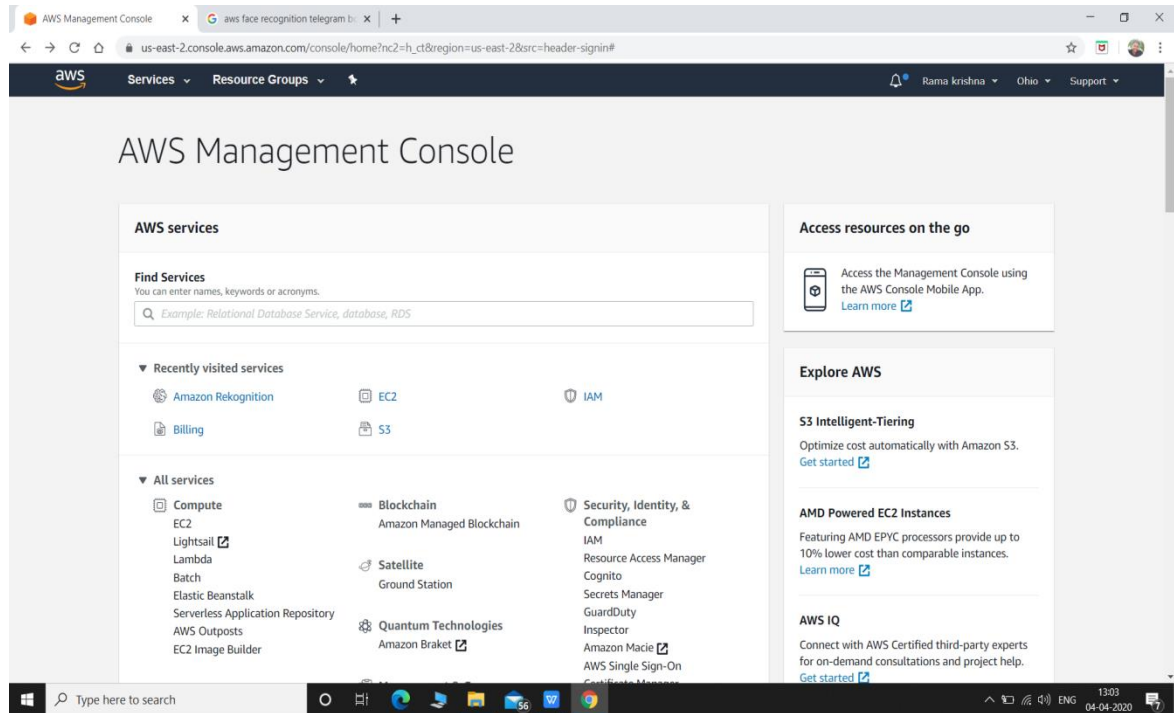


First image is uploaded in the Telegram Bot. Then the image is taken by EC2, it stores the image in the S3 (in buckets). Then EC2 sends the image to Amazon Rekognition, and it detects the image and gives the details or description of that image, which will be sent to EC2 and then sent to Telegram Bot.

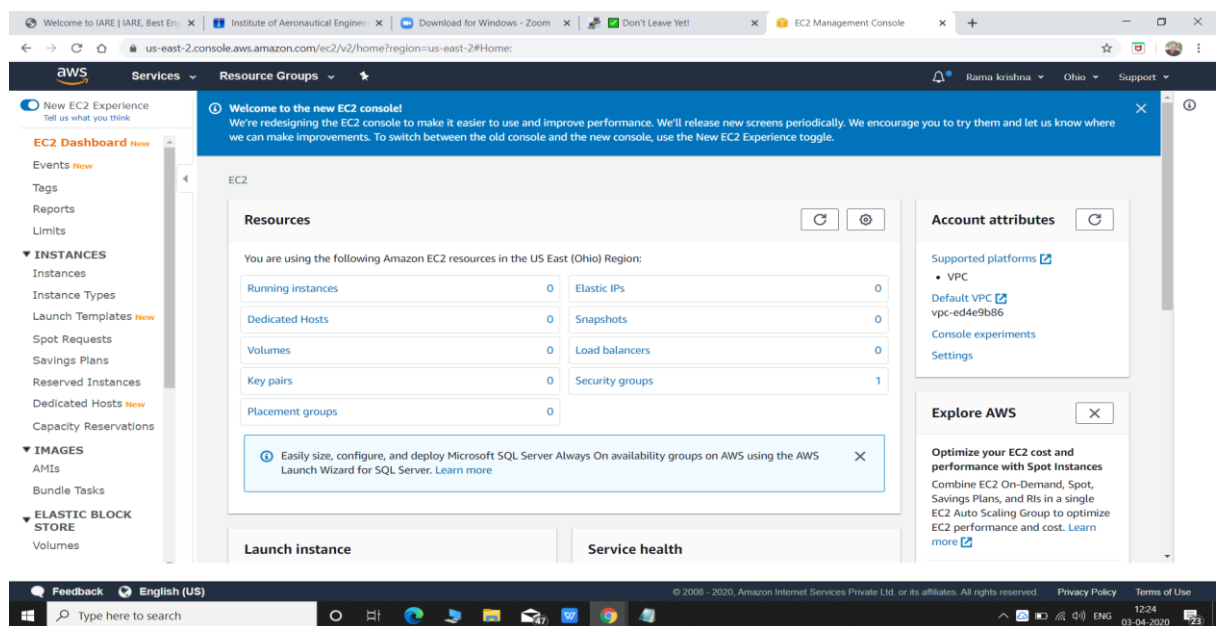
SCREENSHOTS:

● AWS:

1. AWS Login screen with Username:



2. EC2 Dashboard:



3. S3 Dashboard:

The screenshot displays the Amazon S3 console interface. At the top, a navigation bar includes the AWS logo, 'Services', and 'Resource Groups'. A left-hand sidebar lists various S3 features like 'Buckets', 'Batch operations', and 'Access analyzer for S3'. The main content area shows a green notification banner stating 'Successfully created bucket ramaaws'. Below this, a 'Buckets (1)' table lists the newly created bucket 'ramaaws' in the 'US East (Ohio) us-east-2' region, with public access enabled and a creation timestamp of '2020-04-03T07:04:45.000Z'. The table has columns for Name, Region, Access, and Bucket created. At the bottom of the console, a Windows taskbar is visible with the search bar and system clock showing 12:34 on 03-04-2020.

