

Lab Notebook - Week 6
Submitted by: Shrikrishna Bhat

6.1a EB Guestbook	1
6.1g App Engine Guestbook	4
06.2g: Cloud Run, Secret Manager (Web proxy)	6
6.3 a ECS Guestbook	10
6.3 g Cloud Run Guestbook	12
6.4 G Cloud Functions and Pub Sub	15

6.1a EB Guestbook

6.1a.1 Take a screenshot showing it has been brought up successfully

The screenshot shows the AWS Elastic Beanstalk console in Mozilla Firefox. The left sidebar shows applications and environments, with 'Eb-hello' selected. The main area displays the 'Eb-hello-env' environment overview. It includes sections for Health (Ok), Environment ID (e-69yym2wdmp), Domain (Eb-hello-env.eba-inrjnwp.us-east-1.elasticbeanstalk.com), Application name (eb-hello), and Events (21). The Events table lists log entries from May 6, 2023, detailing the successful launch and deployment of the application. A yellow sticky note in the top right corner contains the text: 'odin id: shbhat email: shbhat@pdx.edu'. The bottom status bar shows the date as May 6, 2023, and the time as 7:07 PM.

The screenshot shows the AWS Elastic Beanstalk Python application running successfully. The browser window displays a green 'Congratulations' page with the message: 'Your first AWS Elastic Beanstalk Python Application is now running on your own dedicated environment in the AWS Cloud.' Below this, it says 'This environment is launched with Elastic Beanstalk Python Platform.' To the right, there is a 'What's Next?' section with links to various AWS Elastic Beanstalk resources. A yellow sticky note in the top right corner contains the text: 'odin id: shbhat email: shbhat@pdx.edu'. The bottom status bar shows the date as May 6, 2023, and the time as 7:07 PM.

6.1a.2 Take a screenshot of the replacement VM being started.

The screenshot shows the AWS EC2 Management Console in Mozilla Firefox. The URL is [https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#instancesv=3;\\$.case=true;\\$.regex=false;\\$.client=false](https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#instancesv=3;$.case=true;$.regex=false;$.client=false). The browser title bar says "Instances | EC2 Management Console — Mozilla Firefox". The top right corner shows the user's information: "odin id: shbhat" and "email: shbhat@pdx.edu". The main content area displays a table titled "Instances (3) Info". The table has columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, Public IPv4 MAC, Elastic IP, and If. There are three rows:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 MAC	Elastic IP	If
Eb-hello-env	i-0b61956a67d78b44	Running	t3.micro	Initializing	No alarms	us-east-1b	ec2-5-83-69-54.compute...	5.83.69.34	-	-
Eb-hello-env	i-0a131d8041e8f5071	Running	t3.micro	2/2 checks passed	No alarms	us-east-1c	ec2-54-159-124-117.co...	54.159.124.117	-	-
Eb-hello-env	i-002c7d71731619872	Terminated	t3.micro	-	No alarms	us-east-1b	-	-	-	-

Below the table, a message says "Successfully terminated i-002c7d71731619872". On the left sidebar, under the "Instances" section, there is a note: "Tell us what you think". The sidebar also lists other EC2-related options like Dashboard, Global View, Events, Limits, and Instance Types.

6.1a.3 Take a screenshot of the Guestbook including the URL with the entry in it.

The screenshot shows a "Guestbook" application running on Amazon Elastic Beanstalk. The URL is <http://guestbook-env.eba-ujbds2tr.us-east-1.elasticbeanstalk.com/>. The browser title bar says "My Visitors — Mozilla Firefox". The top right corner shows the user's information: "odin id: shbhat" and "email: shbhat@pdx.edu". The main content area displays the guestbook entries:

Guestbook

[Sign here](#)

Entries

shbhat <shbhat@pdx.edu>
signed on 2023-05-01 18:44:56.712083
Hello DynamoDB

shbhat <shbhat@pdx.edu>
signed on 2023-05-02 01:58:04.339982
Hello Docker DynamoDB

shbhat <shbhat@pdx.edu>
signed on 2023-05-02 22:32:46.667032
Hello Cloud9!

shbhat <shbhat@pdx.edu>
signed on 2023-05-03 16:43:25.437366
Hello EC2!

6.1a.4 Then, visit the EC2 console to see that the specified minimum number of instances has been created, Take a screenshot of them.

Instances | EC2 Management Console — Mozilla Firefox

Lab 6 - Google Docs 06.1a: EB Guestbook Learner Lab Instances | EC2 Management Console My Visitors +

https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstancesinstanceState=running

New EC2 Experience Tell us what you think

EC2 Dashboard EC2 Global View

Events Limits

Instances

Instance Types Launch Templates

Spot Requests Savings Plans Reserved Instances Dedicated Hosts Scheduled Instances Capacity Reservations

Instances (3) info

Find instance by attribute or tag (case-sensitive)

Instance state = running

Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 IP	Elastic IP
guestbook-env	i-0e1ed8d6f97569f3	Running	t3.micro	2/2 checks passed	No alarms	us-east-1a	ec2-52-3-235-156.com...	52.3.235.156	-
guestbook-env	i-07c36585433595f4d	Running	t3.micro	2/2 checks passed	No alarms	us-east-1c	ec2-107-20-129-132.co...	107.20.129.132	-
guestbook-env	i-075f9c7e7dd0ebc04	Running	t3.micro	2/2 checks passed	No alarms	us-east-1b	ec2-5-83-212-58.com...	3.83.212.58	-

Select an instance

AWS CloudShell

Actions

2023-05-07 02:27:34 INFO instance deployment completed successfully.

2023-05-07 02:27:34 INFO Batch 2: Completed application deployment command execution.

2023-05-07 02:27:34 INFO Batch 2: Registering instance(s) with the load balancer and waiting for them to be healthy.

2023-05-07 02:29:39 INFO Batch 2: Completing application deployment command execution.

2023-05-07 02:30:02 INFO Batch 3: Starting application deployment command execution.

2023-05-07 02:30:11 INFO Batch 3: Completing application deployment command execution.

2023-05-07 02:30:11 INFO Instance deployment completed successfully.

2023-05-07 02:30:15 INFO Batch 3: Registering instance(s) with the load balancer and waiting for them to be healthy.

2023-05-07 02:30:16 INFO Batch 3: Completing application deployment command execution.

2023-05-07 02:32:23 INFO New application version was deployed to running EC2 instances.

2023-05-07 02:32:23 INFO Environment update completed successfully.

