**Task 1. Create a Cloud Storage bucket and upload an object**

1. In the console, go to **Navigation menu (Navigation menu)** > **Cloud Storage** > **Buckets**.
2. To create a new Cloud Storage bucket, click **Create**.
3. Set the bucket name as Bucket Name

**Note:** Bucket naming rules

 Do not include sensitive information in the bucket name, because the bucket namespace is global and publicly visible.

 Bucket names must contain only lowercase letters, numbers, dashes (-), underscores (\_), and dots (.). Names containing dots require [verification](https://cloud.google.com/storage/docs/domain-name-verification).

 Bucket names must start and end with a number or letter.

 Bucket names must contain 3 to 63 characters. Names containing dots can contain up to 222 characters, but each dot-separated component can be no longer than 63 characters.

 Bucket names cannot be represented as an IP address in dotted-decimal notation (for example, 192.168.5.4).

 Bucket names cannot begin with the "goog" prefix.

 Bucket names cannot contain "google" or close misspellings of "google".

 Also, for DNS compliance and future compatibility, you should not use underscores (\_) or have a period adjacent to another period or dash. For example, ".." or "-." or ".-" are not valid in DNS names.

1. Click **Continue**.
2. For **Location type**, select Region, and choose Bucket Region.
3. Click **Continue**.
4. The default storage class for your bucket is Standard. Click **Continue**.
5. Uncheck Enforce public access prevention on this bucket checkbox under **Prevent public access**.
6. Choose **Fine-grained** under **Access Control**.
7. Click **Continue**.
8. Click **Create**.

That's it — you've just created a Cloud Storage bucket!

Upload an Object to the bucket

Now upload an object into a bucket.

1. Run the following command in Cloud Shell, to download an image to Cloud Shell. A Google image from the Google homepage is used for this lab.

wget --output-document google.png https://www.google.com/images/branding/googlelogo/1x/googlelogo\_color\_272x92dp.png

Copied!

content\_copy

1. Use the gsutil cp command to upload the image from the Cloud Shell to the bucket you created:

gsutil cp google.png gs://Bucket Name

Copied!

content\_copy

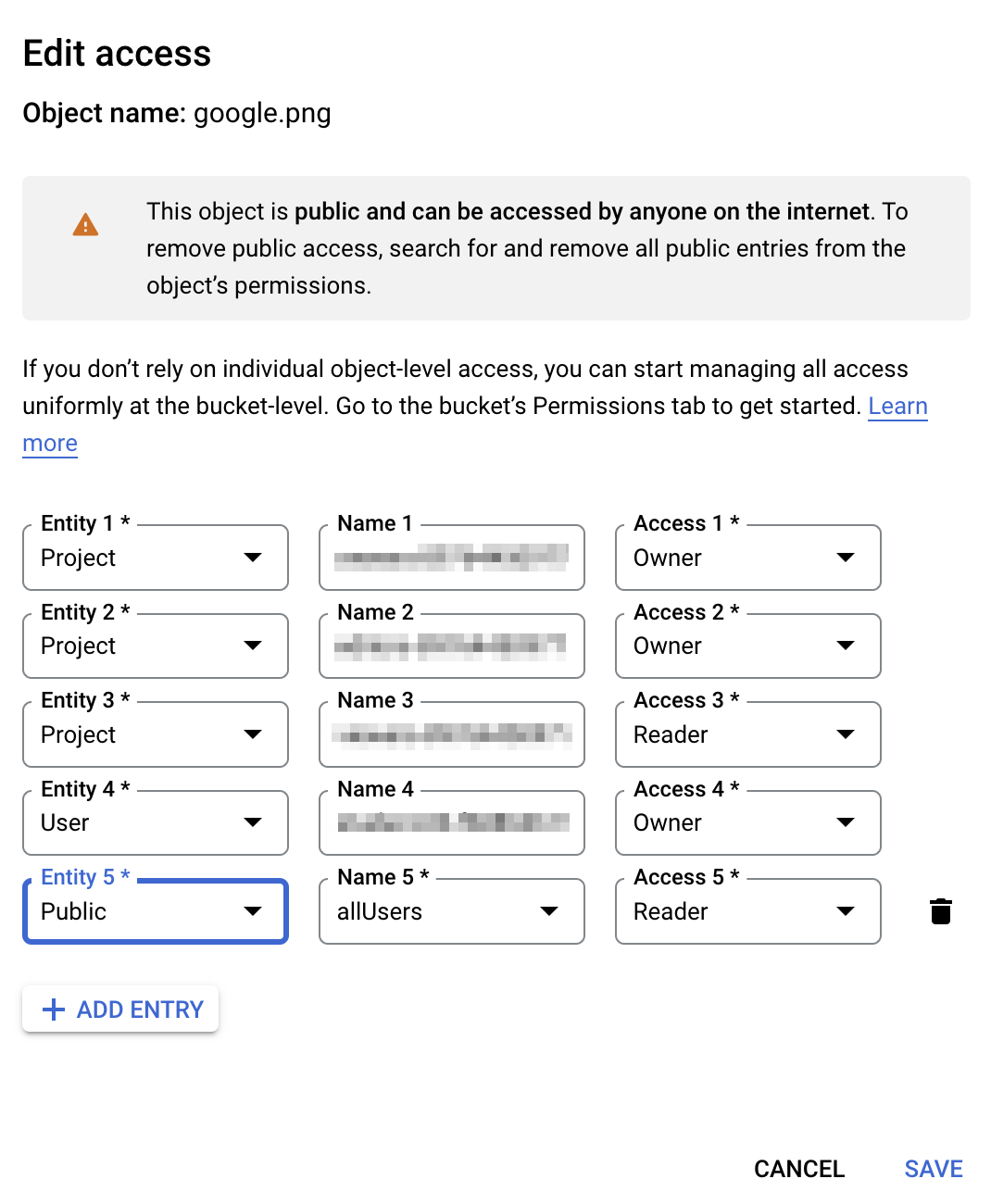
1. Remove the downloaded image from Cloud Shell:

rm google.png

Copied!

content\_copy

1. Locate the object you have uploaded to the bucket by navigating to **Cloud Storage > Buckets > Bucket Name**.
2. Now, click on the three dots on the right side of the object you uploaded and click **Edit access**.
3. Click on **Add Entry** and set the entity as **Public** from the drop-down list.
4. Click **Save**.



Click **Check my progress** to verify the objective.

Create a Cloud Storage Bucket and upload an object

Check my progress

**Task 2. Create a Load Balancer**

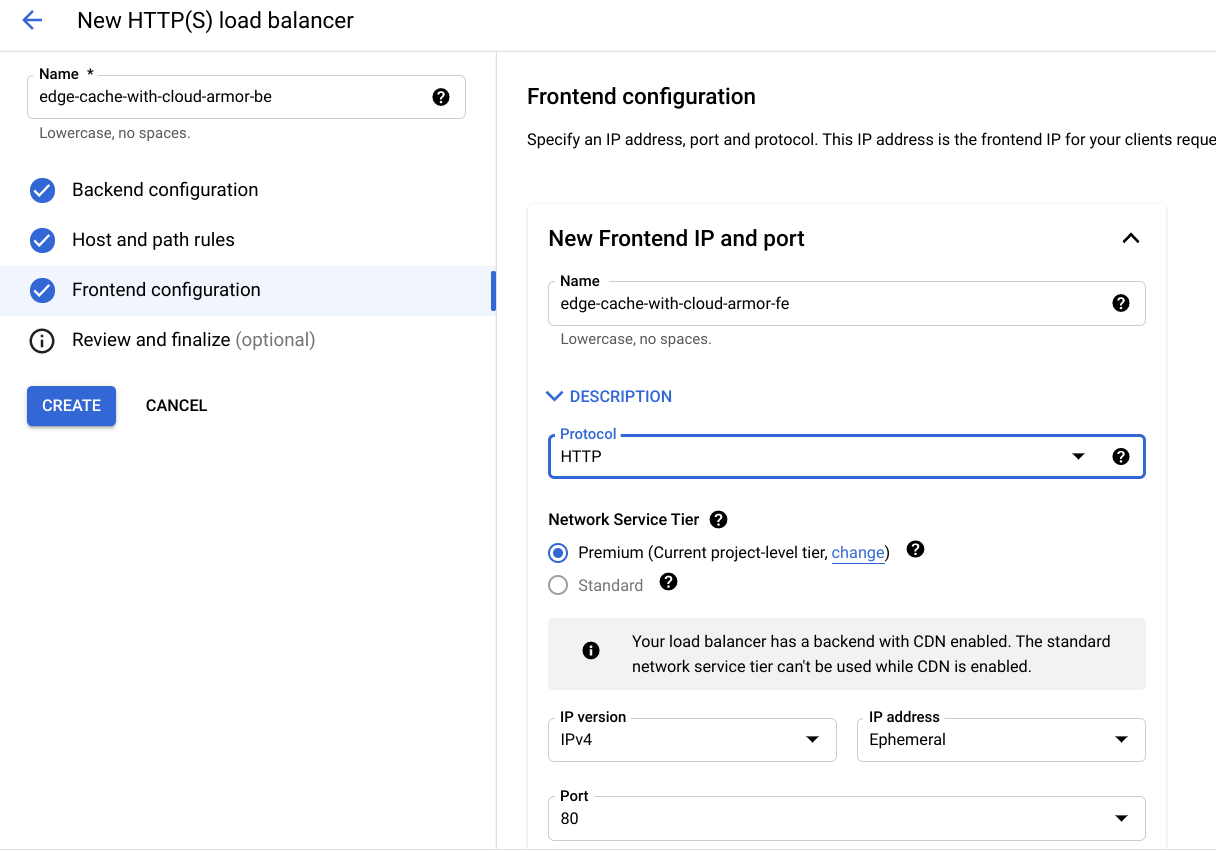
In this section, you create an HTTP Load balancer.