Name	Region	Access	Bucket created
ramaaws	US East (Ohio) us-east-2	Objects can be public	2020-04-03T07:04:45.000Z

4. Rekognition Dashboard:

The screenshot shows the Amazon Rekognition console home page. The header features the AWS logo and navigation links. A left sidebar contains a list of services and demos, including 'Custom Labels', 'Image moderation', and 'Video analysis'. The main area has a large hero section with the title 'Amazon Rekognition' and a description: 'Deep learning-based visual analysis service. Search, verify, and organize millions of images and videos'. Below the hero section, there are three columns of information: 'Easily Integrate Powerful Visual Analysis into Your App', 'Continuously Learning', and 'Integrated with AWS Services'. The Windows taskbar at the bottom shows the search bar and system clock at 14:34 on 03-04-2020.

● EC2:

5. Choosing an AMI:

Welcome to IARE | IARE, Best En... | Don't Leave Yet! | Launch instance wizard | EC2 Ma... | +

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

aws Services Resource Groups Rama Krishna 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 1

Amazon Linux 2 AMI (HVM, SSD Volume Type) - ami-0e01ce4ee18447327 (64-bit x86) / ami-03201f374ab6a26e (64-bit Arm) Select

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)

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

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

64-bit (x86)

Red Hat Enterprise Linux 8 (HVM, SSD Volume Type) - ami-0520e698dd500b1d1 (64-bit x86) / ami-0099847d600887c9f (64-bit Arm) Select

Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type

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

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

SLES Linux Enterprise Server 15 SP1 (HVM, SSD Volume Type) - ami-04e5b451ce148295 (64-bit x86) / ami-02e73002f19019474 (64-bit Arm) Select

Feedback English (US)

Type here to search

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

12:25 03-04-2020

6. Choosing an Instance Type:

Welcome to IARE | IARE, Best En... | Don't Leave Yet! | Launch instance wizard | EC2 Ma... | +

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

aws Services Resource Groups Rama Krishna Ohio Support

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

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, 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 types Current generation Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<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

Cancel Previous Review and Launch Next: Configure Instance Details

Feedback English (US)

Type here to search

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

12:26 03-04-2020

7. Adding Storage:

Welcome to IARE | IARE, Best En... x | Don't Leave Yet! x | Launch instance wizard | EC2 M... x

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

Services Resource Groups

Rama Krishna 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 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)

Type here to search

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

1227 03-04-2020

8. Configuring Security Group:

Welcome to IARE | IARE, Best En... x | Don't Leave Yet! x | Launch instance wizard | EC2 M... x

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

Services Resource Groups

Rama Krishna Ohio Support

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

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group ☐ Select an existing security group

Security group name: launch-wizard-1

Description: launch-wizard-1 created 2020-04-03T12:27:53.091+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
HTTP	TCP	80	Anywhere 0.0.0.0/0, ::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**

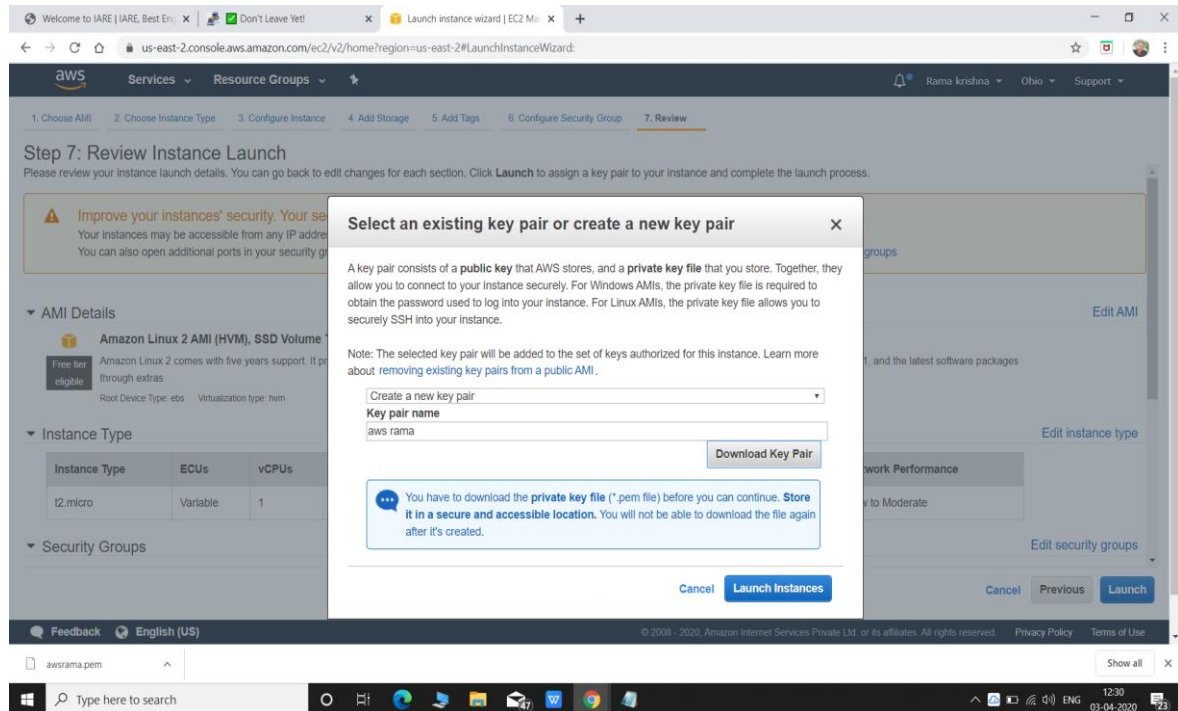
Feedback English (US)