(env) [cloudshell-user@10-4-73-37 ~]\$

CloudShell Feedback Language

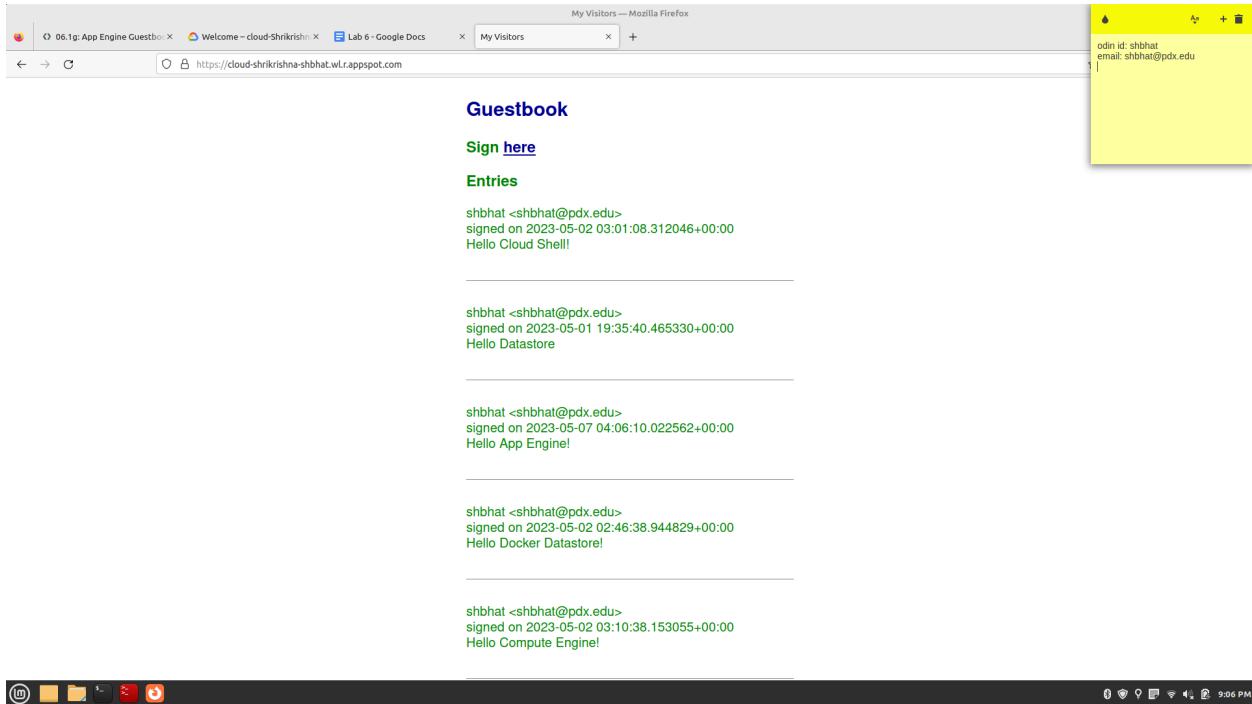
© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

N. Virginia vocabs/user250064

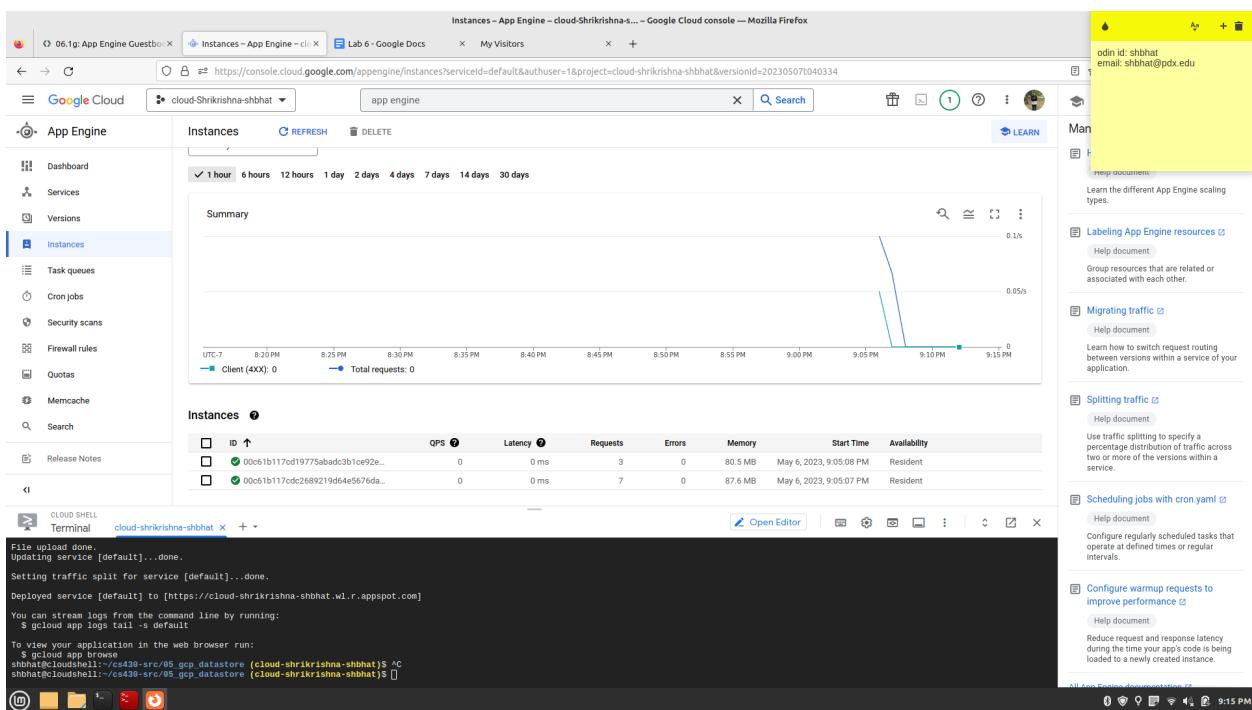
odin id: sbhhat
email: sbhhat@pdx.edu

6.1g App Engine Guestbook

6.1g.1 Take a screenshot of the output that includes the URL in the address bar for your lab notebook.

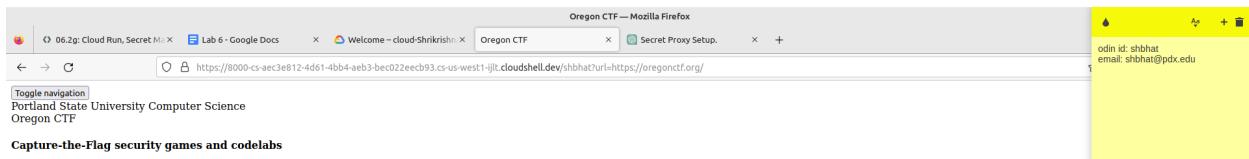
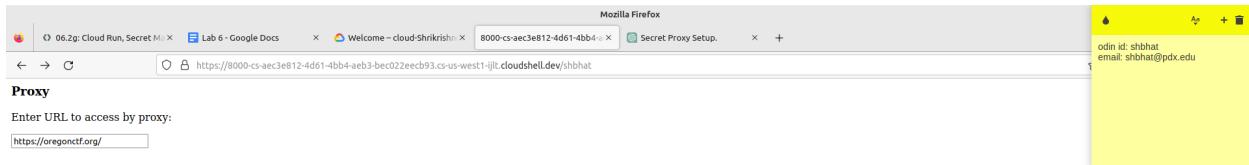


6.1g.2 Take a screenshot of them.



