

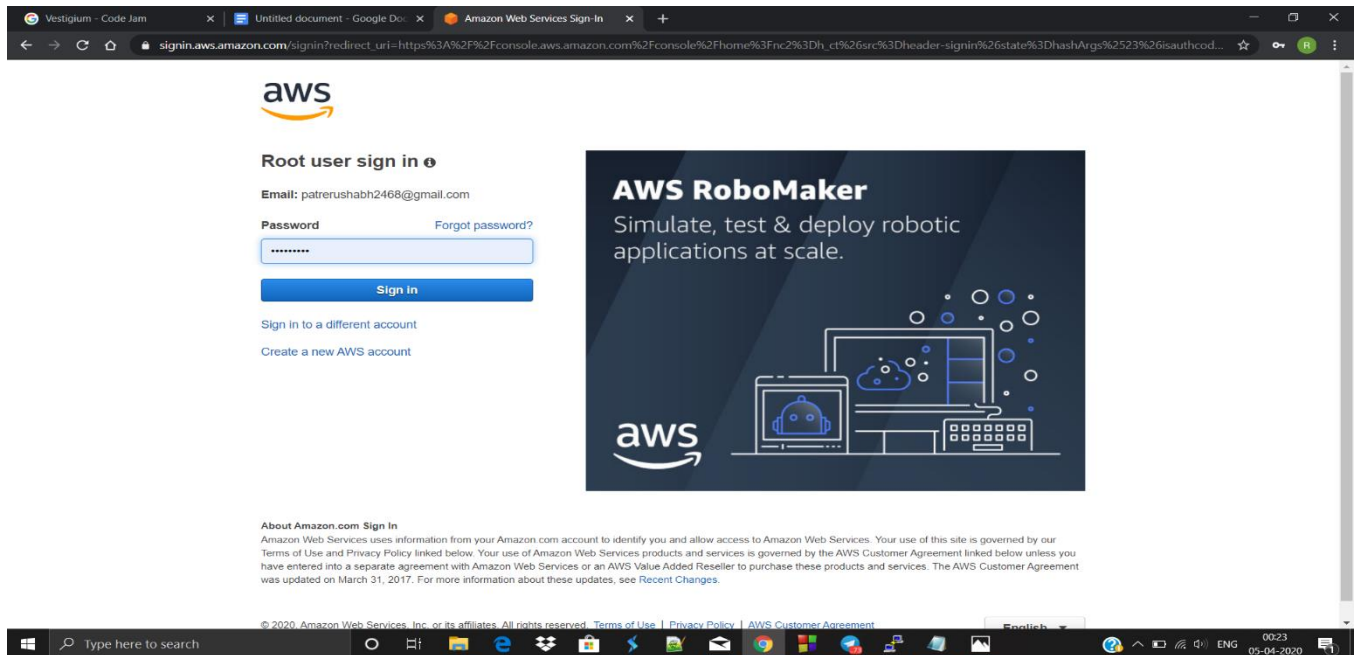
Name – Rushabh Patre

Email – patrerushabh@gmail.com

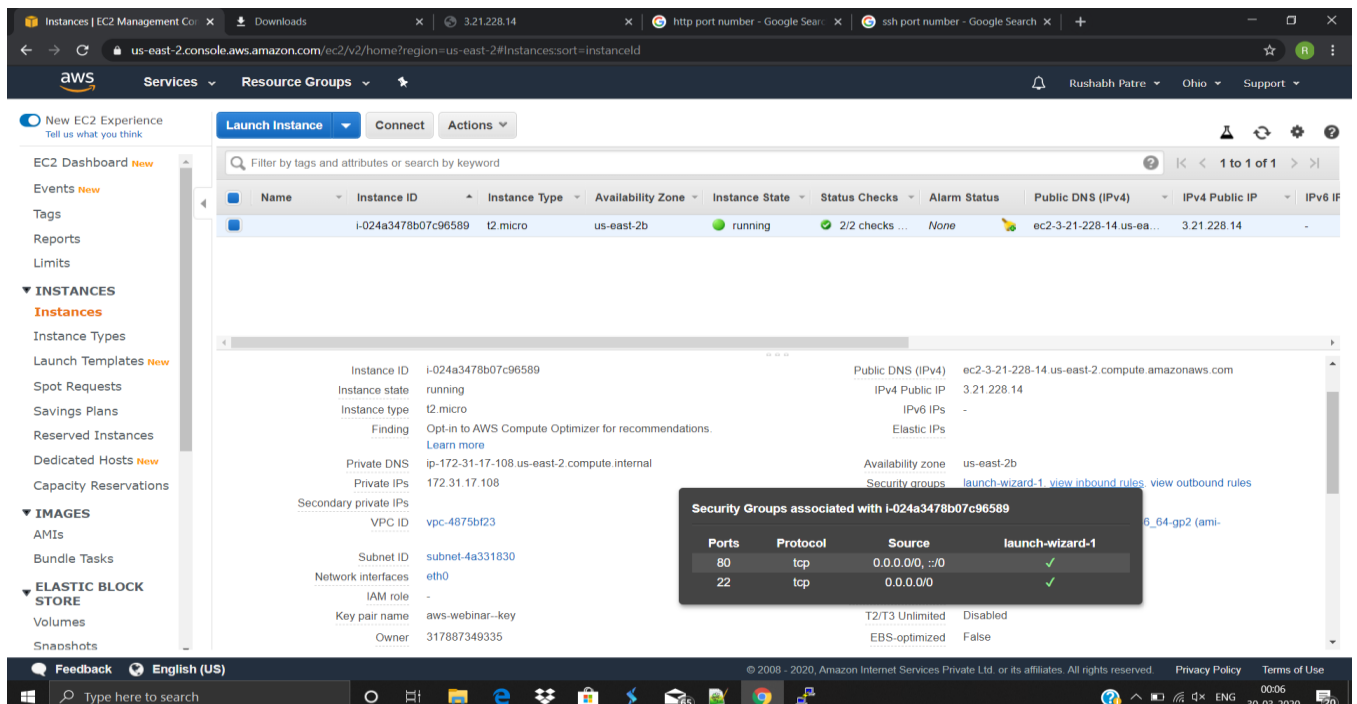
Screenshots

Screenshots of Dashboard

1.AWS login screen with username



2. EC2 Dashboard



3. S3 Dashboard

Amazon S3

Buckets (2)

Copy ARN Empty Delete Create bucket

Find bucket by name

Name	Region	Access	Bucket created
aws-webinar001	US East (Ohio) us-east-2	Objects can be public	2020-03-29T19:00:12.000Z
myawswebinarbucket	US East (Ohio) us-east-2	Not Public	2020-04-03T15:04:44.000Z

Feedback English (US)

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

Type here to search

0004 05-04-2020

4. Rekognition Dashboard

Amazon S3

Buckets (2)

