

# EXPLORE FALSE POSITIVES THROUGH INCIDENT DETECTION

## PROJECT DESCRIPTION

In this lab activity I'll recreate the activity that generates a false positive alert. Then, I'll access and analyze the false positive threat using the Security Command Center (SCC) and take action to address it. I'll be using two separate accounts in this lab: one account to trigger the false positive and another account to analyze and remediate the false positive.

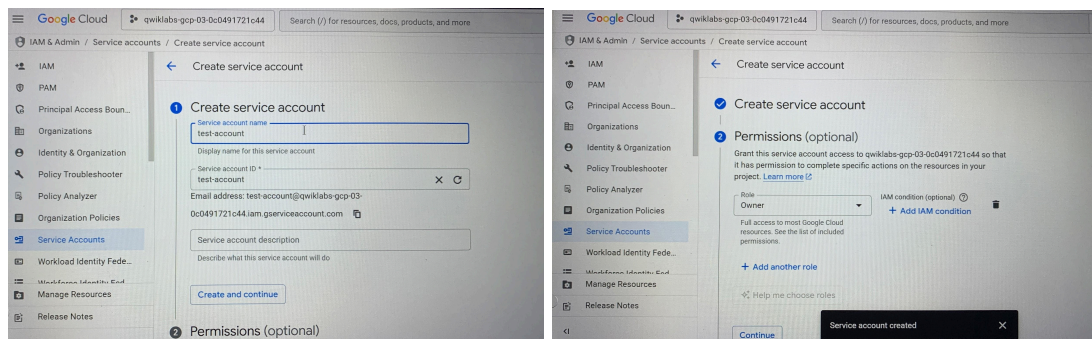
## SCENARIO

During testing of a new service account, a team member inadvertently created a user-managed key with overly broad permissions. This triggered a low-severity security alert due to insecure key management practices.

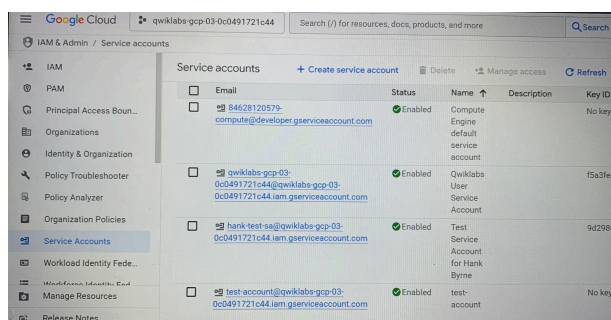
Here's how I addressed this alert and closed it as a false positive;

### First Step : Create a Service Account

Here, I created a Service Account and granted it permissions sufficient to trigger anomalous threat findings in SCC.

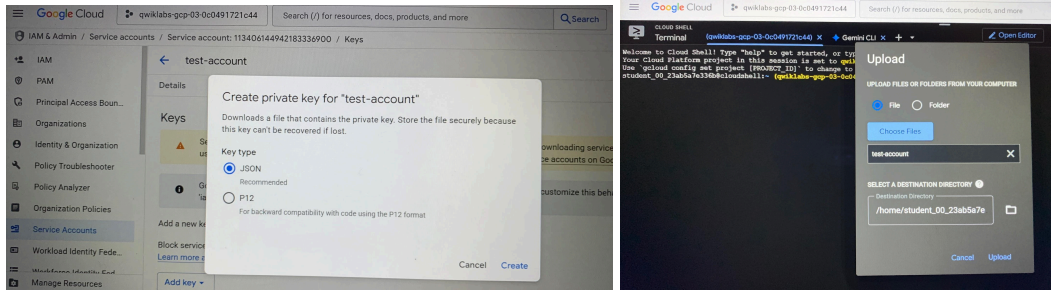


I noticed a **test-account** service account listed in the **service accounts** list.

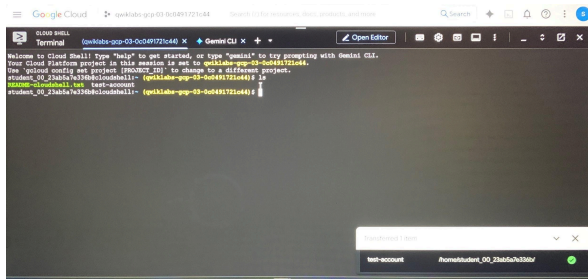


### Second Step : Create a JSON authentication key for the Service Account

Here, I created and downloaded a JSON authentication key for the new service account I created previously. I then used Cloud Shell to upload that key to my google account. This triggered a threat finding in SCC.

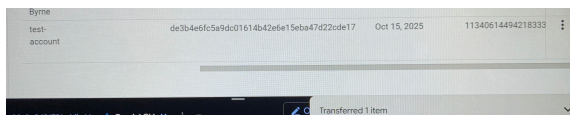


I inputted the “**ls**” command into the cloud shell terminal



This command lists the key file that was previously uploaded

In the **test account** page, in the **key** list, I noticed the key created with the **key creation date** as the current date.