06.2g: Cloud Run, Secret Manager (Web proxy)

6.2g.1 Take a screenshot of the proxy and its results including the URL containing your OdinID



Capture-the-Flag security games and codelabs

Ones we've developed:

- Computer Systems Programming (CS 205) [CTF](#)
- Malware Reverse Engineering (CS 492) [CTF](#)
- angr Symbolic Execution (CS 492) [CTF](#)
- Cloud Security (CS 430/495) [Thunder CTF](#)
- Fuzzing (CS 492) [codelab](#)
- Smart contract symbolic execution (CS 410) [codelabs](#)
- Divergent Cryptography and Security (CyberPDX camp) [CTF](#)

Ones we like to teach from:

- bandit (Linux tools) [CTF](#)
- natas (Web Security) [CTF](#)
- PortSwigger (Web Security) [CTF](#)
- OWASP Damn Vulnerable (Node.js Application (Web Security)) [CTF](#)
- flows-cloud (Cloud Security) [v1](#) [v2](#)
- CloudCopter (Cloud Security) [CTF](#)
- Microcorruption (Reverse Engineering) [CTF](#)
- Security Innovation (Ethereum) [CTF](#)
- Ethernaut (Ethereum) [CTF](#)
- CryptoPals (Cryptanalysis) [CTF](#)

Portland State's CTF Slack channel [here](#)

Resources

Some recommended resources include:

- Download a Windows XP VM with IDA Pro Free installed [here](#)
- Or download IDA Pro Free [here](#)
- Download a Linux OS Box [here](#)
- PSU's CS 205 Computer Systems Programming [course](#)
- PSU's CS 430 Internet, Web, and Cloud Systems [course](#)
- PSU's CS 495 Web and Cloud Security [course](#)
- PSU's CS 492 Malware Reverse Engineering [course](#)
- PSU's CS 410 Blockchain Development and Security [course](#)



6.2g.2 What is the security advantage of passing in the secret proxy route as an environment variable?

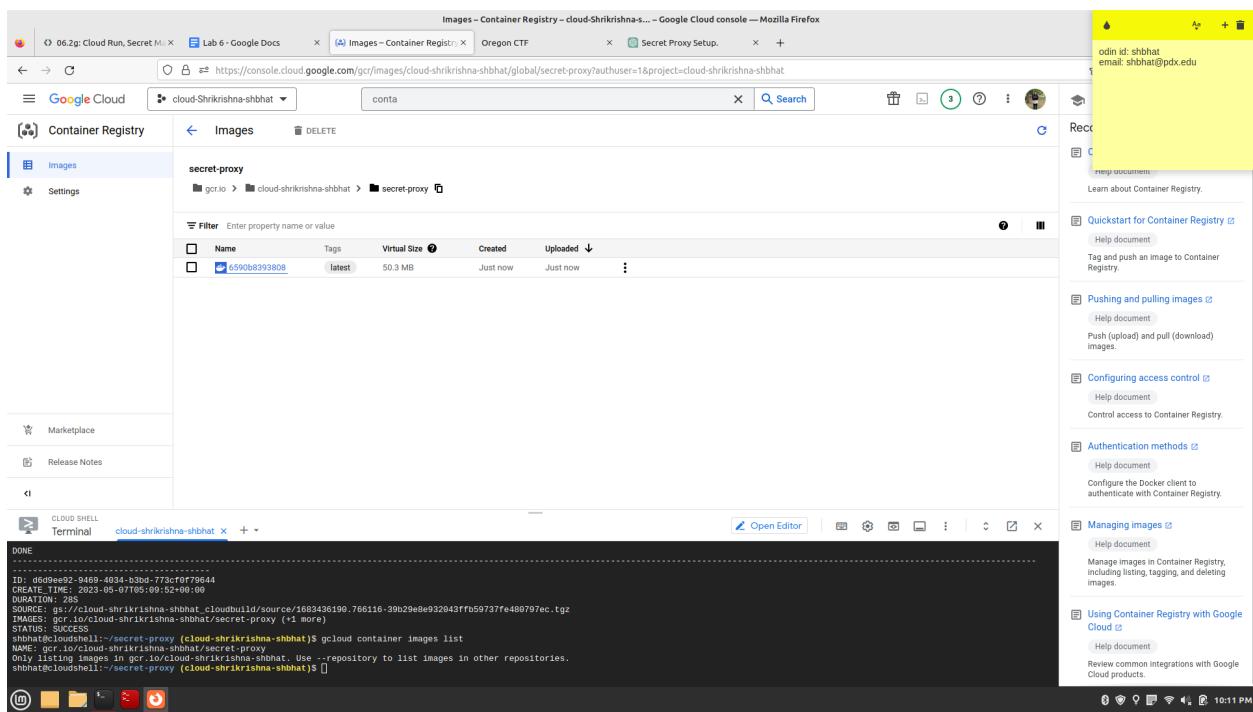
Passing the secret proxy route as an environment variable provides a security advantage by allowing the application to keep sensitive configuration details separate from the application code.

If the secret proxy route were hard-coded into the application code, it would be visible to anyone who has access to the source code, including potential attackers. This would make it easier for attackers to identify the proxy route and potentially use it for malicious purposes.

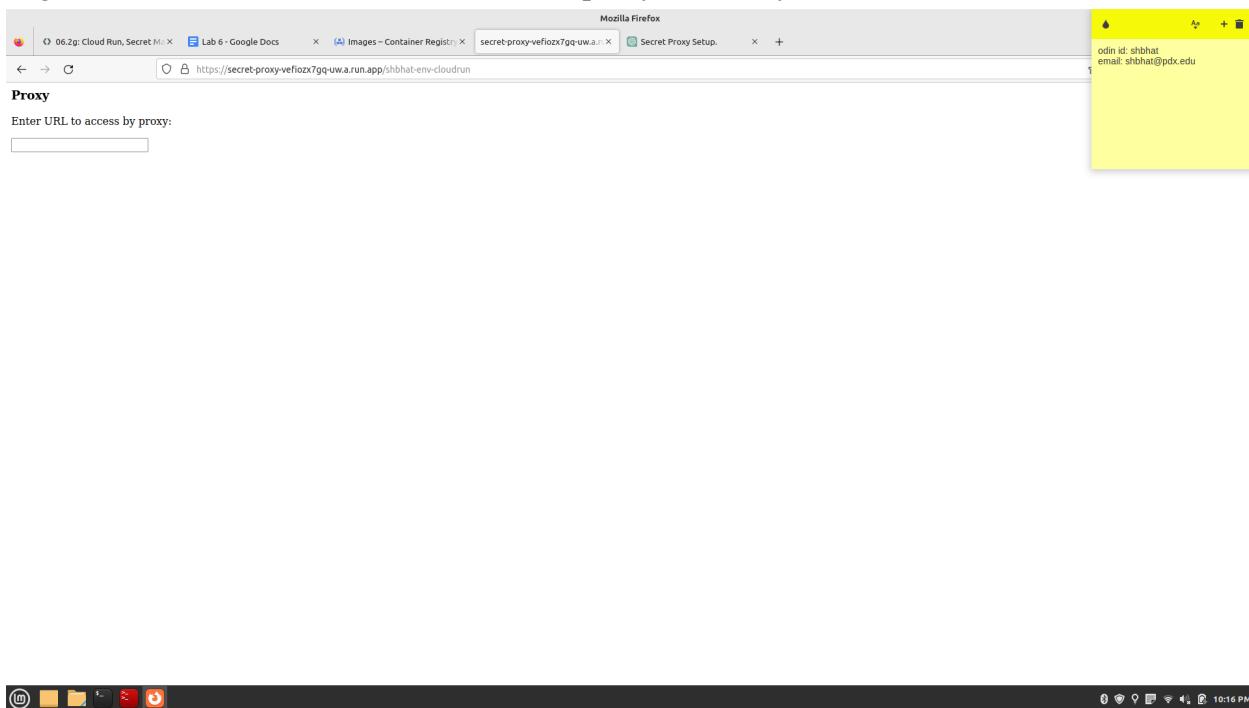
By passing the secret proxy route as an environment variable, it is kept separate from the application code and can be stored securely, for example, in a configuration file or a deployment environment. This makes it more difficult for attackers to discover the proxy route and exploit it.