Copy ARN Empty Delete Create bucket

Find bucket by name

Name	Region	Access	Bucket created
aws-webinar001	US East (Ohio) us-east-2	Objects can be public	2020-03-29T19:00:12.000Z
myawswebinarbucket	US East (Ohio) us-east-2	Not Public	2020-04-03T15:04:44.000Z

Feedback English (US)

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

Type here to search

0004 05-04-2020

Screenshots needed for EC2

1. Choosing an AMI

The screenshot shows the 'Step 1: Choose an Amazon Machine Image (AMI)' page in the AWS Management Console. The page title is 'Step 1: Choose an Amazon Machine Image (AMI)'. Below the title, there is a search bar with the placeholder text 'Search for an AMI by entering a search term e.g. "Windows"'. On the left, there is a sidebar with 'Quick Start' and 'Free tier only' filters. The main content area displays a list of AMIs. The first AMI is 'Amazon Linux 2 AMI (HVM, SSD Volume Type)' with a 'Select' button. The second AMI is 'Amazon Linux AMI 2018.03.0 (HVM, SSD Volume Type)' with a 'Select' button. The third AMI is 'Red Hat Enterprise Linux 8 (HVM, SSD Volume Type)' with a 'Select' button. The bottom of the page shows the 'Feedback' and 'English (US)' buttons.

2. Choosing an instance type

The screenshot shows the 'Step 2: Choose an Instance Type' page in the AWS Management Console. The page title is 'Step 2: Choose an Instance Type'. Below the title, there is a filter section with 'All Instance types', 'Current generation', and 'Show/Hide Columns' buttons. 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'. The 't2.micro' instance type is selected and highlighted. The bottom of the page shows the 'Cancel', 'Previous', 'Review and Launch', and 'Next: Configure Instance Details' buttons.

	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

3. Adding Storage

EC2 Management Console

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

Services Resource Groups

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

4. Configuring Security Group

EC2 Management Console

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

Services Resource Groups

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-2

Description: launch-wizard-2 created 2020-04-05T00:51:31.280+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

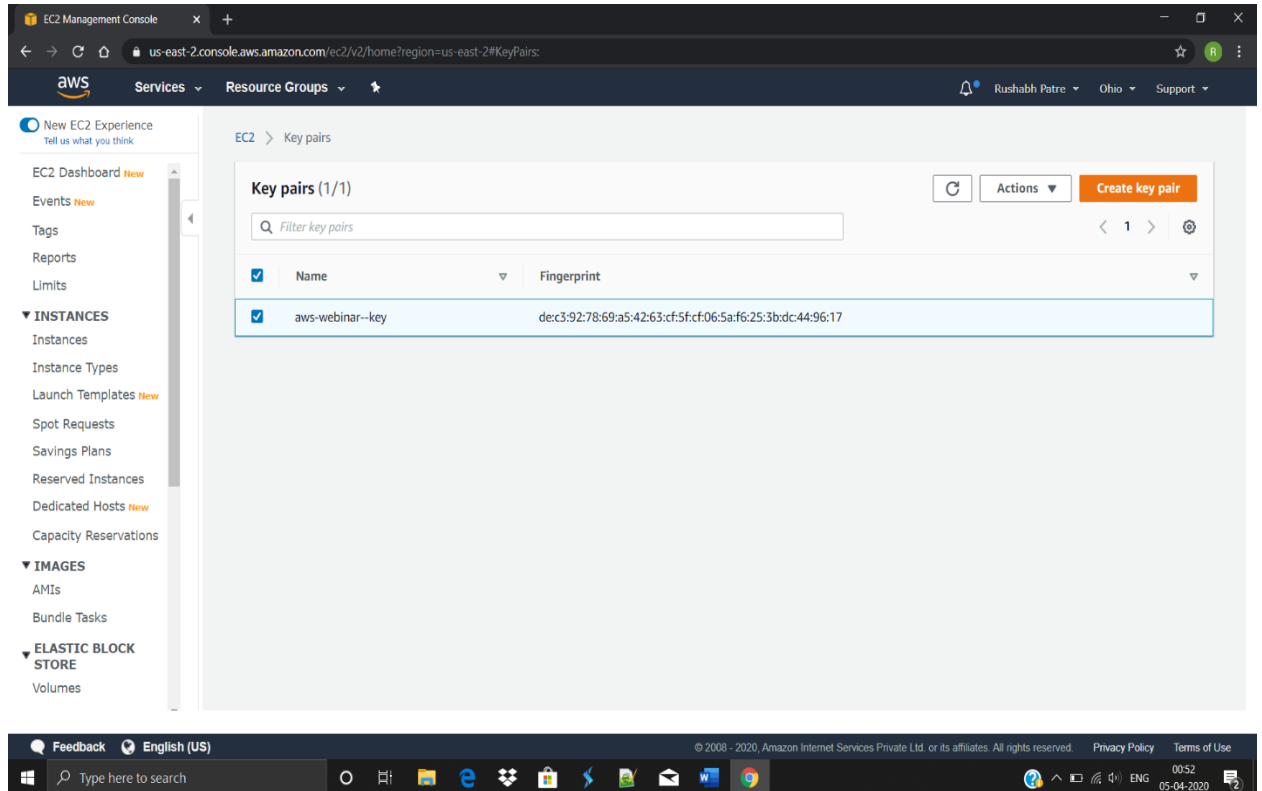
Add Rule

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.