Type here to search

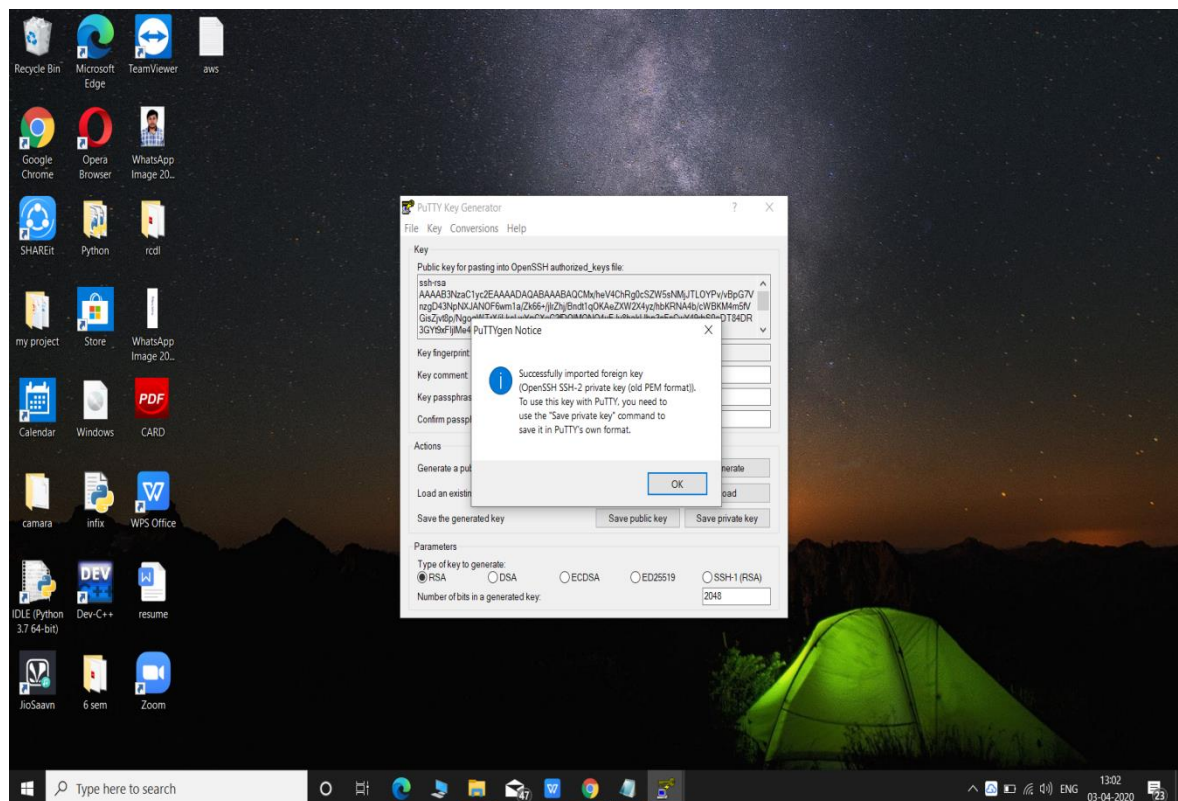
© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

1228 03-04-2020

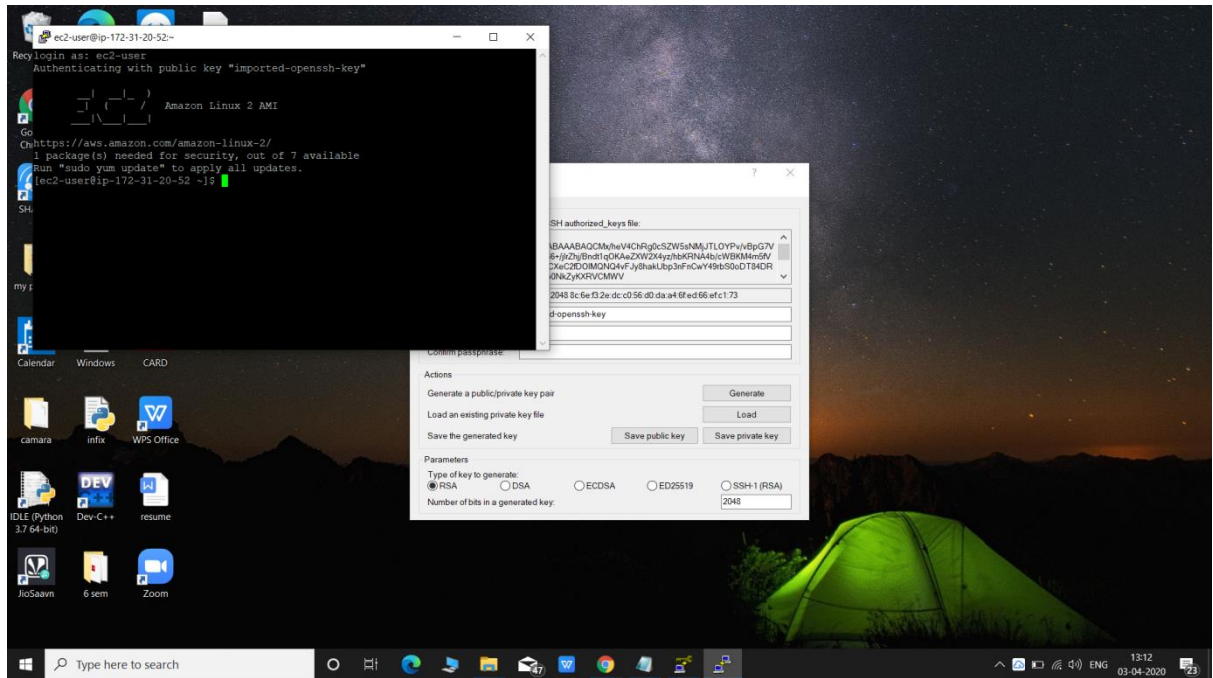
9. Key Pair Download:



10. PuTTYgen conversion from pem to ppk :

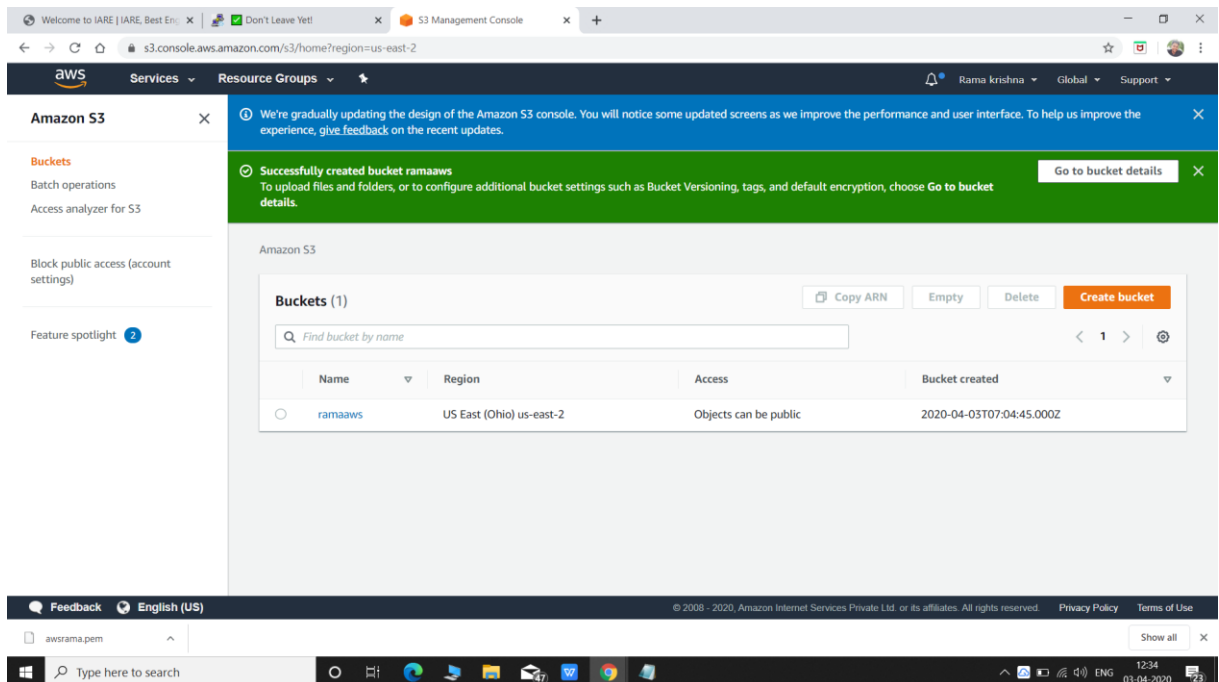


11. Logged in EC2 black screen:

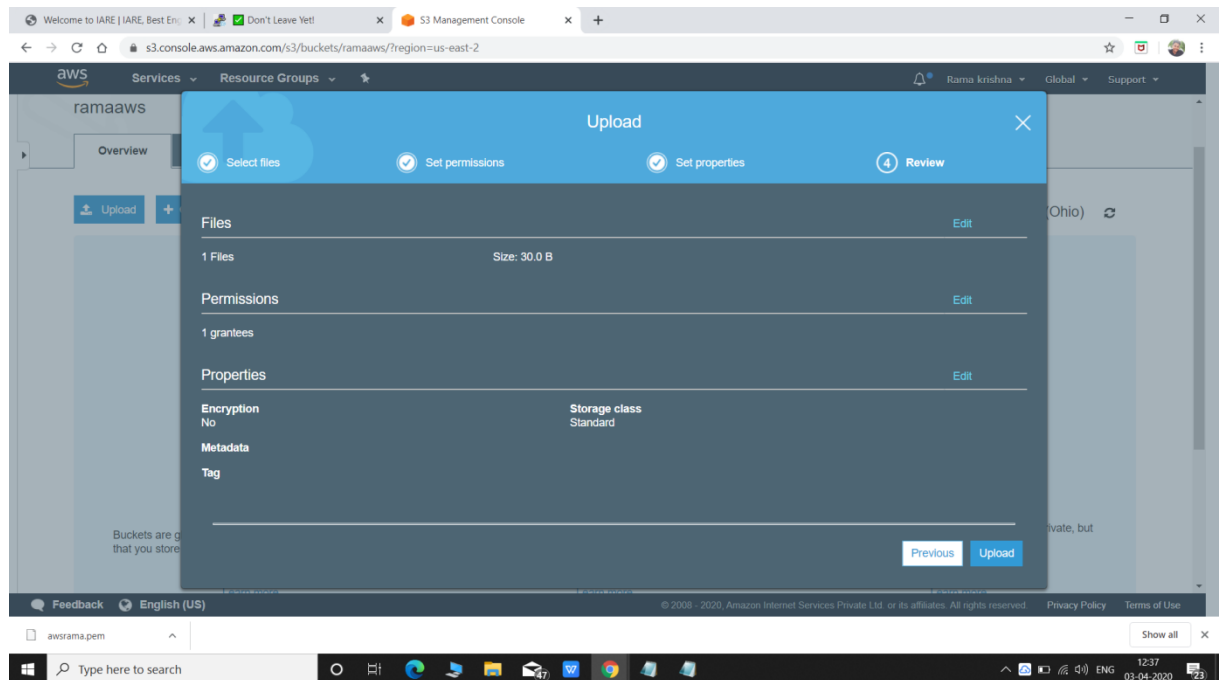