Additionally, passing the secret proxy route as an environment variable allows for easy configuration changes without having to modify the application code. This can make it simpler to manage the application's configuration and reduce the risk of introducing errors or vulnerabilities during code changes.

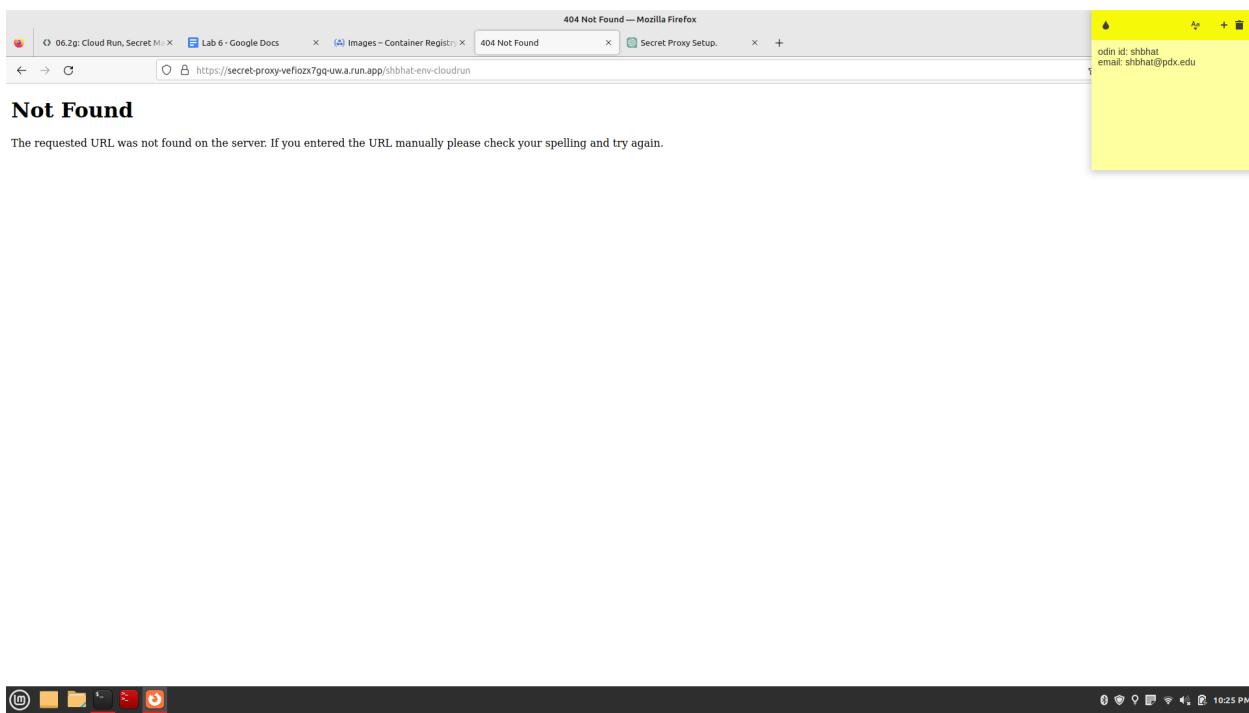
6.2g.3 Take a screenshot of the image in the registry that shows the size of the container for your lab notebook.



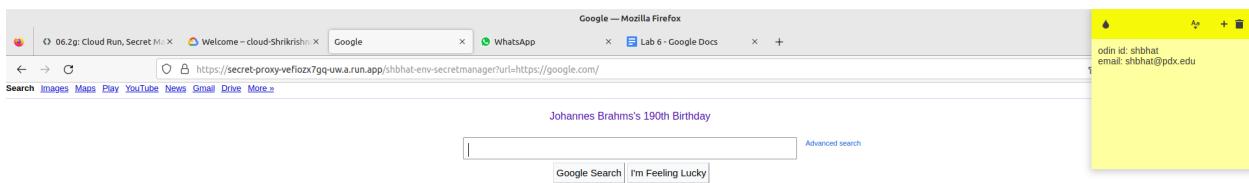
6.2g.4 Take a screenshot of it that includes the proxy URL for your lab notebook.



6.2g.5 Take a screenshot of the error page that includes the proxy URL for your lab notebook.



6.2g.6 Take a screenshot of it that includes the proxy URL for your lab notebook.



6.2g.7 Identify the vulnerability in your lab notebook that Google has prevented.

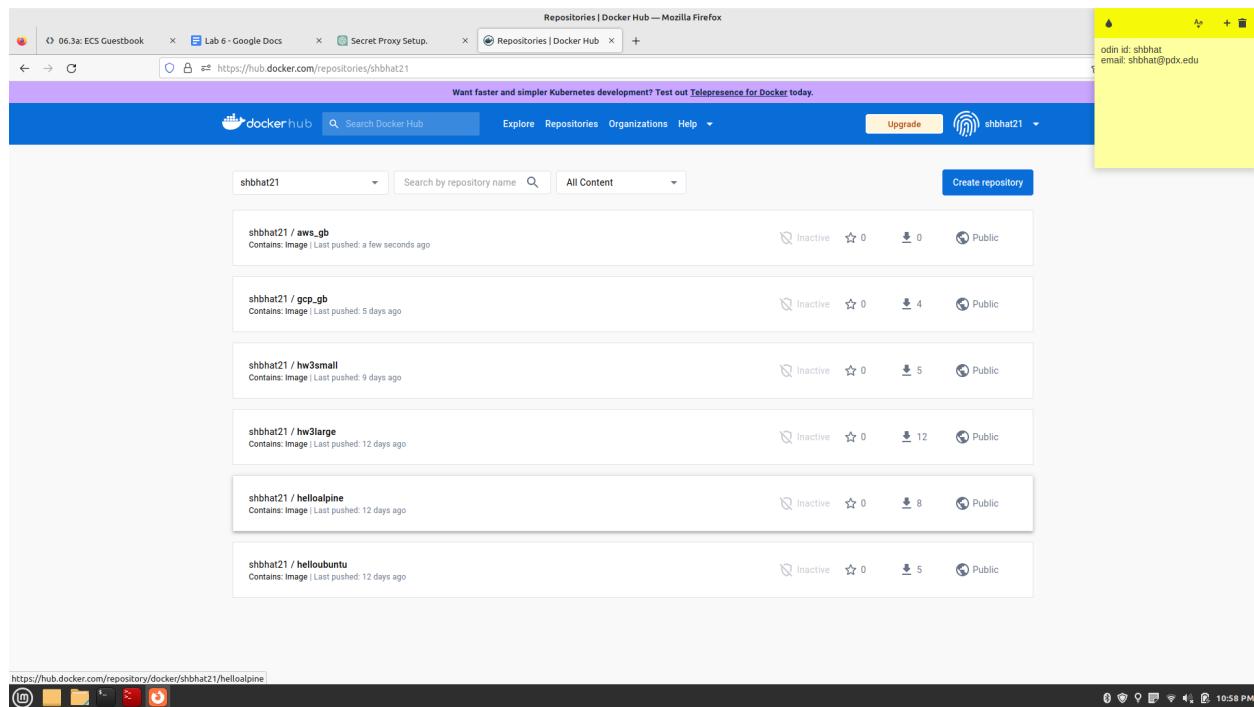
It appears that the vulnerability that Google has prevented is related to the unauthorized access of sensitive information through the Metadata service associated with the VM that runs a container. The URLs <http://169.254.169.254/computeMetadata> and <http://169.254.169.254/computeMetadata/v1> are used to access the Metadata service, which provides information about the VM's configuration and identity, including its authentication tokens.