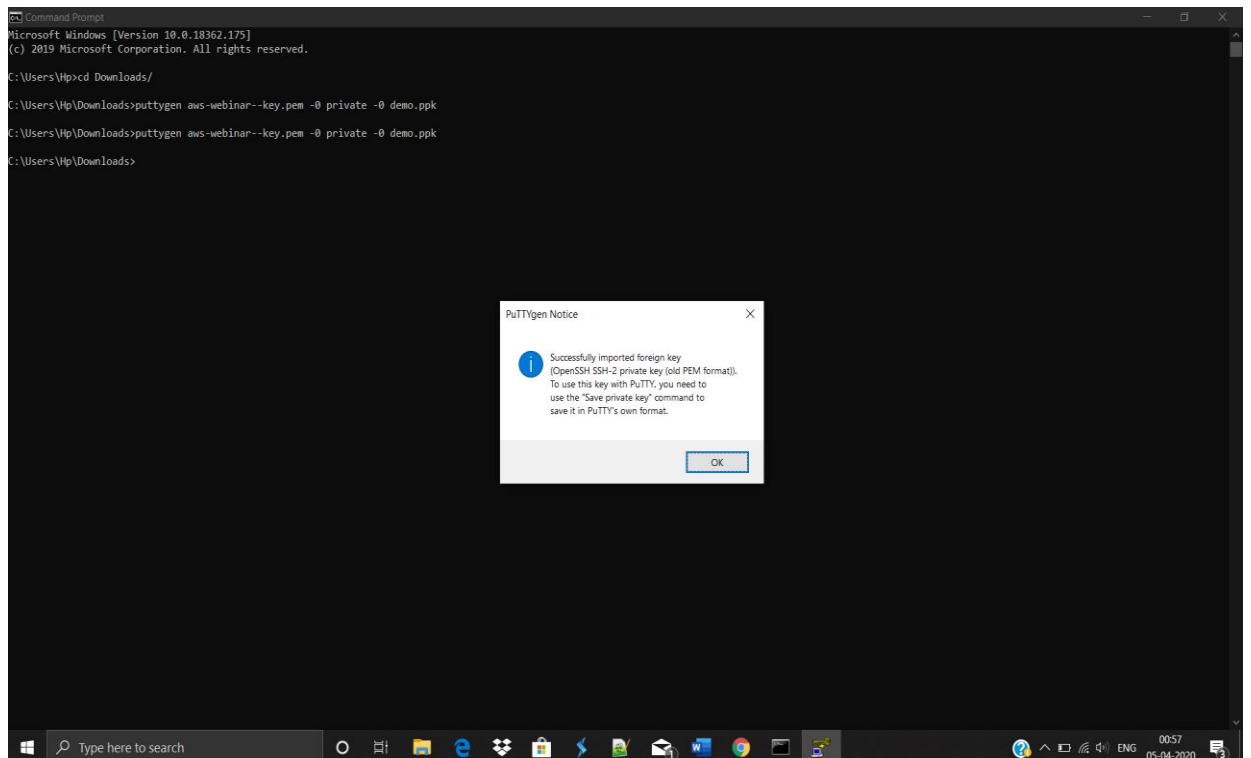
Cancel Previous **Review and Launch**

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

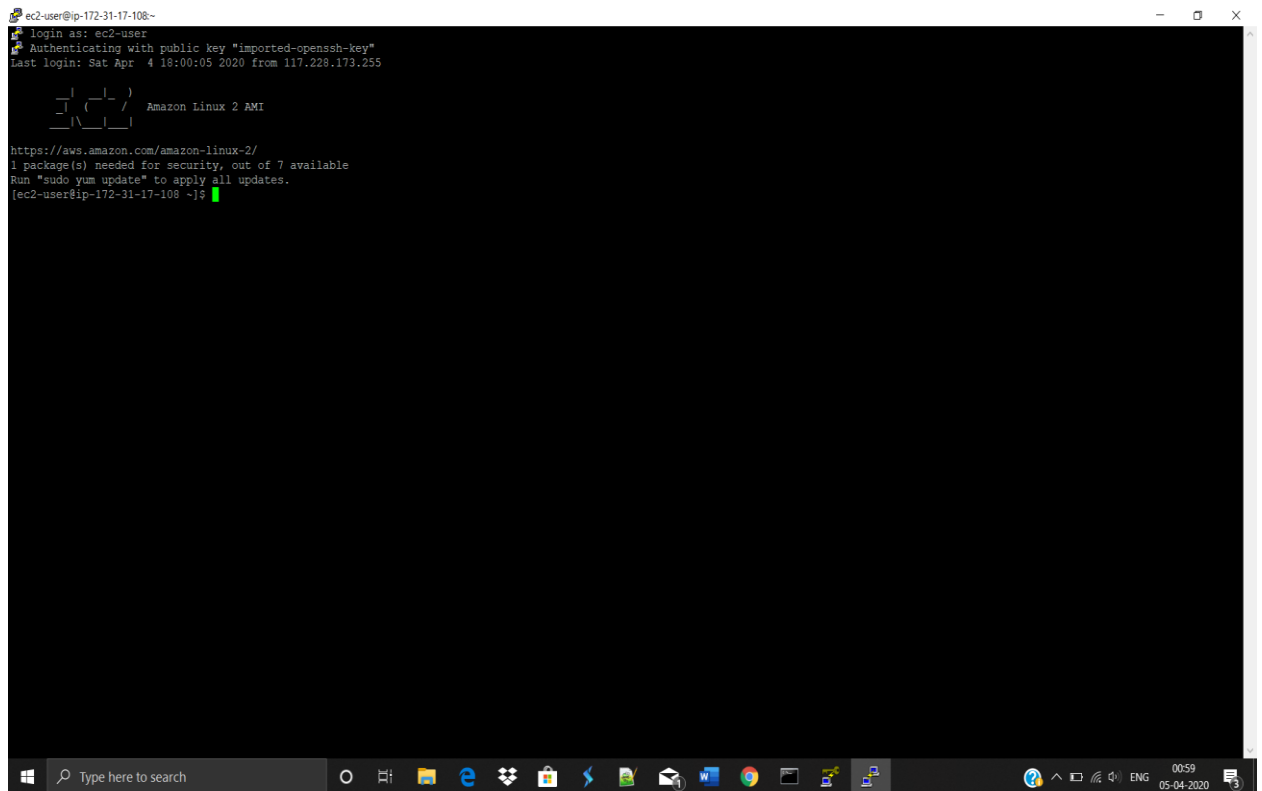
5. Key Pair



6. PUTTYgen conversion from pem to ppk

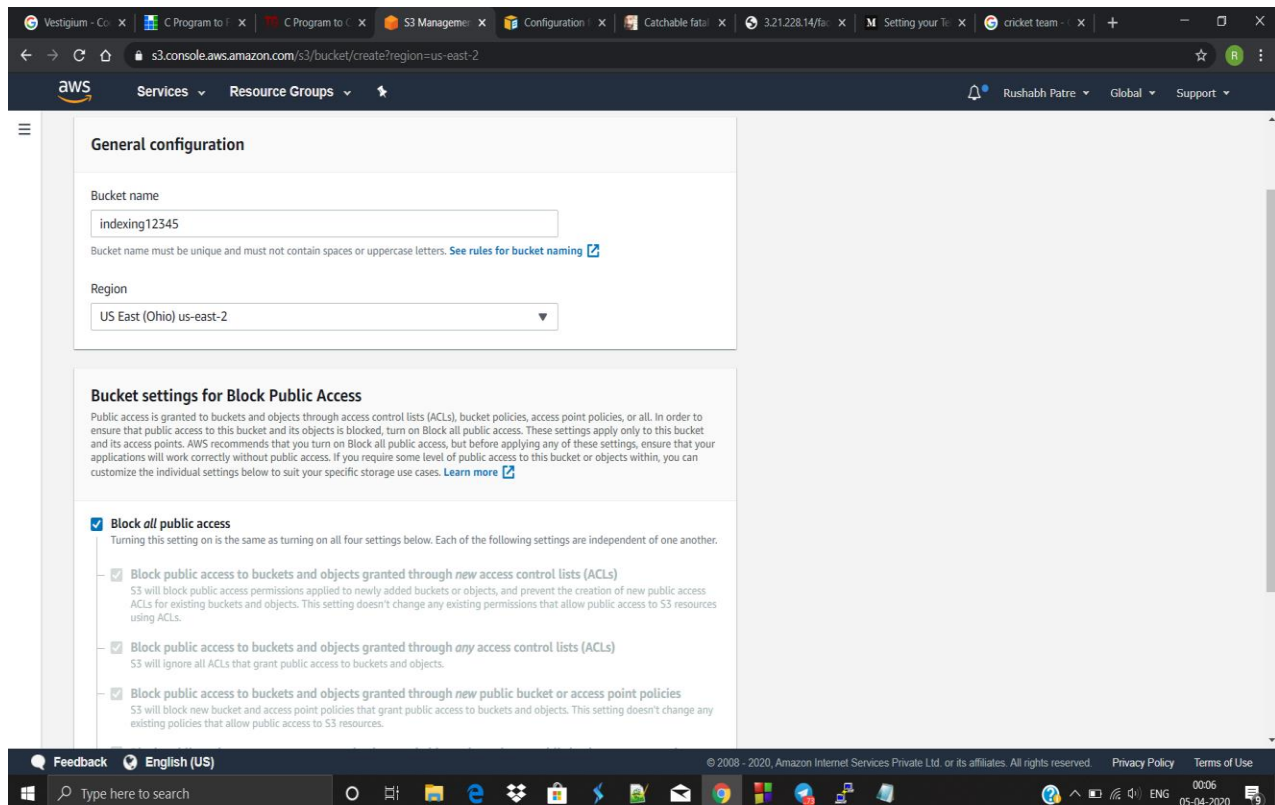


7. Logged in EC2 black screen

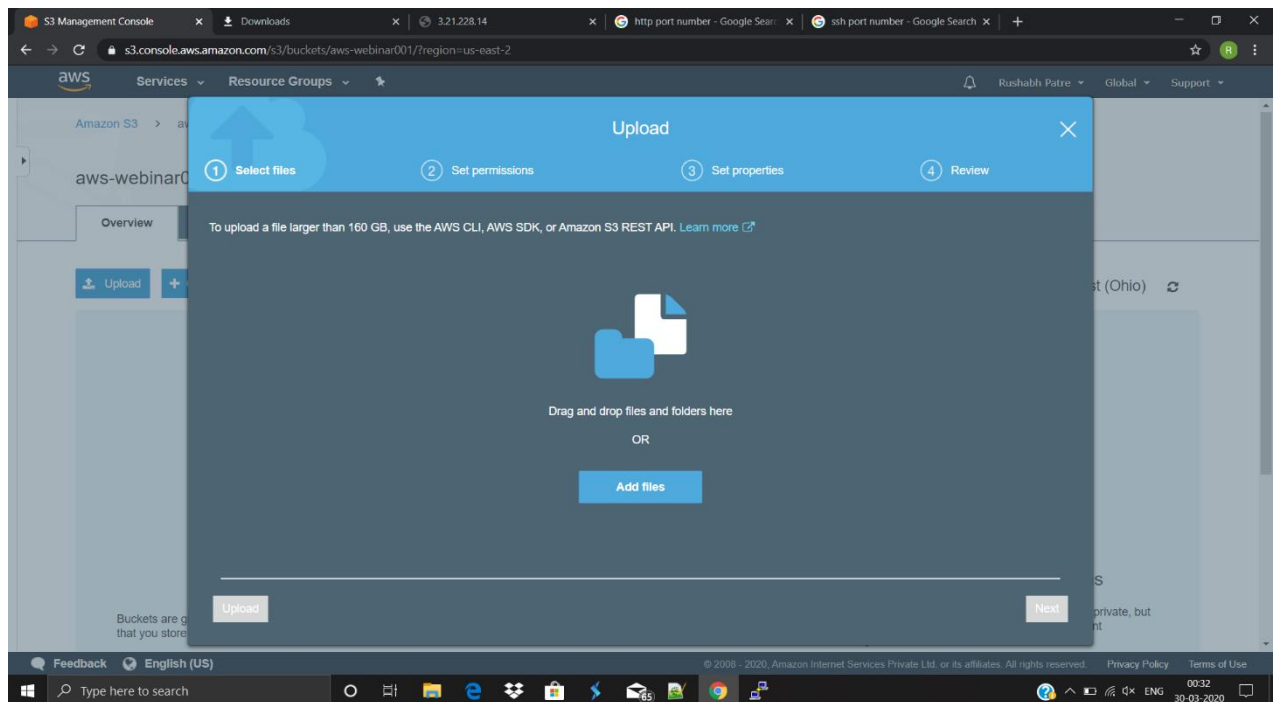


Screenshots Needed for S3