1. Navigate to **Networking (Navigation menu icon)** > **Network services** > **Load Balancing**.
2. Click **Create Load Balancer**.
3. Under **HTTP(S) Load Balancing**, click on **Start configuration**.
4. Select **From Internet to my VMs or serverless** and **Global HTTP(S) Load Balancer (classic)**.
5. Click **Continue**.
6. Name the load balancer as edge-cache-lb.

Create frontend configuration

To create the frontend configuration:

1. Click on **Frontend configuration**.
2. For the frontend configuration use HTTP (though HTTPS also works if you have a certificate) and an ephemeral IP address and ensure that you have selected the premium tier network. This is by default.
3. Click **Done**.



Create backend configuration

To create the backend configuration:

1. Click on **Backend configuration**.
2. For **Backend services & backend buckets**, click **Create a backend bucket**.
3. Set the **Backend bucket name** to lb-backend-bucket.
4. In the next field, select the Cloud Storage bucket created in the earlier section, by clicking the **Browse** button.
5. Leave all other values at their defaults.
6. Click **Create**.

Create host and path rules

To create host and path rules:

1. Click on **Host and path rules** on the left.
2. Select **simple host/path rule** to send any request to the bucket. This is the default option.

Review and create the HTTP Load Balancer

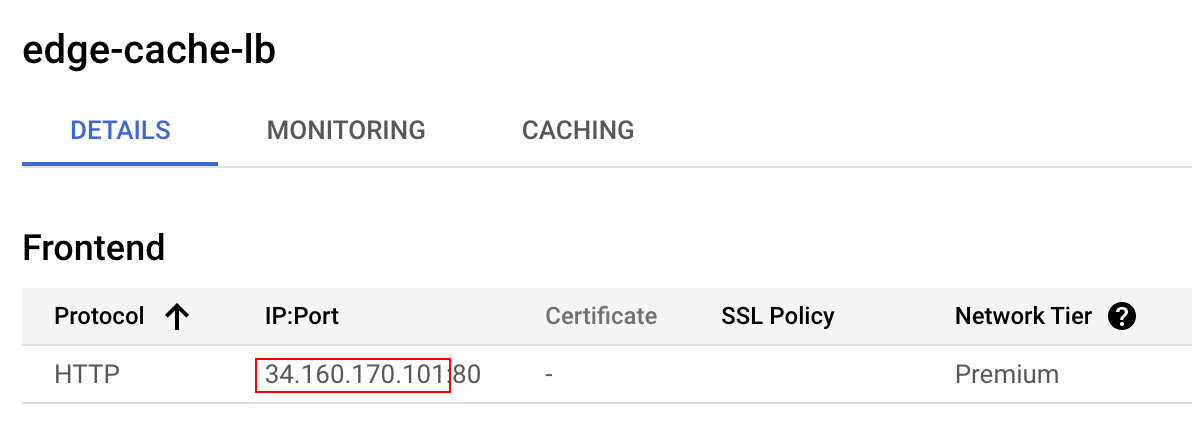
To review and create the HTTP Load Balancer:

1. Click on **Review and finalize**.
2. Review the **Backend services** and **Frontend**.
3. Click on **Create**.

Get Load Balancer IP

To get the Load Balancer IP from the console:

* Click the load balancer name in the list of load balancers for the project. Note the IPv4 address of the load balancer for the next task. Refer to it as [LOAD\_BALANCER\_IP].