If a container running on a VM can access the Metadata service, it may be possible for an attacker to exploit this to gain unauthorized access to the sensitive information stored on the VM or to perform other malicious actions.

Google has implemented various security measures to prevent this type of vulnerability, including limiting the scope of access to the Metadata service, requiring authentication for access, and monitoring access to the service for suspicious activity.

6.3 a ECS Guestbook

6.3a.1 Show that your image was uploaded to your account on [Docker Hub](#).



6.3a.2 Take a screenshot of the DNS name of the guestbook-lb load balancer for your lab notebook

The screenshot shows the AWS CloudWatch Metrics interface. The left sidebar lists 'Metrics' and 'Logs'. The main area displays a chart titled 'guestbook-service' with two data series: 'Latency' and 'Throughput'. The 'Latency' series has a single data point at approximately 10 seconds. The 'Throughput' series has a single data point at approximately 100 MB/s. The X-axis represents time from May 1 to May 2, 2023.

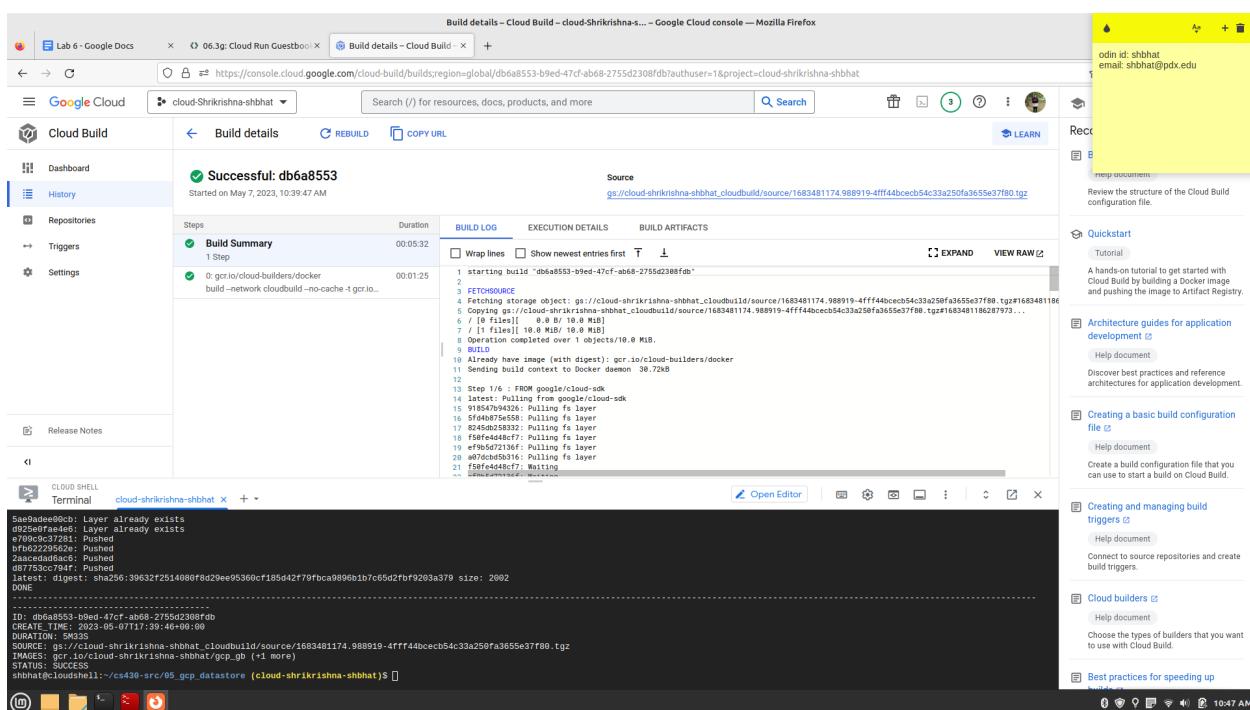
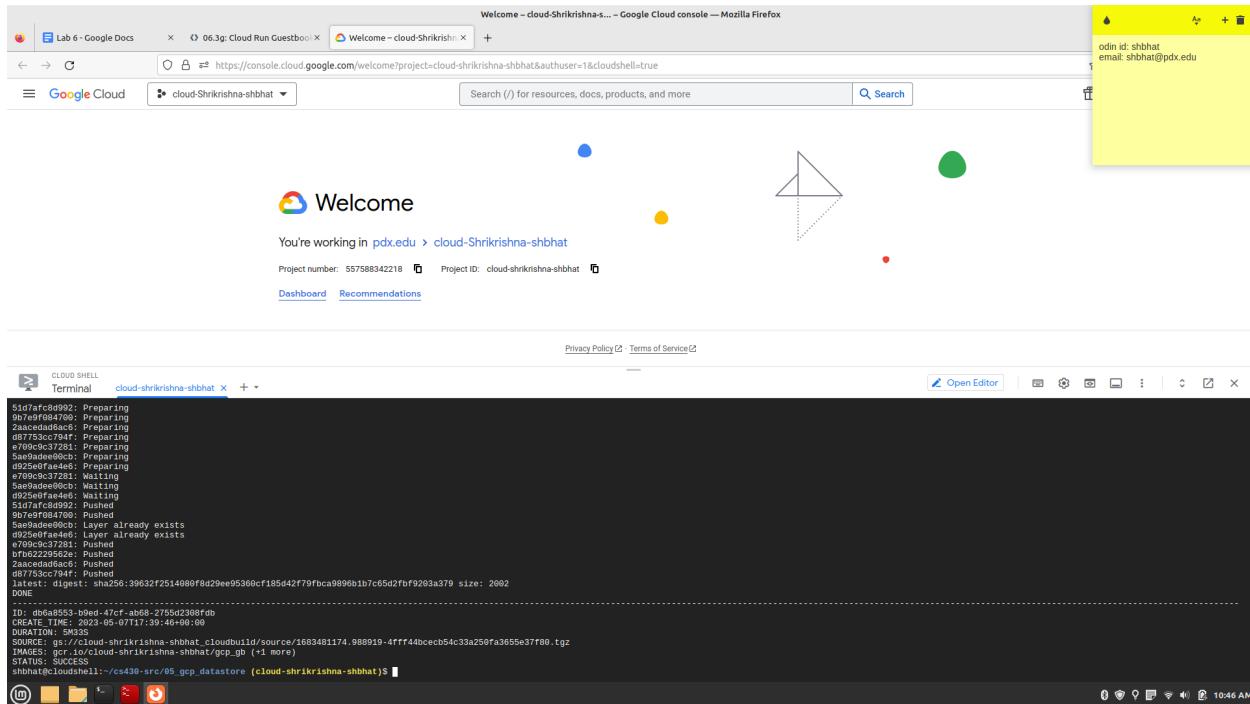
6.3a.3 Take a screenshot of the Guestbook app running in a browser that includes the DNS name of the site.