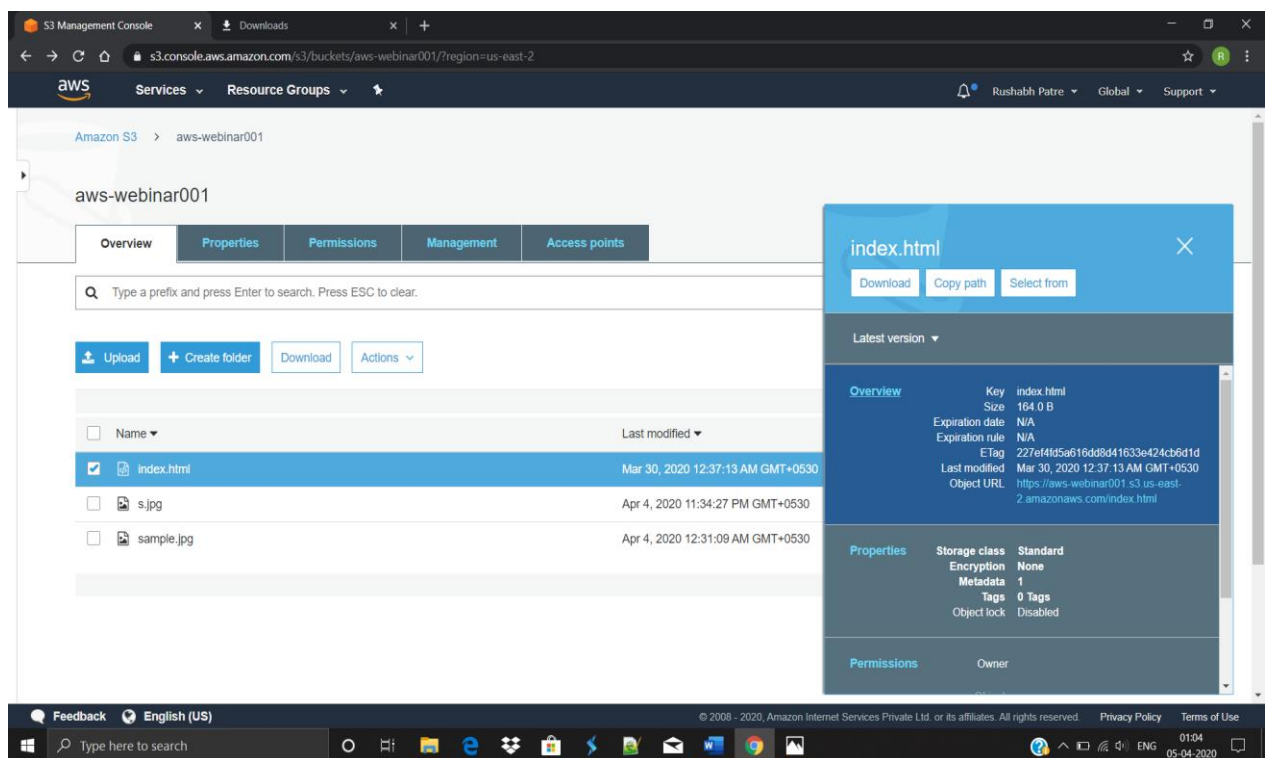
1. Create a bucket



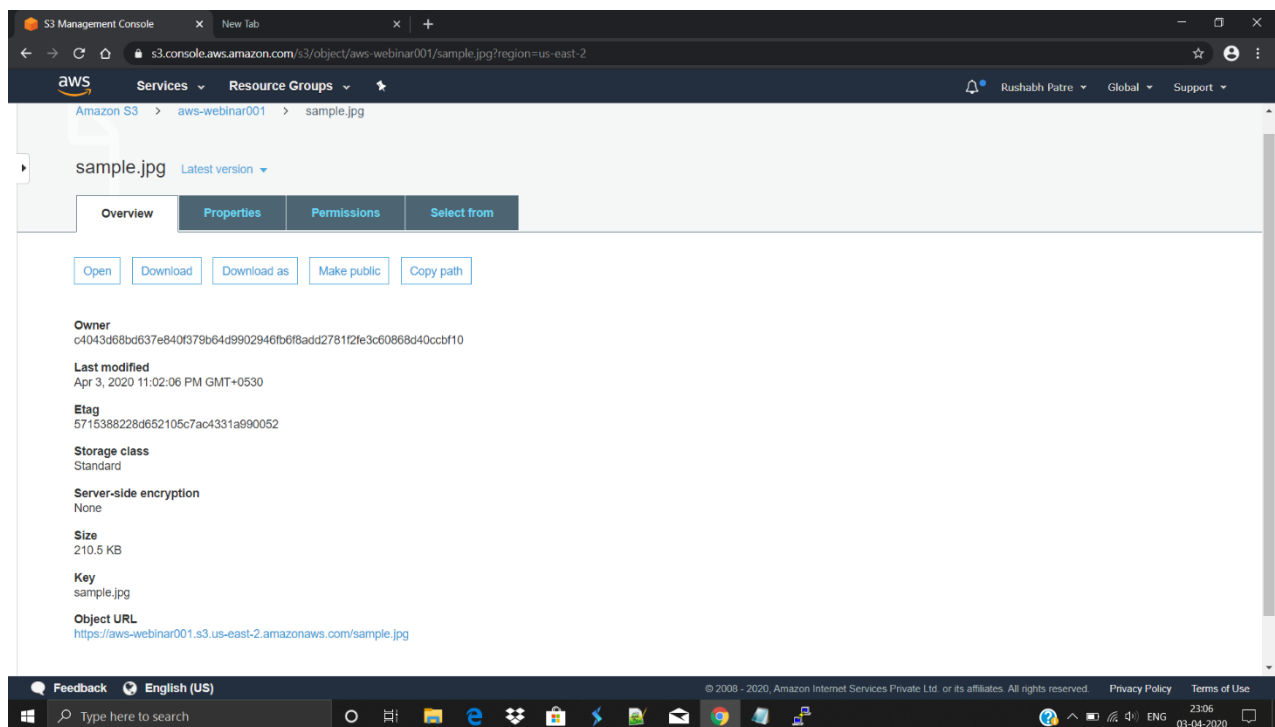
2. Uploading an Object



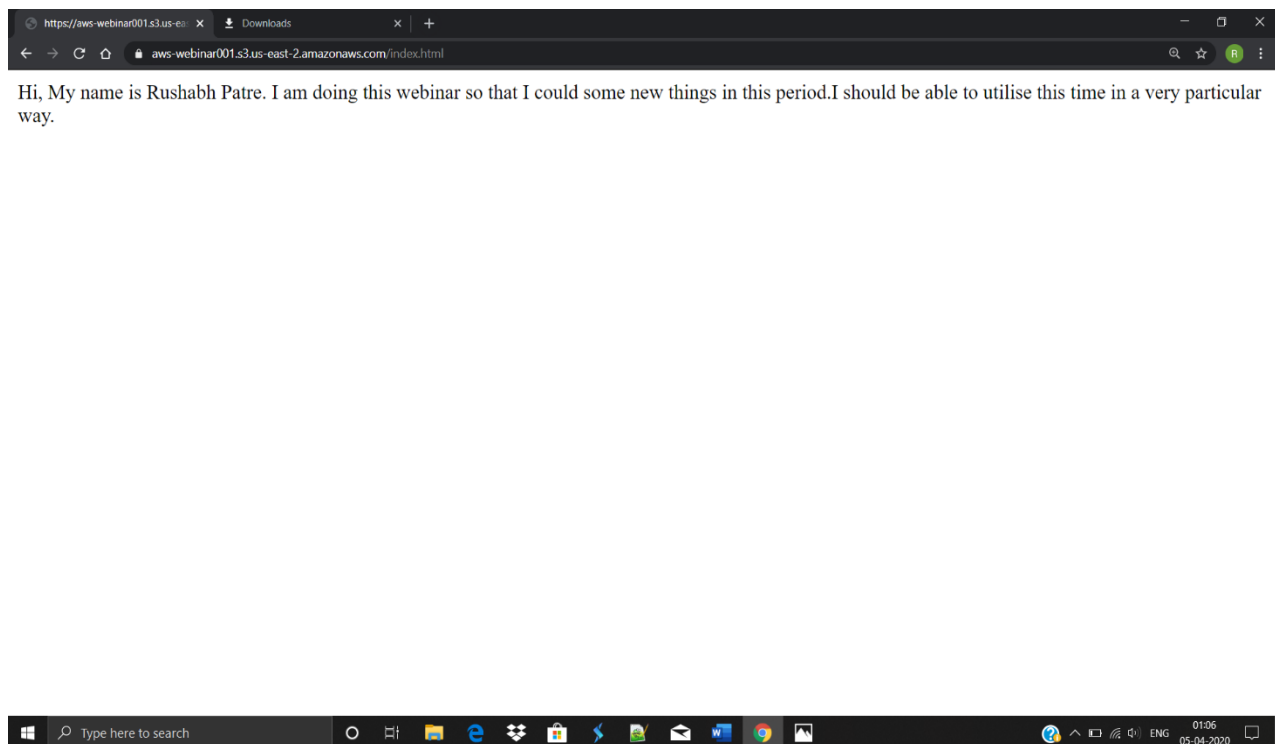
3. Enabling Static website



4. Making the object public

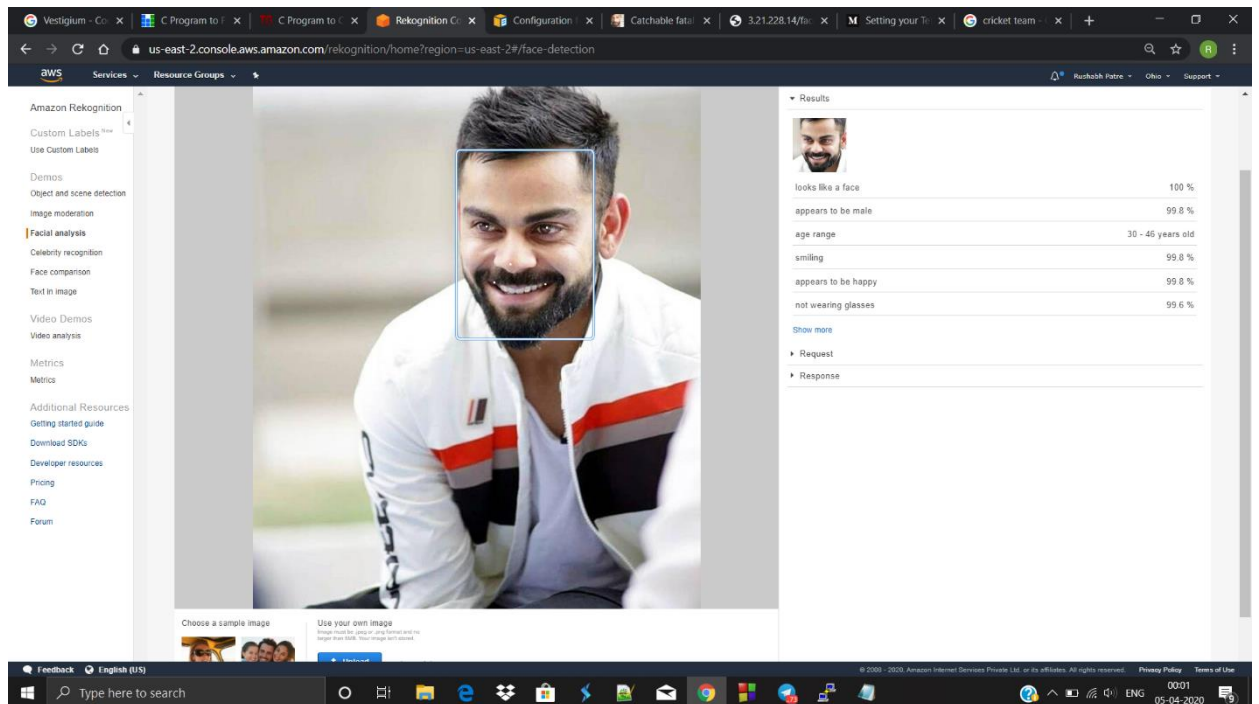


5. Checking the S3 link on the browser

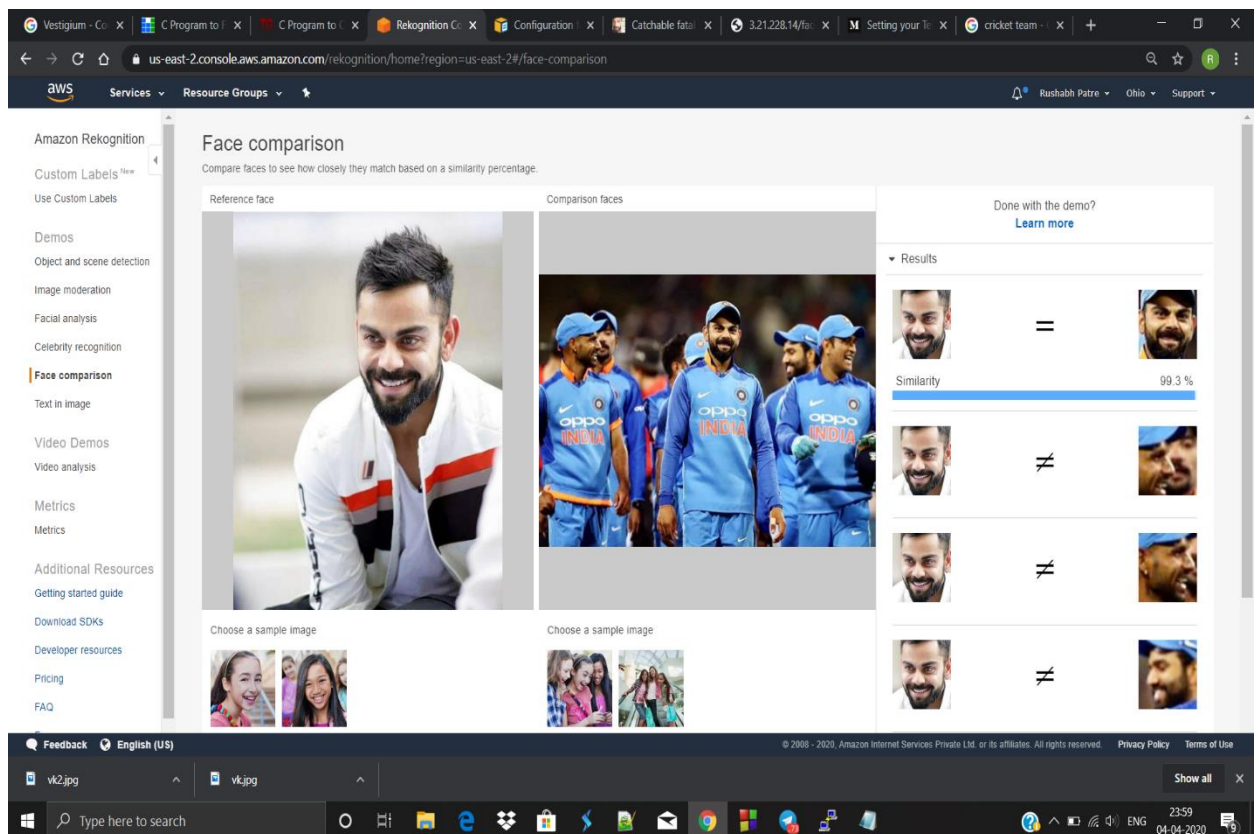


Screenshots needed for Rekognition

1. Face Detect



2. Face Compare



3. Celebrity Rekognition

The screenshot shows the AWS Rekognition console's 'Celebrity recognition' demo. The left sidebar lists various services, with 'Celebrity recognition' highlighted. The main content area features a large image of Hrithik Roshan with a bounding box around his face. Below this, there are options to 'Choose a sample image' (showing two small portraits) or 'Use your own image' (with an 'Upload' button and a text field for 'Use image URL'). To the right, the 'Results' section displays 'Hrithik Roshan' with a 'Match confidence' of 99%. The bottom of the browser window shows the Windows taskbar with various application icons and the system clock.

