

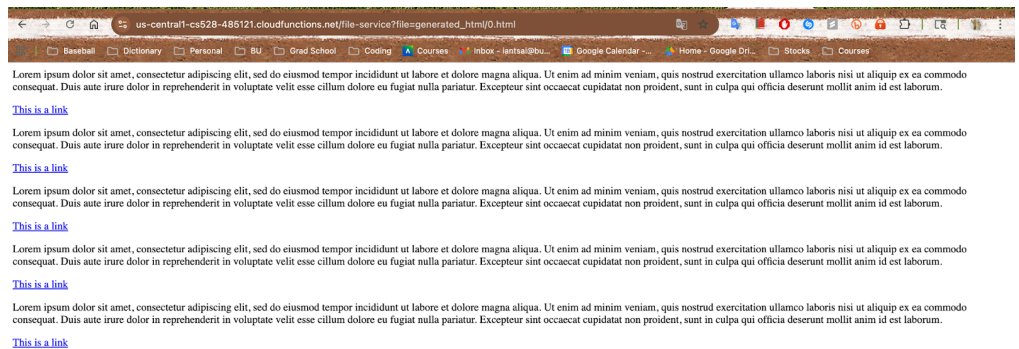
Github Repo: <https://github.com/lanTsai1/CS528-Homeworks/tree/main/HW3>

- Step 1, 2, 3
 - Create service account
 - Cloud Function
 - Two files (reference lecture 6 slides)
 - main.py (contains your code)
 - Used structure log framework from <https://docs.cloud.google.com/functions/1stgendocs/monitoring/logging?hl=en#functions-log-structured-python>
 - requirements.txt
 - Deploy service account
 - 'gcloud functions deploy file-service \
 - --gen2 \
 - --runtime=python312 \
 - --region=us-central1 \
 - --source=. \
 - --entry-point=file_service \
 - --memory=528Mi \
 - --trigger-http \
 - --allow-unauthenticated \
 - --service-account hw3-101@cs528-485121.iam.gserviceaccount.com'
 - Function url = <https://us-central1-cs528-485121.cloudfunctions.net/file-service>
- Step 4
 - ./http-client -d us-central1-cs528-485121.cloudfunctions.net -w 'file-service?file=generated_html' -i 20000 -s -b none -n 100

>	i	2026-02-20 00:01:24.901 EST	file_served						
>	i	2026-02-20 00:01:25.196 EST	GET 200 20.01 KB 159 ms Unknown	https://us-central1-cs528-485121.cloudfunctions.net/file-service?file=generated_html/10533.html					
>	i	2026-02-20 00:01:25.199 EST	incoming_request						
>	i	2026-02-20 00:01:25.355 EST	file_served						
>	i	2026-02-20 00:01:25.533 EST	GET 200 97.43 KB 103 ms Unknown	https://us-central1-cs528-485121.cloudfunctions.net/file-service?file=generated_html/16857.html					
>	i	2026-02-20 00:01:25.537 EST	incoming_request						
>	i	2026-02-20 00:01:25.637 EST	file_served						
>	i	2026-02-20 00:01:25.997 EST	GET 200 81.95 KB 109 ms Unknown	https://us-central1-cs528-485121.cloudfunctions.net/file-service?file=generated_html/6762.html					
>	i	2026-02-20 00:01:26.001 EST	incoming_request						
>	i	2026-02-20 00:01:26.107 EST	file_served						
>	i	2026-02-20 00:01:26.298 EST	GET 200 1.14 KB 118 ms Unknown	https://us-central1-cs528-485121.cloudfunctions.net/file-service?file=generated_html/9183.html					
>	i	2026-02-20 00:01:26.301 EST	incoming_request						
>	i	2026-02-20 00:01:26.417 EST	file_served						
>	i	2026-02-20 00:01:26.654 EST	incoming_request						
>	i	2026-02-20 00:01:26.754 EST	file_served						
>	i	2026-02-20 00:01:26.996 EST	GET 200 168.06 KB 130 ms Unknown	https://us-central1-cs528-485121.cloudfunctions.net/file-service?file=generated_html/3770.html					
>	i	2026-02-20 00:01:27.000 EST	incoming_request						
>	i	2026-02-20 00:01:27.127 EST	file_served						
>	i	2026-02-20 00:01:27.348 EST	GET 200 55.33 KB 107 ms Unknown	https://us-central1-cs528-485121.cloudfunctions.net/file-service?file=generated_html/1068.html					
>	i	2026-02-20 00:01:27.352 EST	incoming_request						
>	i	2026-02-20 00:01:27.456 EST	file_served						

- Step 5 and 6

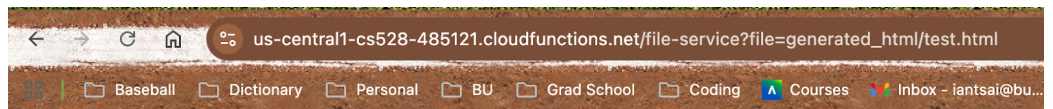
- o 200:



- o 404:

- curl

'https://us-central1-cs528-485121.cloudfunctions.net/file-service?file=generated_html/test.html'