The screenshot shows the 'Guestbook' application running in a browser. The URL in the address bar is guestbook-lb-272752171.us-east-1.elb.amazonaws.com. The page displays a list of entries:

- shbhat <shbhat@pdx.edu>
signed on 2023-05-01 18:44:56.712083
Hello DynamoDB
- shbhat <shbhat@pdx.edu>
signed on 2023-05-02 01:58:04.339982
Hello Docker DynamoDB
- shbhat <shbhat@pdx.edu>
signed on 2023-05-02 22:32:46.667032
Hello Cloud9!
- shbhat <shbhat@pdx.edu>
signed on 2023-05-03 16:43:25.437366
Hello EC2!
- shbhat <shbhat@pdx.edu>
signed on 2023-05-07 06:23:27.077332
Hello ECS!

6.3 g Cloud Run Guestbook

6.3g.1 Take a screenshot that includes the output of the command and the time it took to execute.



6.3g.2 Take a screenshot showing the container image and its virtual size

The screenshot shows the Google Cloud Container Registry interface. A single image named 'gcp_gb' is listed, with a virtual size of 1.2 GB. The terminal window below shows the command used to push the image.

```

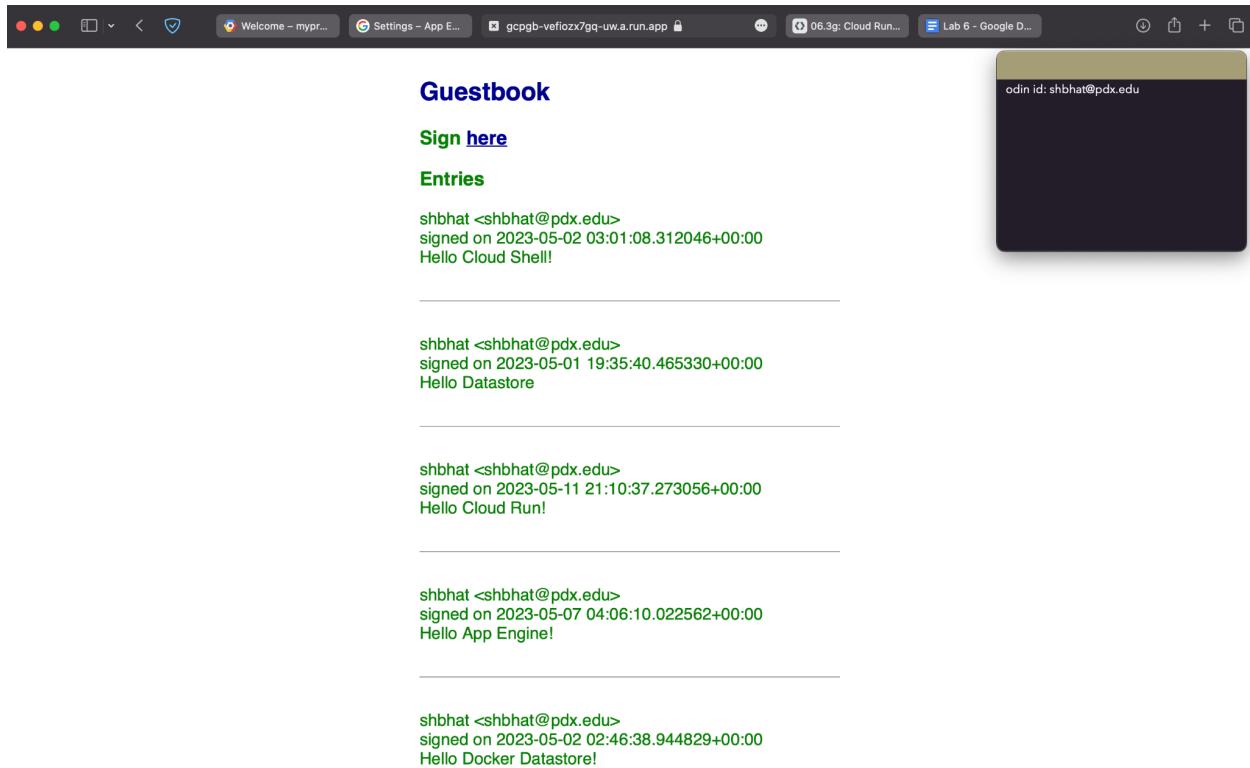
$ gcloud docker push gcr.io/cloud-shrikirshna-shbhat/gcp_gb
latest: digest: sha256:39632f2514080f8d29ee95360cf185d42f79fbca9896b1b7c65d2fb9203a379 size: 2002
DONE
...
ID: ab6a8553-b0e4-47cf-ab68-2756d2308fb0
CREATE_TIME: 2023-05-07T17:39:46+00:00
DURATION: 5M3S
SOURCE: gs://cloud-shrikirshna-shbhat/_cloudBuild
IMAGE: gcr.io/cloud-shrikirshna-shbhat/gcp_gb (+1 more)
STATUS: SUCCESS
shbhat@cloudshell:~/cs439-src/05_gcp_datstore (cloud-shrikirshna-shbhat)$
  
```

The screenshot shows the Google Cloud Storage Buckets interface. A list of buckets is displayed, including 'artifacts.cloud-shrikirshna-shbhat.ap_'. The terminal window below shows the command used to push artifacts.

```

$ gcloud docker push gcr.io/cloud-shrikirshna-shbhat/_cloudBuild
latest: digest: sha256:39632f2514080f8d29ee95360cf185d42f79fbca9896b1b7c65d2fb9203a379 size: 2002
DONE
...
ID: ab6a8553-b0e4-47cf-ab68-2756d2308fb0
CREATE_TIME: 2023-05-07T17:39:46+00:00
DURATION: 5M3S
SOURCE: gs://cloud-shrikirshna-shbhat/_cloudBuild
IMAGE: gcr.io/cloud-shrikirshna-shbhat/gcp_gb (+1 more)
STATUS: SUCCESS
shbhat@cloudshell:~/cs439-src/05_gcp_datstore (cloud-shrikirshna-shbhat)$
  
```

6.3g.3 Take a screenshot that includes the URL Cloud Run has created for your site.



6.3g.4 What port do container instances listen on?

Answer: 8080

6.3g.5 What are the maximum number of instances Cloud Run will autoscale up to for your service?

Answer: 100

6.4 G Cloud Functions and Pub Sub

6.4g.1 After downloading the file from the bucket, where is it stored?