4. Text in Image

The screenshot shows the AWS Rekognition console's 'Text in image' demo. The left sidebar lists various services, with 'Text in image' highlighted. The main content area features a large image of a cricket team in blue uniforms with the text 'Go India!' at the bottom. Below this, there are options to 'Choose a sample image' (showing two small images) or 'Use your own image' (with an 'Upload' button and a text field for 'Use image URL'). To the right, the 'Results' section displays a list of detected text: 'NEWS | 18 |', 'book |', 'book |', 'smrt |', 'gok | mild |', 'book |', 'ool |', 'mil | smile |', and 'Go | India |'. The bottom of the browser window shows the Windows taskbar with various application icons and the system clock.

Screenshots needed for EC2 and S3

1. Installing AWS-sdk

```
ec2-user@ip-172-31-17-108:/var/www/html/face
[ec2-user@ip-172-31-17-108 face]$ sudo php -d memory_limit=-1 ~/composer.phar require aws/aws-sdk-php
Using version 2.4 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-172-31-17-108 face]$ ls
a.jpg  b97ea33b5842c7894b804923c6c05580.jpg  composer.json  composer.lock  face  Face  index3.php  index.php  sample.jpg  vendor
[ec2-user@ip-172-31-17-108 face]$ sudo rm index.php
[ec2-user@ip-172-31-17-108 face]$ sudo rm index3.php
[ec2-user@ip-172-31-17-108 face]$ sudo rm a.jpg
[ec2-user@ip-172-31-17-108 face]$ sudo rm sample.jpg
[ec2-user@ip-172-31-17-108 face]$ sudo rmdir Face
rmdir: failed to remove 'Face': Directory not empty
[ec2-user@ip-172-31-17-108 face]$ sudo rm Face
rm: cannot remove 'Face': is a directory
[ec2-user@ip-172-31-17-108 face]$ ls
b97ea33b5842c7894b804923c6c05580.jpg  composer.json  composer.lock  face  Face  vendor
[ec2-user@ip-172-31-17-108 face]$ ^C
[ec2-user@ip-172-31-17-108 face]$ sudo rm b97ea33b5842c7894b804923c6c05580.jpg
[ec2-user@ip-172-31-17-108 face]$ ls
composer.json  composer.lock  face  Face  vendor
[ec2-user@ip-172-31-17-108 face]$ sudo wget https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
--2020-04-03 17:24:06-- https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg
Resolving i.pinimg.com (i.pinimg.com)... 184.86.252.250, 2a04:4e42:2f::84
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

2020-04-03 17:24:06 (7.63 MB/s) - 'b97ea33b5842c7894b804923c6c05580.jpg' saved [215551/215551]
[ec2-user@ip-172-31-17-108 face]$ sudo mv b97ea33b5842c7894b804923c6c05580.jpg sample.jpg
[ec2-user@ip-172-31-17-108 face]$ ls
composer.json  composer.lock  face  Face  sample.jpg  vendor
[ec2-user@ip-172-31-17-108 face]$ sudo vim index.php
[ec2-user@ip-172-31-17-108 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 and secret access key in the "key" and "secret" options when creating a client or provide an instantiated Aws\Common\Credentials\CredentialsInterface object. (Client error response
[status code] 404
[reason phrase] Not Found
[url] http://169.254.169.254/latest/meta-data/iam/security-credentials/)
[ec2-user@ip-172-31-17-108 face]$ sudo rm index.php
[ec2-user@ip-172-31-17-108 face]$ sudo vim index.php
[ec2-user@ip-172-31-17-108 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-webinar001.s3.us-east-2.amazonaws.com/sample.jpg[ec2-user@ip-172-31-17-108 face]$
```

2. Installing PHP

```
ec2-user@ip-172-31-17-108-
login as: ec2-user
Authenticating with public key "imported-openssh-key"

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
1 package(s) needed for security, out of 7 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-17-108 ~]$ sudo yum install httpd
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 2.4 kB 00:00
amzn2extra-docker | 1.8 kB 00:00
Resolving Dependencies
--> Running transaction check
--> Package httpd.x86_64 0:2.4.41-1.amzn2.0.1 will be installed
--> Processing Dependency: httpd-tools = 2.4.41-1.amzn2.0.1 for package: httpd-2.4.41-1.amzn2.0.1.x86_64
--> Processing Dependency: httpd filesystem = 2.4.41-1.amzn2.0.1 for package: httpd-2.4.41-1.amzn2.0.1.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.41-1.amzn2.0.1.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.41-1.amzn2.0.1.x86_64
--> Processing Dependency: httpd filesystem for package: httpd-2.4.41-1.amzn2.0.1.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.41-1.amzn2.0.1.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.41-1.amzn2.0.1.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.41-1.amzn2.0.1.x86_64
--> Running transaction check
--> Package apr.x86_64 0:1.6.3-5.amzn2.0.2 will be installed
--> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package: apr-util-1.6.1-5.amzn2.0.2.x86_64
--> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
--> Package httpd filesystem.noarch 0:2.4.41-1.amzn2.0.1 will be installed
--> Package httpd-tools.x86_64 0:2.4.41-1.amzn2.0.1 will be installed
--> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
--> Package mod_http2.x86_64 0:1.15.3-2.amzn2 will be installed
--> Running transaction check
--> Package apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
```

```
ec2-user@ip-172-31-17-108~$
Installing : httpd-2.4.41-1.amzn2.0.1.x86_64 9/9
Verifying : apr-util-1.6.1-5.amzn2.0.2.x86_64 1/9
Verifying : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64 2/9
Verifying : httpd-2.4.41-1.amzn2.0.1.x86_64 3/9
Verifying : httpd-filessystem-2.4.41-1.amzn2.0.1.noarch 4/9
Verifying : mod_http2-1.15.3-2.amzn2.x86_64 5/9
Verifying : apr-1.6.3-5.amzn2.0.2.x86_64 6/9
Verifying : mailcap-2.1.41-2.amzn2.noarch 7/9
Verifying : generic-logos-httpd-18.0.0-4.amzn2.noarch 8/9
Verifying : httpd-tools-2.4.41-1.amzn2.0.1.x86_64 9/9

Installed:
httpd.x86_64 0:2.4.41-1.amzn2.0.1

Dependency Installed:
apr.x86_64 0:1.6.3-5.amzn2.0.2
apr-util.x86_64 0:1.6.1-5.amzn2.0.2
apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2
generic-logos-httpd.noarch 0:18.0.0-4.amzn2
httpd-filessystem.noarch 0:2.4.41-1.amzn2.0.1
httpd-tools.x86_64 0:2.4.41-1.amzn2.0.1
mailcap.noarch 0:2.1.41-2.amzn2
mod_http2.x86_64 0:1.15.3-2.amzn2

Complete!
[ec2-user@ip-172-31-17-108 ~]$ sudo service httpd start
Redirecting to /bin/systemctl start httpd.service
[ec2-user@ip-172-31-17-108 ~]$ sudo service httpd status
Redirecting to /bin/systemctl status httpd.service
• httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor prese
t: disabled)
   Active: active (running) since Sun 2020-03-29 18:21:03 UTC; 36s ago
     Docs: man:httpd.service(8)
   Main PID: 13409 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes se
rved/sec: 0 B/sec"
   CGroup: /system.slice/httpd.service
           └─13409 /usr/sbin/httpd -DFOREGROUND
             └─13410 /usr/sbin/httpd -DFOREGROUND
               └─13411 /usr/sbin/httpd -DFOREGROUND
                 └─13412 /usr/sbin/httpd -DFOREGROUND
                   └─13413 /usr/sbin/httpd -DFOREGROUND
                     └─13414 /usr/sbin/httpd -DFOREGROUND

Mar 29 18:21:03 ip-172-31-17-108.us-east-2.compute.internal systemd[1]: Start...
Mar 29 18:21:03 ip-172-31-17-108.us-east-2.compute.internal systemd[1]: Start...
Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-172-31-17-108 ~]$ sudo vim /var/www/html/index.html
[ec2-user@ip-172-31-17-108 ~]$
```

