

Brian Davis
tug75084
915409443
25 March 2020

Project 1: Image Processing Lab

This Image Processing Application is composed of a web tier, application tier, and storage tier. The web tier takes an image from the user, allows the user to select an image filter, and displays the processed image to the user. The application tier handles the processing of the image. The storage tier stores the processed images in a reliable and secure fashion. The server is built using the Django framework, a powerful tool for crafting functional websites. The steps below demonstrate how to set up this project and document my implementation:

The first step to ensuring that this server works properly is creating an EC2 instance with the proper Security Groups in order to handle clients accessing the domain. This can be seen in the image below, where the EC2 instance was launched with 2 additional TCP ports, 8000 and 8080, which can be used to handle incoming requests to the server.

The screenshot shows the AWS EC2 Security Groups configuration for an instance named "launch-wizard-4". The instance ID is i-0f15cdb4efa65454c. The security group has four inbound rules defined:

Ports	Protocol	Source	Action
80	tcp	0.0.0.0/0, ::/0	✓
8080	tcp	0.0.0.0/0, ::/0	✓
8000	tcp	0.0.0.0/0, ::/0	✓
22	tcp	0.0.0.0/0	✓

Below the security group table, it shows the instance details: Root device type ebs.

After creating the EC2 instance, create a new S3 bucket as seen below. This bucket will be used for storage persistence of the processed images:

The screenshot shows the AWS S3 Buckets list. There are two buckets listed:

Name	Region	Access	Bucket created
brian-bucket-imagepro	US East (N. Virginia) us-east-1	Not Public	2020-03-22T23:38:20.000Z
imageprocessinglab	US East (N. Virginia) us-east-1	Not Public	2020-03-15T21:18:07.000Z

Next, use the Identity and Access Management entity, known as IAM role, in order to interface the new EC2 instance with the S3 bucket:

AWS Services Resource Groups ▾

vocstartsoft/user597430=tug7... Global Support

Identity and Access Management (IAM)

Summary

Delete role

Role ARN arn:aws:iam::630936438235:role/ec2-s3-fullaccess

Role description Allows EC2 instances to call AWS services on your behalf. | Edit

Instance Profile ARNs arn:aws:iam::630936438235:instance-profile/ec2-s3-fullaccess

Path /

Creation time 2020-03-22 19:47 EDT

Last activity 2020-03-24 21:45 EDT (Yesterday)

Maximum CLI/API session duration 1 hour | Edit

Permissions Trust relationships Tags Access Advisor Revoke sessions

▼ Permissions policies (1 policy applied)

Attach policies Add inline policy

Policy name ▾

▶  AmazonS3FullAccess Policy type AWS managed policy X

▶ Permissions boundary (not set)

Search IAM

Now, launch the EC2 instance and connect to it via SSH. Keep the Public DNS handy for the Web Application.

The screenshot shows the AWS Management Console with the EC2 service selected. The left sidebar contains navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, IMAGES, AMIs, Bundle Tasks, ELASTIC BLOCK STORE, Volumes, Snapshots, Lifecycle Manager, NETWORK & SECURITY, Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, and LOAD BALANCING. The main content area displays a table of instances with columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS (IPv4), IPv4 Public IP, and IPv6 IPs. One instance is highlighted: i-0f15cd84fa65454c, which is running in us-east-1b. Below this, a detailed view for the same instance is shown with tabs for Description, Status Checks, Monitoring, and Tags. The Description tab lists various attributes with their values.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs
	i-07b38fe4939db005a	t2.micro	us-east-1b	stopped	None	None	-	-	-
	i-09cef975acc3706f3	t2.micro	us-east-1b	stopped	None	None	-	-	-
	i-0d308794cd00f2ee3	t2.micro	us-east-1b	stopped	None	None	-	-	-
	i-0f15cd84fa65454c	t2.micro	us-east-1b	running	Initializing	None	ec2-18-212-30-100.compute-1.amazonaws.com	18.212.30.100	-

Instance: i-0f15cd84fa65454c Public DNS: ec2-18-212-30-100.compute-1.amazonaws.com

Description	Status Checks	Monitoring	Tags
Instance ID	i-0f15cd84fa65454c		Public DNS (IPv4) ec2-18-212-30-100.compute-1.amazonaws.com
Instance state	running		IPv4 Public IP 18.212.30.100
Instance type	t2.micro		IPv6 IPs -
Finding	You may not have permission to access AWS Compute Optimizer		Elastic IPs
Private DNS	ip-172-31-92-70.ec2.internal		Availability zone us-east-1b
Private IPs	172.31.92.70		Security groups launch-wizard-4, view inbound rules, view outbound rules
Secondary private IPs			Scheduled events No scheduled events
VPC ID	vpc-3e376844		AMI ID amzn-ami-hvm-2018.03.0.20202006.0-x86_64-gp2 (ami-0e2ff2fb72a4e45)
Subnet ID	subnet-248a840a		Platform details -
Network interfaces	eth0		Usage operation -
IAM role	ec2-s3-fullaccess		Source/dest. check True
Key pair name	tug		T2/T3 Unlimited Disabled
Owner	630936438235		EBS-optimized False
Launch time	March 25, 2020 at 11:17:38 PM UTC-4 (less than one hour)		Root device type ebs
Termination protection	False		Root device /dev/xvda
Lifecycle	normal		Block devices /dev/xvda
Monitoring	basic		Elastic Graphics ID -
Alarm status	None		Elastic Inference accelerator ID -
Kernel ID	-		Capacity Reservations -