Query the Load Balancer

After a couple minutes, attempt to query the load balancer for the object you uploaded. You will need the load balancer IP address and the name of the image.

1. Run the following from CloudShell and replace the LOAD\_BALANCER\_IP with the IPv4 address of the load balancer:

curl -svo /dev/null http://LOAD\_BALANCER\_IP/google.png

Copied!

content\_copy

**Note:**It might take up to 5 minutes to access the HTTP Load Balancer.

**Output:**

student-cloudshell% curl -svo /dev/null http://34.98.81.123/google.png

\* Trying 34.98.81.123...

\* TCP\_NODELAY set

\* Connected to 34.98.81.123 (34.98.81.123) port 80 (#0)

> GET /google.png HTTP/1.1

> Host: YOUR\_IP

> User-Agent: curl/7.64.1

> Accept: \*/\*

>

< HTTP/1.1 200 OK

< X-GUploader-UploadID: ADPycdtoILI76KVsvBvdVGvSfzaxys1m3zYqCepBrmJxAI48ni24cWCRIdNu-53PX3DS6iycxp6xwFbMpwtcHHZQUQmEBxAgng

< Expires: Mon, 13 Dec 2021 22:58:26 GMT

< Date: Mon, 13 Dec 2021 21:58:26 GMT

< Cache-Control: public, max-age=3600

< Last-Modified: Mon, 13 Dec 2021 21:45:57 GMT

< ETag: "8f9327db2597fa57d2f42b4a6c5a9855"

< x-goog-generation: 1639431957957903

< x-goog-metageneration: 2

< x-goog-stored-content-encoding: identity

< x-goog-stored-content-length: 5969

< Content-Type: image/png

< x-goog-hash: crc32c=TeiHTA==

< x-goog-hash: md5=j5Mn2yWX+lfS9CtKbFqYVQ==

< x-goog-storage-class: STANDARD

< Accept-Ranges: bytes

< Content-Length: 5969

< Server: UploadServer

1. Validate that your content is being served from the CDN via CDN or Load Balancing Monitoring. You should be able to get close to a 100% hit ratio.

If you need to run a few queries you can run this command in CloudShell:

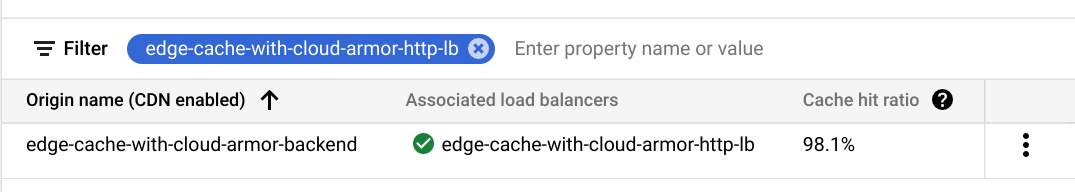
for i in `seq 1 50`; do curl http://LOAD\_BALANCER\_IP/google.png; done

Copied!

content\_copy

Confirm content served by Cloud CDN

* Validate that you are serving traffic from CDN by navigating to **Network Services > CDN**.



Click **Check my progress** to verify the objective.

Create a Load Balancer

Check my progress

**Task 3. Delete the object from Cloud Storage bucket**

Now that the cache is populated, delete the object from the bucket, reinforcing that you are applying the policy to the cache and not the backend.

1. Navigate to **Cloud Storage > Bucket Name**.
2. Select the object and delete it by clicking the **Delete** button at the top.
3. Click **Delete** at the prompt.

Click **Check my progress** to verify the objective.