3. Index.php file code

```
C:\Users\Hp\Desktop\5_6338941033870524687.php - Sublime Text 1.4 (UNREGISTERED)
File Edit Selection Find View Tools Project Preferences Help

p.py index.html 5_6338941033870524687.php
21 error_reporting(0);
22
23 require_once(__DIR__ . '/vendor/autoload.php');
24
25 use Aws\S3\S3Client;
26 use Aws\Rekognition\RekognitionClient;
27
28 $bucket = 'aws-webinar001';
29 $keyname = 's.jpg';
30
31
32 $s3 = S3Client::factory([
33     'profile' => 'default',
34     'region' => 'us-east-2',
35     'version' => '2006-03-01',
36     'signature' => 'v4',
37     'key' => 'ARIAJWNGXAY52W2WOLTQ',
38     'secret' => '2cuFjstxfzDlaehglw0V0YusgIw0xh2c8e7COW',
39 ]);
40
41 try {
42     // Upload data
43     $result = $s3->putObject([
44         'Bucket' => $bucket,
45         'Key' => $keyname,
46         'SourceFile' => __DIR__ . "/" . $keyname,
47         'ACL' => 'public-read'
48     ]);
49
50     // Print the URL to the object.
51     $imageUrl = $result['ObjectURL'];
52     if($imageUrl) {
53         echo "Image upload done... Here is the URL: " . $imageUrl;
54     }
55     $rekognition = new RekognitionClient([
56         'region' => 'us-east-2',
57         'version' => 'latest',
58     ]);
59
60     $result = $rekognition->detectFaces([
61         'Attributes' => ['DEFAULT'],
62         'Image' => [
63             'S3Object' => [
64                 'Bucket' => $bucket,
65                 'Name' => $keyname,
```


4. Upload Success Screenshot

```
ec2-user@ip-172-31-17-108:/var/www/html/face
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Sat Apr  4 19:29:43 2020 from 117.228.173.255

 _ _ _ _ _
| | | | |   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-172-31-17-108 ~]$ cd /var/www/html
[ec2-user@ip-172-31-17-108 html]$ cd /var/www/html/face
[ec2-user@ip-172-31-17-108 face]$ pwd
/var/www/html/face
[ec2-user@ip-172-31-17-108 face]$ ls
composer.json  face  index.php  s.jpg  vendor
composer.lock  face  sample.jpg  tele.php
[ec2-user@ip-172-31-17-108 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-webinar001.s3.us-east-2.amazonaws.com/s.jpg[ec2-user@ip-172-31-17-108 face]$
```

5. Face detect success screenshot

```
PuTTY (inactive)
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Sat Apr  4 16:49:59 2020 from 117.228.173.255

 _ _ _ _ _
| | | | |   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-172-31-17-108 ~]$ cd /var/www/html
[ec2-user@ip-172-31-17-108 html]$ cd /var/www/html/face
[ec2-user@ip-172-31-17-108 face]$ pwd
/var/www/html/face
[ec2-user@ip-172-31-17-108 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-webinar001.s3.us-east-2.amazonaws.com/s.jpg[ec2-user@ip-172-31-17-108 face]$ ls
composer.json  composer.lock  face  face  index.php  sample.jpg  s.jpg  vendor
[ec2-user@ip-172-31-17-108 face]$ sudo vim tele.php
[ec2-user@ip-172-31-17-108 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-webinar001.s3.us-east-2.amazonaws.com/s.jpg[ec2-user@ip-172-31-17-108 face]$ sudo service httpd status
Redirecting to /bin/systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Sun 2020-03-29 18:21:03 UTC; 5 days ago
     Docs: man:httpd.service(8)
   Main PID: 13409 (httpd)
   Status: "Total requests: 459; Idle/Busy workers 100/0; Requests/sec: 0.000887; Bytes served/sec: 0 B/sec"
   CGroup: /system.slice/httpd.service
           └─13409 /usr/sbin/httpd -DFOREGROUND
             └─13410 /usr/sbin/httpd -DFOREGROUND
               └─13411 /usr/sbin/httpd -DFOREGROUND
                 └─13412 /usr/sbin/httpd -DFOREGROUND
                   └─13413 /usr/sbin/httpd -DFOREGROUND
                     └─13414 /usr/sbin/httpd -DFOREGROUND
                       └─13519 /usr/sbin/httpd -DFOREGROUND
                         └─13525 /usr/sbin/httpd -DFOREGROUND
                           └─17821 /usr/sbin/httpd -DFOREGROUND
                             └─17827 /usr/sbin/httpd -DFOREGROUND
                               └─17828 /usr/sbin/httpd -DFOREGROUND

Mar 29 18:21:03 ip-172-31-17-108.us-east-2.compute.internal systemd[1]: Starting The Apache HTTP Server...
Mar 29 18:21:03 ip-172-31-17-108.us-east-2.compute.internal systemd[1]: Started The Apache HTTP Server.
[ec2-user@ip-172-31-17-108 face]$ ls
composer.json  composer.lock  face  face  index.php  sample.jpg  s.jpg  tele.php  vendor
[ec2-user@ip-172-31-17-108 face]$
```