● S3:

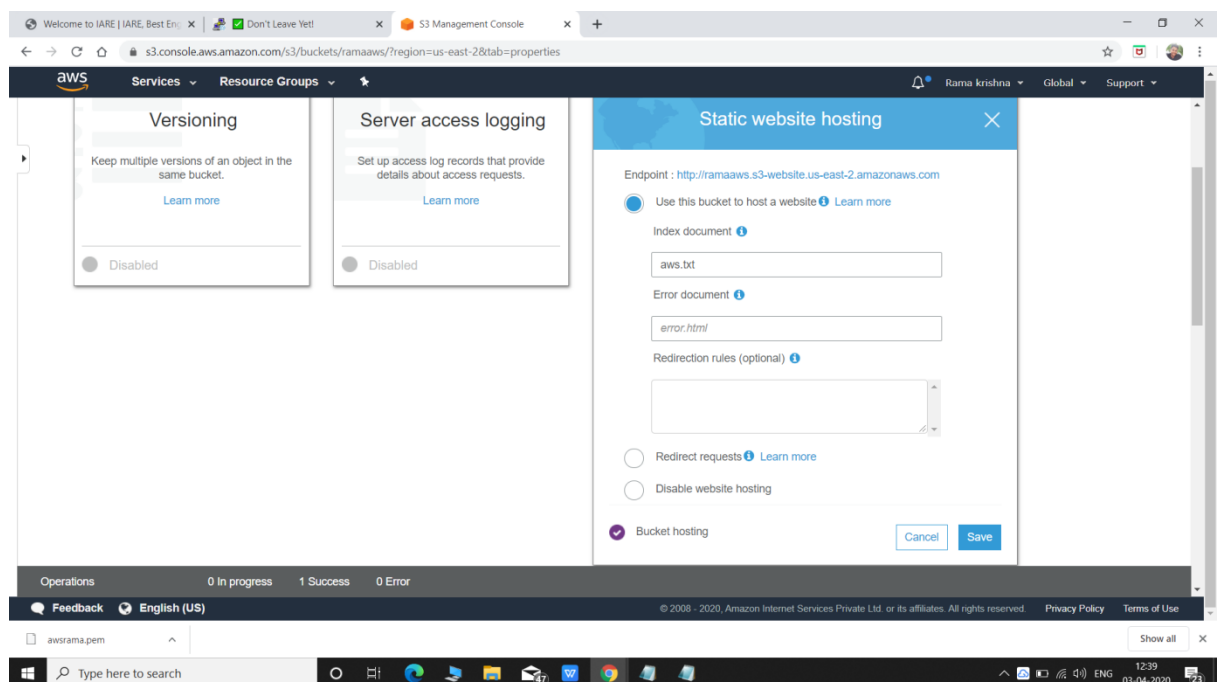
12. Creating a Bucket:



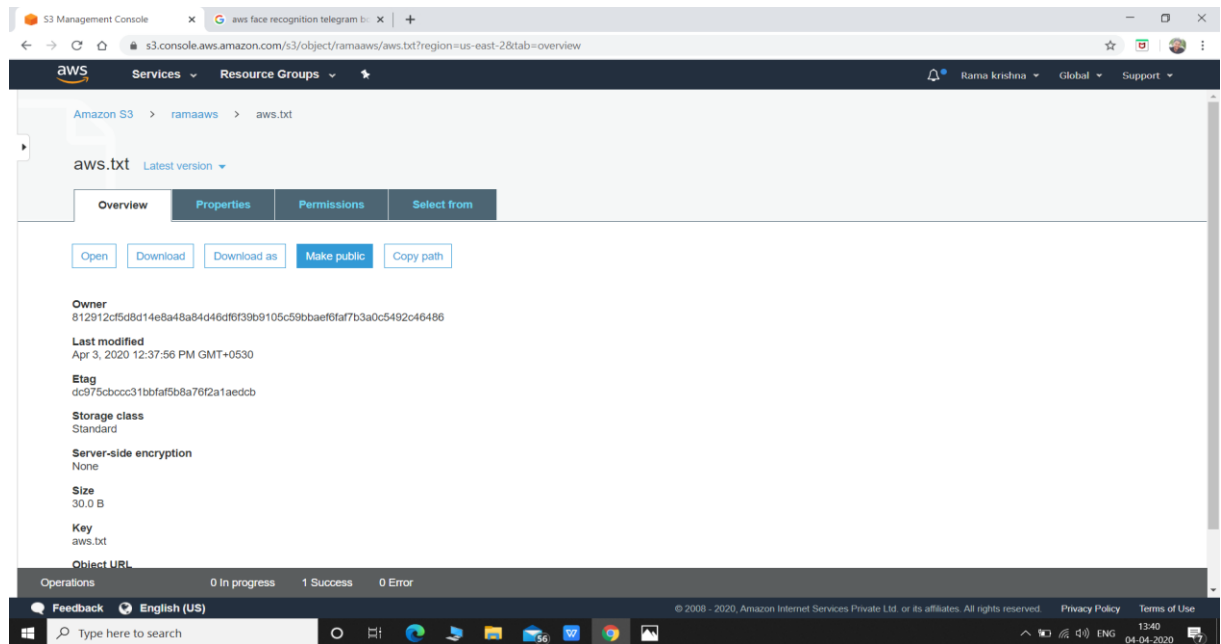
13. Uploading an Object:



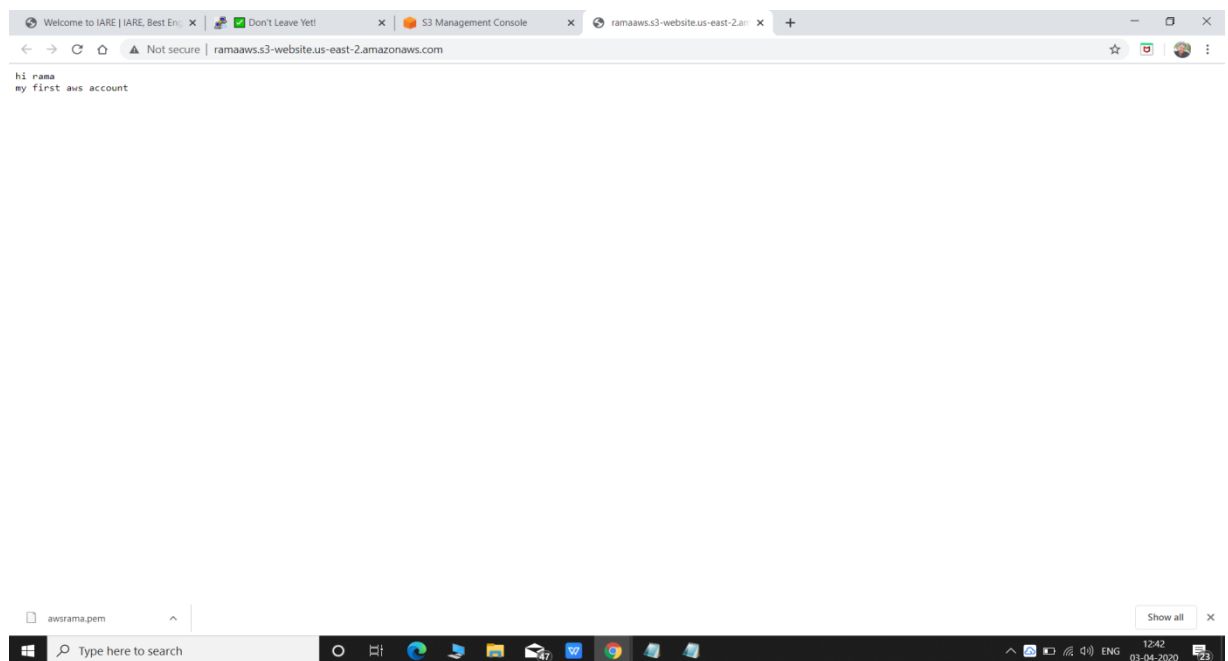
14. Enabling Static Website:



15. Making the Object Public:



16. Checking the S3 link on the browser:



- **Rekognition:**

17. Face Detect:

The screenshot shows the AWS Rekognition console interface. The left sidebar contains navigation links: Amazon Rekognition, Custom Labels, Use Custom Labels, Demos, Object and scene detection, Image moderation, Facial analysis, Celebrity recognition, Face comparison, Text in image, Video Demos, Video analysis, Metrics, and Additional Resources. The main content area displays a demo for 'Object and scene detection' with a large image of a cricket player (Virat Kohli) and a bounding box. Below the image are options to 'Choose a sample image' or 'Use your own image' with an 'Upload' button. The right panel shows the 'Results' section with a table of detected objects and their confidence percentages.

Object	Confidence
Person	99 %
Human	99 %
Sport	92.7 %
Sports	92.7 %
People	80.3 %
Face	76.9 %

18. Face Compare:

The screenshot shows the AWS Rekognition console interface for the 'Face comparison' demo. The left sidebar contains navigation links: Use Custom Labels, Demos, Object and scene detection, Image moderation, Facial analysis, Celebrity recognition, Face comparison, Text in image, Video Demos, Video analysis, Metrics, and Additional Resources. The main content area displays a demo for 'Face comparison' with a 'Reference face' and 'Comparison faces' section. Below these are options to 'Choose a sample image'. The right panel shows the 'Results' section with a table of similarity percentages for different face comparisons.

Comparison	Similarity
Reference face vs. Comparison face 1	89.1 %
Reference face vs. Comparison face 2	Similarity
Reference face vs. Comparison face 3	Similarity

19. Celebrity Recognition:


Rekognition Console

us-east-2.console.aws.amazon.com/rekognition/home?region=us-east-2#/celebrity-detection

Celebrity recognition
Rekognition automatically recognizes celebrities in images and provides confidence scores.

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

▼ Results

 **Pawan Kalyan**
[Learn More](#)

Match confidence 100 %

► 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 stored.