Connect to the EC2 instance via SSH and run the server with TCP port 8080:

```
briandavis@Brians-MacBook-Pro Desktop % ssh -i "tug.pem" ec2-user@ec2-18-212-30-100.compute-1.amazonaws.com
The authenticity of host 'ec2-18-212-30-100.compute-1.amazonaws.com (18.212.30.100)' can't be established.
ECDSA key fingerprint is SHA256:23qPUaaGcqIx8rcZ4S24SqX45f46BNNe5jFV1DKes0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-18-212-30-100.compute-1.amazonaws.com,18.212.30.100' (ECDSA) to the list of known hosts.
Last login: Tue Mar 24 17:40:15 2020 from c-71-230-4-228.hsd1.pa.comcast.net
--|_--|_
_| ( _ / Amazon Linux AMI
---|_\---|_

https://aws.amazon.com/amazon-linux-ami/2018.03-release-notes/
16 package(s) needed for security, out of 20 available
Run "sudo yum update" to apply all updates.
[[ec2-user@ip-172-31-92-70 ~]$ ls
imagepro __MACOSX
[[ec2-user@ip-172-31-92-70 ~]$ cd imagepro/
[[ec2-user@ip-172-31-92-70 imagepro]$ python manage.py runserver 0:8080
Validating models...
0 errors found
March 25, 2020 - 22:21:26
Django version 1.6.1, using settings 'imagepro.settings'
Starting development server at http://0:8080/
Quit the server with CONTROL-C.
```

State of the Image Processing Application during launch:

Image Processing App
a cloud-based app for Image Processing

Upload image file

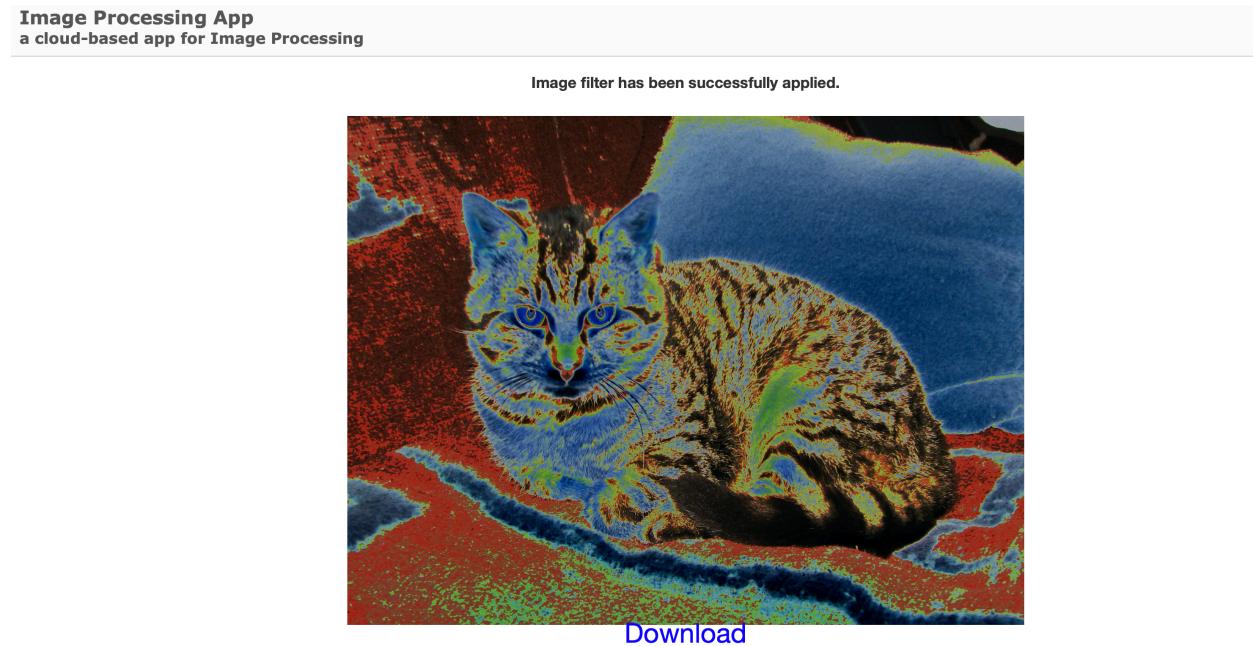
1 Step 1 Select image file 2 Step 2 Select image filter 3 Step 3 Apply filter

Select image file
Choose the image file:
Choose File no file selected

Previous Next Finish

Download image files

After an image is selected and a filter is selected, the filtered image is displayed. The user has the option to download the image immediately by clicking the blue download button:



The filtered image is now stored in S3 storage as a means of storage persistence:

A screenshot of the AWS S3 console. The top navigation bar shows "AWS Services", "Resource Groups", and other account details. The main area shows a bucket named "brian-bucket-imagepro". The "Overview" tab is selected. There is a search bar with the placeholder "Type a prefix and press Enter to search. Press ESC to clear.". Below the search bar are buttons for "Upload", "+ Create folder", "Download", and "Actions". To the right, it shows the location "US East (N. Virginia)" and a link to "Viewing 1 to 1". A table lists one item: "Arthur_the_Cat-out.jpg", which was last modified on "Mar 25, 2020 11:24:07 PM GMT-0400", has a size of "954.7 KB", and is in "Standard" storage class. At the bottom, another "Viewing 1 to 1" link is visible.

When the webpage is reloaded, the filtered images that are in S3 storage will be displayed and available to download via the blue download button. This Download Portal displays all of the filtered images from S3. The user can either download some of the previously filtered images or filter another image.

1 Step 1 Select image file 2 Step 2 Select image filter 3 Step 3 Apply filter

Select image file
Choose the image file:
 no file selected

Previous Next Finish

Download image files

Arthur_the_Cat-out.jpg
[Download](#)