Answer: temp_local_filename

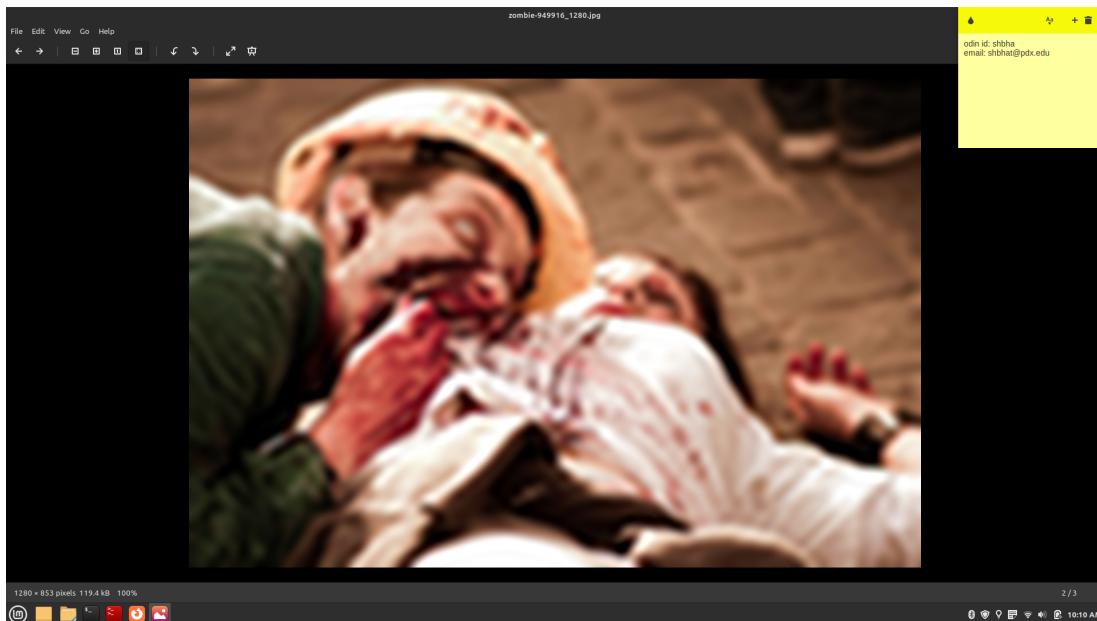
6.4g.2 What class in the ImageMagick package is used to do the blurring of the file?

Answer: def __blur_image is used for blurring the image

6.4g.3 What lines of code perform the blurring of the image and its storage back into the filesystem?

Answer: Lines 71-74

6.4g.4 Take a screenshot of the blurred image in the output bucket for your lab notebook

A screenshot of the Google Cloud Storage console showing the "Bucket details" page for "shbhat21". The "OBJECTS" tab is selected, displaying a single object named "zombie-949916_1280.jpg". The object details table shows the following information:

Name	Size	Type	Created	Storage class	Last modified	Public access	Version history	Encryption	Retention expiration
zombie-949916_1280.jpg	116.6 KB	application/octet-stream	May 10, 2023, 10:08:40 AM	Standard	May 10, 2023, 10:08:40 AM	Not public	—	Google-managed key	—

A yellow sticky note is overlaid on the top right corner of the page, containing the text "odin id: shbha" and "email: shbhat@pdx.edu". The browser interface includes a toolbar at the top, a status bar at the bottom showing "10:10 AM", and a navigation bar with multiple tabs.

6.4g.5 Include a screenshot of the output logs that show that the above image was blurred.

shbhat21 ~ Bucket details - Cloud Storage - cloud-Shrikrishna-s... - Google Cloud console - Mozilla Firefox

odin id: shbhat
email: shbhat@pdx.edu

```
shbhat21 - Bucket details - Cloud Storage - cloud-Shrikrishna-s... - Google Cloud console - Mozilla Firefox
https://console.cloud.google.com/storage/browser/shbhat21?tab=objects?authuser=1&project=cloud-Shrikrishna-shbhat&prefix=&forceOnObjectsSortingFiltering=false
Google Cloud
cloud-Shrikrishna-shbhat
buckets
Open Editor
CLOUD SHELL
Terminal
cloud-shrikrishna-shbhat + ▾
LEVEL: I
NAME: blur_offensive_images
EXECUTION_ID: tqjh14m1518
TIME UTC: 2023-05-10 17:08:40.666
LOG: Blurred image uploaded to: gs://shbhat21/zombie-949916_1280.jpg

LEVEL: I
NAME: blur_offensive_images
EXECUTION_ID: tqjh14m1518
TIME UTC: 2023-05-10 17:08:40.420
LOG: Image zombie-949916_1280.jpg was blurred.

LEVEL: I
NAME: blur_offensive_images
EXECUTION_ID: tqjh14m1518
TIME UTC: 2023-05-10 17:08:30.515
LOG: 

LEVEL: I
NAME: blur_offensive_images
EXECUTION_ID: tqjh14m1518
TIME UTC: 2023-05-10 17:08:30.515
LOG: Image zombie-949916_1280.jpg was downloaded to /tmp/tmpdxj_6xm.

LEVEL: I
NAME: blur_offensive_images
EXECUTION_ID: tqjh14m1518
TIME UTC: 2023-05-10 17:08:30.420
LOG: 

LEVEL: I
NAME: blur_offensive_images
EXECUTION_ID: tqjh14m1518
TIME UTC: 2023-05-10 17:08:30.420
LOG: The image zombie-949916_1280.jpg was detected as inappropriate.

LEVEL: I
NAME: blur_offensive_images
EXECUTION_ID: tqjh14m1518
TIME UTC: 2023-05-10 17:08:29.944
LOG: 

LEVEL: I
NAME: blur_offensive_images
EXECUTION_ID: tqjh14m1518
TIME UTC: 2023-05-10 17:08:29.944
LOG: Analyzing zombie-949916_1280.jpg.
```

6.4g.6 Why are there no items returned?

If we immediately run the “`gcloud pubsub subscriptions pull`” command after creating the subscription and publishing a message, it’s possible that the message has not yet been delivered to the subscription. There may be a delay between publishing a message and it being available for subscription, so it’s possible that the message hasn’t arrived in the subscription yet.

But in our case, we have sent the message before creating the subscription in the VM, hence when we pull, no items are returned.

6.4g.7 What is the **messageId** of the published message?

Answer: 7647889646040756

The screenshot shows the Google Cloud Compute Engine interface. The left sidebar has sections for Virtual machines, Storage, and Cloud Shell. The main content area shows a table of VM instances with one row for 'pubsub'. Below the table are several 'Related actions' cards.

Status	Name	Zone	Internal IP	External IP	Network	Connect
Green	pubsub	us-west1-b	10.138.0.28 (nic0)	34.82.39.233 (nic0)	default	SSH

Related actions

- Explore Backup and DR
- View billing report
- Monitor VMs
- Explore VM logs
- Patch management
- Load balance between VMs
- Set up firewall rules

CLOUD SHELL

```
Created topic [projects/cloud-shrikrishna-shbhat/topics/topic-shbhat].
shbhat@cloudshell: ~ [cloud-shrikrishna-shbhat]$ gcloud pubsub topics publish projects/cloud-shrikrishna-shbhat/topics/topic-shbhat --message "#1"
ERROR: (gcloud.pubsub.topics.publish) You cannot send an empty message. You must specify either a MESSAGE, one or more ATTRIBUTE, or both.
shbhat@cloudshell: ~ [cloud-shrikrishna-shbhat]$ gcloud pubsub topics publish projects/cloud-shrikrishna-shbhat/topics/topic-shbhat --message="Message #1"
messageId: "1744896464040/56"
shbhat@cloudshell: ~ [cloud-shrikrishna-shbhat]$ gcloud pubsub topics publish projects/cloud-shrikrishna-shbhat/topics/topic-shbhat --message="Message #2"
messageId: "1744896464040/56"
shbhat@cloudshell: ~ [cloud-shrikrishna-shbhat]$
```

6.4.g.8 Take a screenshot of the output of the successful pull that includes the message and its **messageId**.