Delete the object from Cloud Storage bucket

Check my progress

**Task 4. Create an Edge Security policy for a CDN Cache and validate the policy**

1. Navigate to **Network Security > Cloud Armor** and click **Create Policy**.
2. Set the following values, leave all other values at their defaults:

|  |  |
| --- | --- |
| **Property** | **Value (type value or select option as specified)** |
| Name | edge-security-policy |
| Policy type | Edge security policy |
| Default rule action | Deny |

1. In **Apply policy to targets** section, click **Add Target** and set the following values:

|  |  |
| --- | --- |
| **Property** | **Value** |
| Type 1 | Load balancer backend bucket |
| Backend Bucket target 1 | lb-backend-bucket |

1. Click **Done**.
2. Click **Create Policy**.

Validate Edge Security Policy

Now that you've created an Edge Security Policy in front of our back-end bucket, validate that it works as expected.

Check the security policy

After a few minutes have passed, you can check that the Cloud Armor Policy is running. From the command line, the following command gives you a 403.

* From terminal, enter:

curl -svo /dev/null http://LOAD\_BALANCER\_IP/google.png

Copied!

content\_copy

**Output:**

curl -svo /dev/null http://34.98.81.123/google.png

\* Trying 34.98.81.123...

\* TCP\_NODELAY set

\* Connected to 34.98.81.123 (34.98.81.123) port 80 (#0)

> GET /google.png HTTP/1.1

> Host: YOUR\_IP

> User-Agent: curl/7.64.1

> Accept: \*/\*

>

< HTTP/1.1 403 Forbidden

< X-GUploader-UploadID: ADPycdtS6FtJOGIsiWYDrAAE8VFeQuNutcvbGoQe2t8EZxsuspVtmCjyiTv\_P3CNktroHMOGFXkTCfG-Jj-rUO60ZGPpEbpqcw

< Content-Type: application/xml; charset=UTF-8

< Content-Length: 111

< Date: Mon, 13 Dec 2021 23:09:35 GMT

< Expires: Mon, 13 Dec 2021 23:09:35 GMT

< Cache-Control: private, max-age=0

< Server: UploadServer

Investigate the logs

Next, you check the logs to see the enforced edge security policy.

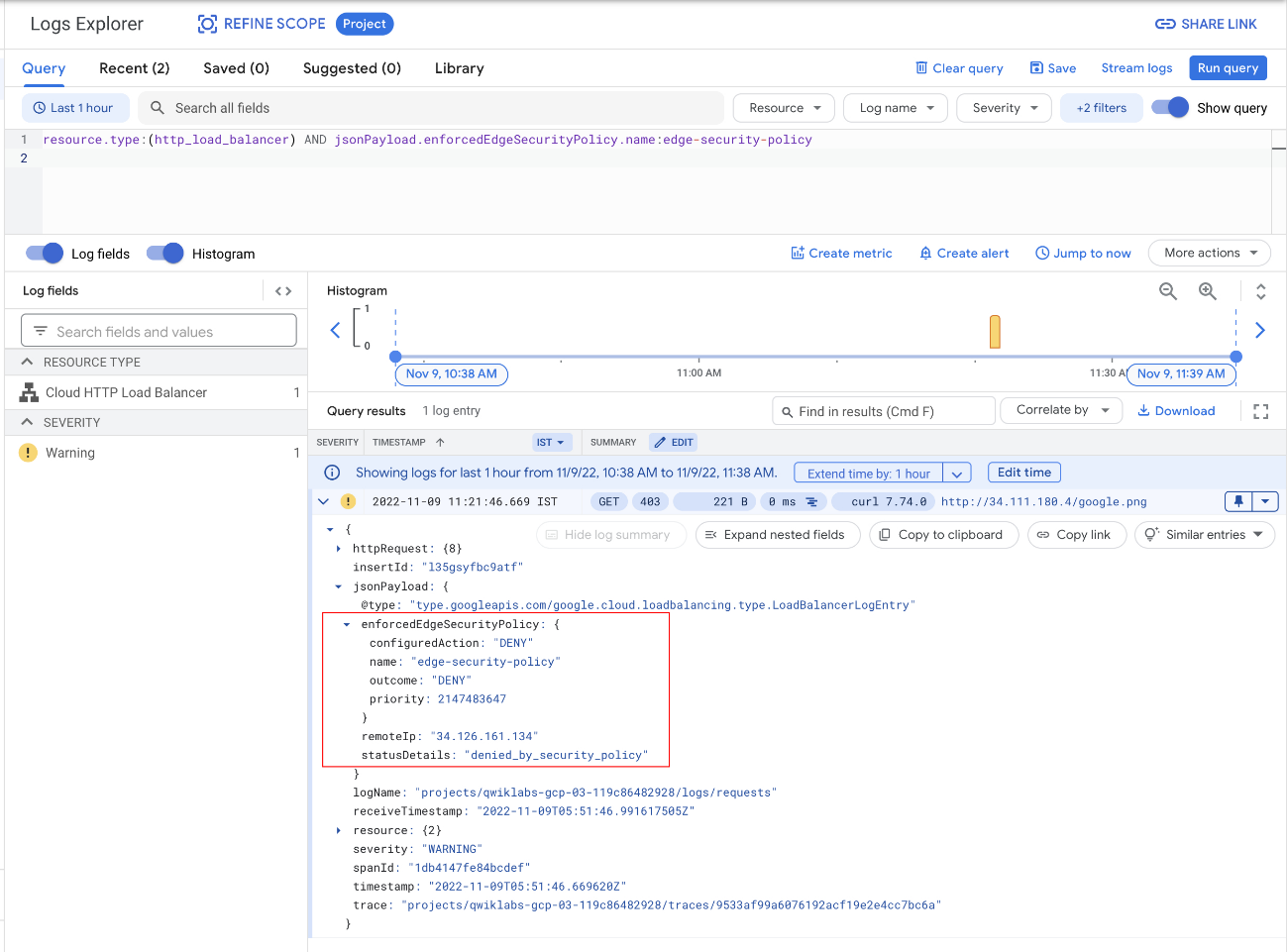
1. Navigate to **Operations > Logging > Logs Explorer**.
2. Enter the below snippet into the query box and click **Run Query**:

resource.type:(http\_load\_balancer) AND jsonPayload.enforcedEdgeSecurityPolicy.name:edge-security-policy

Copied!

content\_copy

1. Note the 403 response and the enforced security policy.



Click **Check my progress** to verify the objective.

Create edge security policy for cloud Armor

Check my progress

Remove the security policy

Now it's time to remove the security policy and query the object from cache.

1. Navigate to **Network Security > Cloud Armor > edge-security-policy > targets**.
2. Select the lb-backend-bucket target and click **Remove** to remove the target bucket.
3. After a few minutes, send another curl to the resource in the storage bucket. You get a 200 response this time. Try it a couple of times if you are getting a 403 status code.
4. From Cloud Shell:

curl -svo /dev/null http://LOAD\_BALANCER\_IP/google.png

Copied!

content\_copy

**Output:**

student-cloudshell% curl -svo /dev/null http://34.98.81.123/google.png

Trying 34.98.81.123...

TCP\_NODELAY set

Connected to 34.98.81.123 (34.98.81.123) port 80 (#0)

GET /google.png HTTP/1.1

Host: YOUR\_IP

User-Agent: curl/7.64.1

Accept: \*/\*

HTTP/1.1 200 OK

X-GUploader-UploadID: ADPycdtI7f49P3MSuZSZ8vl6RwfwmnIDJ59EeSKp7UPvLPawdaiRHXiNWLtseQTxUxceWOvSLvpYmT3pWVkV4qeIP7M

Date: Mon, 13 Dec 2021 23:06:46 GMT

Last-Modified: Mon, 13 Dec 2021 21:45:57 GMT

ETag: "8f9327db2597fa57d2f42b4a6c5a9855"

x-goog-generation: 1639431957957903

x-goog-metageneration: 2

x-goog-stored-content-encoding: identity

x-goog-stored-content-length: 5969

Content-Type: image/png

x-goog-hash: crc32c=TeiHTA==

x-goog-hash: md5=j5Mn2yWX+lfS9CtKbFqYVQ==

x-goog-storage-class: STANDARD

Accept-Ranges: bytes

Content-Length: 5969

Server: UploadServer

Age: 1621

Cache-Control: public,max-age=3600

{ [775 bytes data]

Connection #0 to host 34.98.81.123 left intact

Closing connection 0

**Congratulations!**

You have successfully created a Cloud Storage bucket, uploaded an image to it, bound it to a load balancer, and then enabled Cloud CDN and Edge Security policies on it.