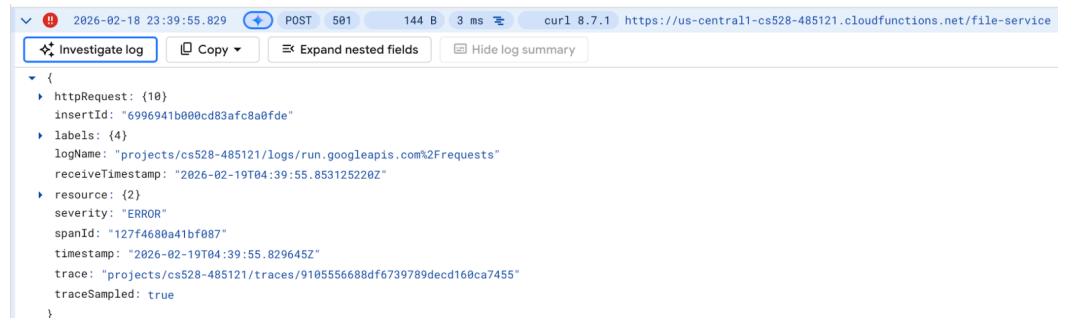
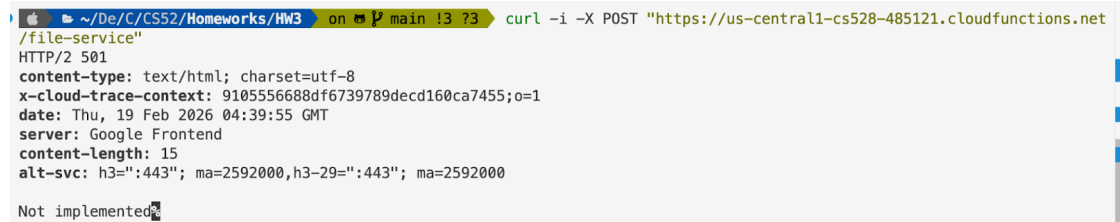
File not found

```
~/De/C/CS52/Homeworks/HW3 on main !3 ?3 curl "https://us-central1-cs528-485121.cloudfunctions.net/file-service?file=generated_html/test.html"
```

File not found



○ 501:



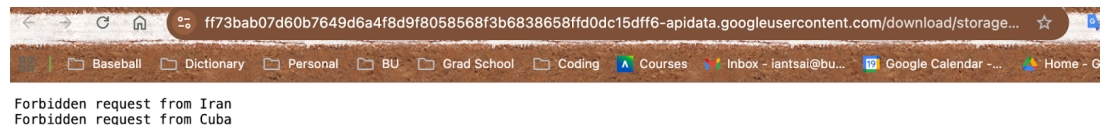
● Step 7 and 8

○ Create pub sub

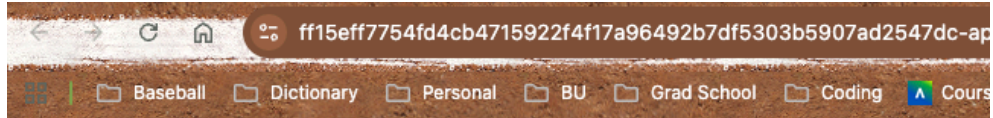
- gcloud pubsub topics create forbidden-topic
- gcloud pubsub subscriptions create forbidden-topic-sub --topic forbidden-topic



○

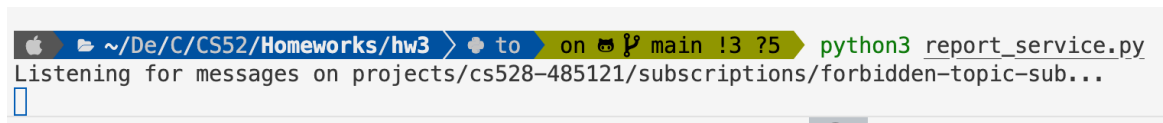


- Updated txt file in gcloud bucket



```
Forbidden request from Iran
Forbidden request from Cuba
Forbidden request from Syria
test from Syria
test from Iran
test from Iran
{"country": "Iran", "message": "test"}
{"country": "Zimbabwe", "message": "forbidden access attempt"}
{"country": "Sudan", "message": "forbidden access attempt"}
{"country": "Cuba", "message": "forbidden access attempt"}
{"country": "Iraq", "message": "forbidden access attempt"}
{"country": "Iran", "message": "forbidden access attempt"}
{"country": "Sudan", "message": "forbidden access attempt"}
{"country": "Libya", "message": "forbidden access attempt"}
{"country": "Iraq", "message": "forbidden access attempt"}
{"country": "Iraq", "message": "forbidden access attempt"}
{"country": "Myanmar", "message": "forbidden access attempt"}
{"country": "Cuba", "message": "forbidden access attempt"}
{"country": "Zimbabwe", "message": "forbidden access attempt"}
{"country": "Iran", "message": "forbidden access attempt"}
{"country": "Zimbabwe", "message": "forbidden access attempt"}
```

○



○

- Ran report_service locally
 - Report_service is the subscriber/receiver of the other end of the pubsub; it receives the message then appends the message to the file in the bucket
- How I impersonated service accounts on a local laptop without using account keys? Explain which mechanism you picked, how it works and why you picked it
 - I used short-lived credentials by setting the service account's email to GOOGLE_IMPERSONATE_SERVICE_ACCOUNT environment variable. Google issues a short-lived token. The script uses this token to perform actions (like writing to the bucket or pulling from Pub/Sub) as if it were the service acc.
 - Resource:
<https://cloud.google.com/blog/topics/developers-practitioners/using-google-cloud-service-account-impersonation-your-terraform-code>

- If I stored my account keys in a json file which is then pushed publicly to github, the account will be stolen. By using this impersonation method, it leaves no sensitive credentials on the local hard drive.