Answer:

The screenshot shows a browser window with multiple tabs and an open terminal window.

Open Tabs:

- 06-4g: Cloud Functions, PubSub
- VM instances - Compute Engine
- Lab 6 - Google Docs
- Pub/Sub Subscription Retrieval
- line number vim - Google Search

Terminal Session (SSH-in-browser):

```
sshhat@pubsub:~$ gcloud pubsub subscriptions create sub-$USER
--topic=projects/cloud-shrirkrishna-shbhat/topics/topic-shbhat
Created subscription [projects/cloud-shrirkrishna-shbhat/subscriptions/sub-shbhat].
sshhat@pubsub:~$ gcloud pubsub subscriptions pull sub-$USER
Listed 0 items.
sshhat@pubsub:~$ gcloud pubsub subscriptions pull sub-$USER
[...]
Per message
+-----+
| DATA | MESSAGE_ID | ORDERING_KEY | ATTRIBUTES | DELIVERY_ATTEMPT | ACK_ID |
+-----+
Back
gc
[...]
| Message #2 | 7647889646046756 | 1F6f9f1f0xxKTTStSBVWb0tQ02zhtgJWJhFVBURqC CVEB02N0dxUfpjG4E | DVWtCFwXTVp4W10bWhwXWm1fzO-ohMj1oAKh0IzsaxuSw
| 0eqzJH9W3L1Lc-NT3Fq4AEw-BRrJy1DCcypYEU4EISE-M0SF | U6RQBhysXUZIUtCZGhR0k9eIz81IC
[...]
```

Left Sidebar (List of Lab Steps):

- Cloud Functions image blurring
- Services setup
- Code
-
- Set up service account
- Deploy the function
- Test function
- Clean up
- PubSub
- PubSub via CLI
-
- PubSub via Python
-

Bottom Buttons:

- Report a mistake
- Back
- Next

Right Sidebar (User Information):

odin id: shbha
email: shbhat@pdx.edu

Message id: 7647889646040756

6.4g.9 Take a screenshot showing the `messageIds` and messages sent

The screenshot shows a Mozilla Firefox window with multiple tabs open. The active tab is a terminal window titled "cloud-shrikrishna-shbhat". The terminal displays a stack trace from a Python script, followed by several command-line commands. The commands include navigating to a directory, running a publisher script, and publishing a message to a topic. The output shows the message ID and the published message content.

```
>
The above exception was the direct cause of the following exception:
Traceback (most recent call last):
File "/home/shbhat/.local/lib/python3.9/site-packages/google/pubsub_v1/services/publisher/client.py", line 11, in <module>
    Publisher.create(topic=name=topic_name)
File "/home/shbhat/.local/lib/python3.9/site-packages/google/pubsub_v1/services/publisher/client.py", line 602, in create_topic
    self._call(
        method='POST',
        path=f'/topics/{topic_name}',
        body=publisher_pb2.PublisherCreateTopicRequest(**kwargs))
        return wrapped_func(*args, **kwargs)
File "/home/shbhat/.local/lib/python3.9/site-packages/google/api_core/retry.py", line 349, in retry_wrapped_func
    return retry_target(
        *args,
        **kwargs)
File "/home/shbhat/.local/lib/python3.9/site-packages/google/api_core/retry.py", line 191, in retry_target
    target()
File "/home/shbhat/.local/lib/python3.9/site-packages/google/api_core/timout.py", line 129, in func_with_timeout
    return func(*args, **kwargs)
File "/home/shbhat/.local/lib/python3.9/site-packages/google/api_grpc_helpers.py", line 74, in error_remaped_callable
    raise google.api_core.exceptions.InvalidArgument(error.message) from exc
google.api.core.exceptions.InvalidArgument: 400 You have passed an invalid argument to the service (argument=GOOGLE_CLOUD_PROJECT).
(env) shbhat@cloudshell:~(cloud-shrikrishna-shbhat)$ vim publisher.py
(env) shbhat@cloudshell:~(cloud-shrikrishna-shbhat)$ ;
bash: syntax error near unexpected token `;'
(env) shbhat@cloudshell:~(cloud-shrikrishna-shbhat)$ ls
publisher.py  README-cloudshell.txt  README-cloudshell.txt  secret-proxy  thunder-ctf  training-data-analyst
(env) shbhat@cloudshell:~(cloud-shrikrishna-shbhat)$ cat p
publisher.py
python-docs-samples/
(env) shbhat@cloudshell:~(cloud-shrikrishna-shbhat)$ cat publisher.py
#!/usr/bin/python
# [REDACTED]
import datetime
from google.cloud import pubsub_v1

google.cloud.project = 'cloud-shrikrishna-shbhat'
topic_name = f'projects/{google.cloud.project}/topics/my_topic'

publisher = pubsub_v1.PublisherClient()
try:
    publisher.get_topic(topic_name)
except:
    publisher.create_topic(name=topic_name)

while True:
    msg = input('Enter a message to send: ')
    now = datetime.datetime.now().strftime('%Y-%m-%d %H:%M:%S')
    message = f'({now}) ({topic_name}): {msg}'
    print(f'Published {message} to topic {topic_name}')
    print(f'{message} published')
    publisher.publish(topic_name, data=message.encode('utf-8')).result()
    (env) shbhat@cloudshell:~(cloud-shrikrishna-shbhat)$ python3 publisher.py
Enter a message to send: hello again shbhat
Published 7647889646040756 to topic projects/cloud-shrikrishna-shbhat/topics/my_topic
Enter a message to send: hello cloudshell shbhat
Published 7647916253787681 to topic projects/cloud-shrikrishna-shbhat/topics/my_topic
Enter a message to send: [REDACTED]
```

6.4g.10 Take a screenshot showing the same `messageIds` and messages received

The screenshot shows a Mozilla Firefox window with multiple tabs open. The active tab is a terminal window titled "SSH-in-browser". The terminal displays a log of received messages from a subscriber script. The log shows three messages being received, each with a timestamp and a message content.

```
(env) shbhat@pubsub:~$ vim subscriber.py
(env) shbhat@pubsub:~$ python3 subscriber.py
Received message: 2023-05-10 17:36:52 (projects/cloud-shrikrishna-shbhat/topics/my_topic) : hello shbhat
Received message: 2023-05-10 17:37:10 (projects/cloud-shrikrishna-shbhat/topics/my_topic) : hello again shbhat
Received message: 2023-05-10 17:37:23 (projects/cloud-shrikrishna-shbhat/topics/my_topic) : hello cloudshell shbhat
```