[Upload](#) or drag and drop

Use image URL [Go](#)

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

20. Text in Image:

Welcome to IARE x Don't Leave Y x Rekognition Cons x ramaaws.s3-webs x (no subject) - ram x PSPK - Google Se x virat-kohli.png (8 x SLOGANS - Goog x

us-east-2.console.aws.amazon.com/rekognition/home?region=us-east-2#/text-detection

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

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

▼ Results US English only

BUSINESS | SLOGANS | 101: |
HOW | TO | WRITE | A |
ROCK-SOLID | TAGLINE |

► 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 stored.

[Upload](#) or drag and drop

Use image URL [Go](#)

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

● EC2 & S3:

21. Installing aws-sdk:

```
ec2-user@ip-172-31-20-52:/var/www/html/face
vendor virat-kohli.png
[ec2-user@ip-172-31-20-52 face]$ sudo vim index.php
[ec2-user@ip-172-31-20-52 face]$ sudo vim index.php
[ec2-user@ip-172-31-20-52 face]$ sudo php index.php
[ec2-user@ip-172-31-20-52 face]$ curl -s https://getcomposer.org/installer | ph
R
All settings correct for using Composer
The installation directory "/var/www/html/face" is not writable
[ec2-user@ip-172-31-20-52 face]$ sudo php -d memory_limit=-1 ~/composer.phar req
uire 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 c
aching, request and response caching, and credentials caching)
aws/aws-sdk-php suggests installing monolog/monolog (Adds support for logging HT
TP requests and responses)
aws/aws-sdk-php suggests installing symfony/yaml (Eases the ability to write man
ifests for creating jobs in AWS Import/Export)
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/gu
zle instead.
Writing lock file
Generating autoload files
[ec2-user@ip-172-31-20-52 face]$
[ec2-user@ip-172-31-20-52 face]$ ls
composer.json composer.lock index.php vendor virat-kohli.png
[ec2-user@ip-172-31-20-52 face]$ sudo php index.php
Error retrieving credentials from the instance profile metadata server. When you
are not running inside of Amazon EC2, you must provide your AWS access key ID a
nd secret access key in the "key" and "secret" options when creating a client or
provide an instantiated Aws\Common\Credentials\CredentialsInterface object. (Cl
ient error response
[status code] 404
[reason phrase] Not Found
[url] http://169.254.169.254/latest/meta-data/iam/security-credentials/)
[ec2-user@ip-172-31-20-52 face]$ sudo php index.php
Image upload done... Here is the URL: https://ramaws.s3.us-east-2.amazonaws.com
[ec2-user@ip-172-31-20-52 face]$
```

22. Installing php:

```
ec2-user@ip-172-31-20-52:/var/www/html/face
g to composer.json, and updates the composer.lock file.
update Upgrades your dependencies to the latest version according
to composer.json, and updates the composer.lock file.
upgrade Upgrades your dependencies to the latest version according
to composer.json, and updates the composer.lock file.
validate Validates a composer.json and composer.lock.
why Shows which packages cause the given package to be instal
why-not Shows which packages prevent the given package from being
installed.
[ec2-user@ip-172-31-20-52 ~]$ cd /var/www/html
[ec2-user@ip-172-31-20-52 html]$ mkdir face
mkdir: cannot create directory 'face': Permission denied
[ec2-user@ip-172-31-20-52 html]$ sudo mkdir face
[ec2-user@ip-172-31-20-52 html]$ cd
[ec2-user@ip-172-31-20-52 ~]$ cd /var/www/html/face
[ec2-user@ip-172-31-20-52 face]$ pwd
/var/www/html/face
[ec2-user@ip-172-31-20-52 face]$ sudo yum install php
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 2.4 kB 00:00
Package php-5.4.16-46.amzn2.0.2.x86_64 already installed and latest version
Nothing to do
[ec2-user@ip-172-31-20-52 face]$
```

Save the generated key

Parameters

Type of key to generate:

☒ RSA ☐ DSA ☐ ECDSA

Number of bits in a generated key:

23. Index.php file code:

```
index(1) - Notepad
File Edit Format View Help
<?php

/*

Install php - sudo yum install php
curl -sS https://getcomposer.org/installer | 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/swapon /var/swap.1

sudo wget https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
sudo mv b97ea33b5842c7894b804923c6c05580.jpg sample.jpg

*/
error_reporting(0);

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

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

|
$bucket = 'ramaaaws';
$keyname = 'virat-kohli.png';

$s3 = S3Client::factory([
    'profile' => 'default',
    'region' => 'us-east-2',
    'version' => '2006-03-01',
    'signature' => 'v4'
]);

try {
    // Upload data.
    $result = $s3->putObject([
        'Bucket' => $bucket,
```

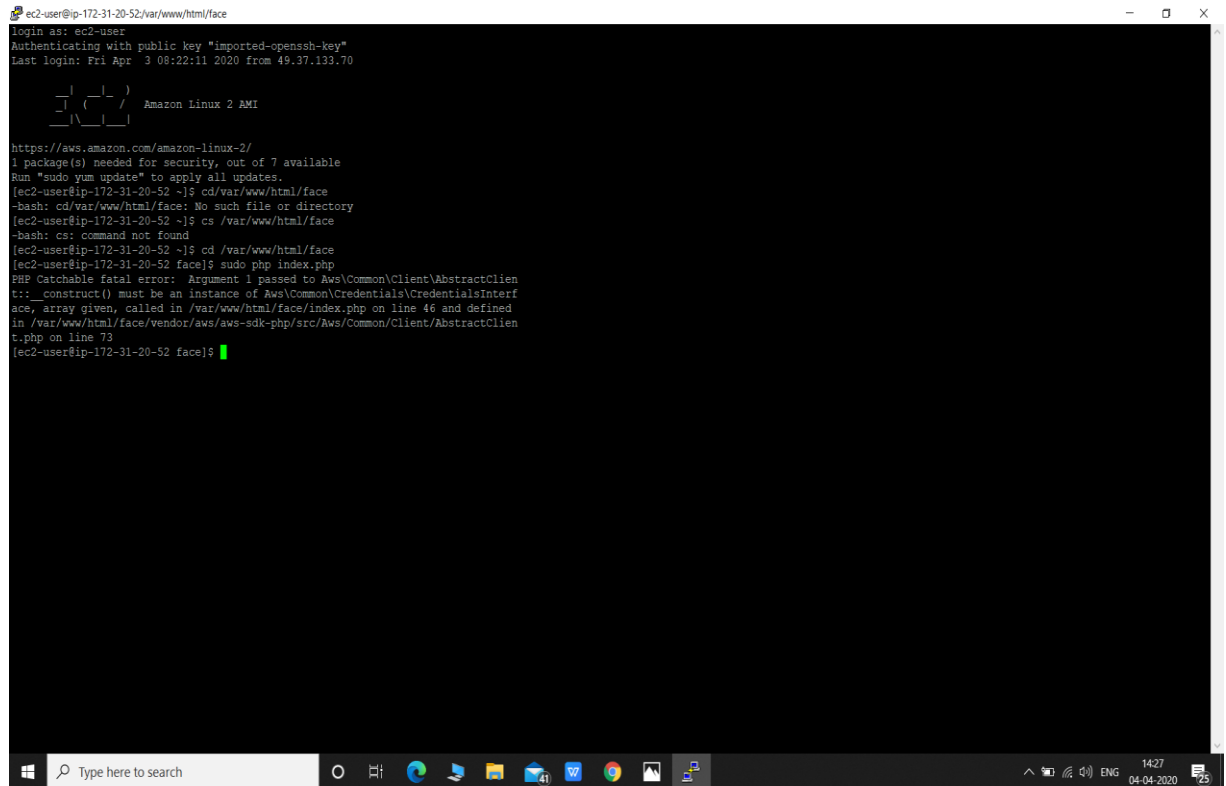
24. Upload success:

```
ec2-user@ip-172-31-20-52:/var/www/html/face
vendor virat-kohli.png
[ec2-user@ip-172-31-20-52 face]$ sudo vim index.php
[ec2-user@ip-172-31-20-52 face]$ sudo php index.php
[ec2-user@ip-172-31-20-52 face]$ sudo php index.php
[ec2-user@ip-172-31-20-52 face]$ curl -sS https://getcomposer.org/installer | ph
p
All settings correct for using Composer
The installation directory "/var/www/html/face" is not writable
[ec2-user@ip-172-31-20-52 face]$ sudo php -d memory_limit=-1 ~/composer.phar req
uire 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
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 c
aching, request and response caching, and credentials caching)
aws/aws-sdk-php suggests installing monolog/monolog (Adds support for logging HT
TP requests and responses)
aws/aws-sdk-php suggests installing symfony/yaml (Eases the ability to write man
ifests for creating jobs in AWS Import/Export)
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/gu
zzle instead.
Writing lock file
Generating autoload files
[ec2-user@ip-172-31-20-52 face]$
[ec2-user@ip-172-31-20-52 face]$ ls
composer.json  composer.lock  index.php  vendor  virat-kohli.png
[ec2-user@ip-172-31-20-52 face]$ sudo php index.php
Error retrieving credentials from the instance profile metadata server. When you
are not running inside of Amazon EC2, you must provide your AWS access key ID a
nd secret access key in the "key" and "secret" options when creating a client or
provide an instantiated Aws\Common\Credentials\CredentialsInterface object. (cl
ient error response
[status code] 404
[reason phrase] Not Found
[url] http://169.254.169.254/latest/meta-data/iam/security-credentials/)
[ec2-user@ip-172-31-20-52 face]$ sudo php index.php
Image upload done... Here is the URL: https://ramaaaws.s3.us-east-2.amazonaws.com
[ec2-user@ip-172-31-20-52 face]$
```


- **EC2 & Rekognition:**

25. Face Detect success:



A terminal window titled "ec2-user@ip-172-31-20-52:/var/www/html/face" showing the following output:

```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Fri Apr  3 08:22:11 2020 from 49.37.133.70

 _ _ _ _ _
| | | | |
|_|_|_|_|_| 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-172-31-20-52 ~]$ cd /var/www/html/face
-bash: cd /var/www/html/face: No such file or directory
[ec2-user@ip-172-31-20-52 ~]$ cd /var/www/html/face
-bash: cd: command not found
[ec2-user@ip-172-31-20-52 ~]$ cd /var/www/html/face
[ec2-user@ip-172-31-20-52 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-172-31-20-52 face]$
```

The terminal window is running on an Amazon Linux 2 AMI. The user 'ec2-user' is logged in. The terminal shows the user navigating to the directory '/var/www/html/face' and running 'sudo php index.php'. A PHP fatal error is displayed, indicating a type mismatch in the 'construct()' method of the 'AbstractClient' class. The error message states: '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'. The terminal window is titled 'ec2-user@ip-172-31-20-52:/var/www/html/face' and has a standard Linux prompt. The window is open on a Windows desktop, as evidenced by the taskbar at the bottom showing various application icons and the system clock indicating 14:27 on 04-04-2020.