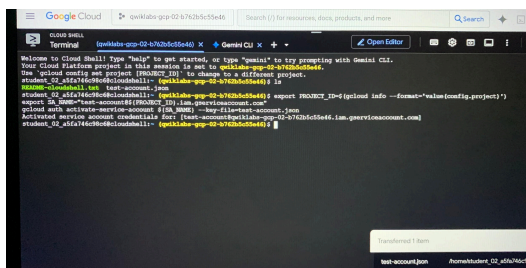
### Third step : Trigger The False Positive Findings

Here, I reconfigured the cloud shell environment to use the new **test-account** service account that I created in the First Step. This triggered a threat Finding in SCC. Then, I assigned excessive permissions to the lab project.

I inputted the following command into the cloud shell terminal;

```
export PROJECT_ID=$(gcloud info --format='value(config.project)')
export SA_NAME="test-account@${PROJECT_ID}.iam.gserviceaccount.com"
gcloud auth activate-service-account ${SA_NAME} --key-file=test-account.json
```

This command activates the new service account

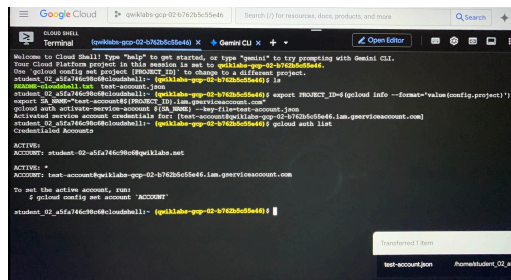


I inputted the following command into the cloud shell terminal;

**gcloud auth list**

This command confirms that you activated the service account and that gcloud is using this service account.

The following output confirms the service account is active:

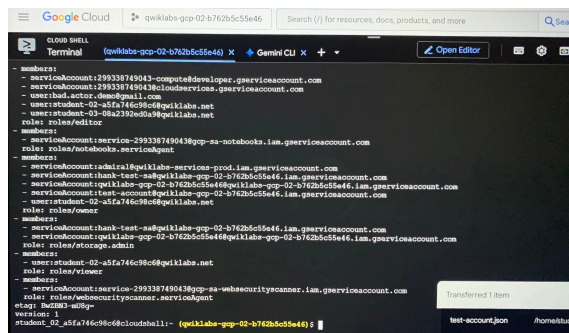


```
student_02_5fa746c9c6@cloudshell: (qwiklabs-gcp-02-b762b5c5e46) X + Gemini CLI X +
Welcome to Cloud Shell! Type "help" to get started, or type "gemini" to try prompting with Gemini CLI.
Your Cloud Platform project in this session is set to qwiklabs-gcp-02-b762b5c5e46.
Use gcloud config set project [PROJECT_ID] to change to a different project.
student_02_5fa746c9c6@cloudshell: (qwiklabs-gcp-02-b762b5c5e46) X + Gemini CLI X +
student_02_5fa746c9c6@cloudshell: (qwiklabs-gcp-02-b762b5c5e46) X + Gemini CLI X +
student_02_5fa746c9c6@cloudshell: (qwiklabs-gcp-02-b762b5c5e46) X + Gemini CLI X +
export SA_NAME="test-account@qwiklabs-gcp-02-b762b5c5e46.iam.gserviceaccount.com"
gcloud auth activate-service-account --key-file /root/.credentials/$SA_NAME
Activated service account credentials for: [test-account@qwiklabs-gcp-02-b762b5c5e46.iam.gserviceaccount.com]
student_02_5fa746c9c6@cloudshell: (qwiklabs-gcp-02-b762b5c5e46) X + Gemini CLI X +
ACTIVE:
ACCOUNT: student-02-5fa746c9c6@qwiklabs.net
ACTIVE: *
ACCOUNT: test-account@qwiklabs-gcp-02-b762b5c5e46.iam.gserviceaccount.com
To set the active account, run:
$ gcloud config set account 'ACCOUNT'
```

I inputted the following command into the Cloud shell terminal:

**export STUDENT2="Google Cloud username 2"**  
**gcloud projects add-iam-policy-binding \$PROJECT\_ID --member user:\$STUDENT2 --role roles/editor**

This command grants the editor role to user 2 so that you can access and remediate the false positive finding in the next task.



```
student_02_5fa746c9c6@cloudshell: (qwiklabs-gcp-02-b762b5c5e46) X + Gemini CLI X +
- members:
  - serviceaccount:299338749043-compute@developer.gserviceaccount.com
  - serviceaccount:299338749043-cloudservices.gserviceaccount.com
  - user:rickad.actor.dms@gmail.com
  - user:student-02-5fa746c9c6@qwiklabs.net
  - user:student-02-5fa746c9c6@qwiklabs.net
  role: roles/editor
- members:
  - serviceaccount:service-299338749043gcp-sa-notebooks.iam.gserviceaccount.com
  role: roles/notebooks.servicemanage
- members:
  - serviceaccount:admin@qwiklabs-services-prod.iam.gserviceaccount.com
  - serviceaccount:hank-test-sa@qwiklabs-gcp-02-b762b5c5e46.iam.gserviceaccount.com
  - serviceaccount:qwiklabs-gcp-02-b762b5c5e46@qwiklabs-gcp-02-b762b5c5e46.iam.gserviceaccount.com
  - serviceaccount:test-account@qwiklabs-gcp-02-b762b5c5e46.iam.gserviceaccount.com
  - user:student-02-5fa746c9c6@qwiklabs.net
  role: roles/owner
- members:
  - serviceaccount:hank-test-sa@qwiklabs-gcp-02-b762b5c5e46.iam.gserviceaccount.com
  - serviceaccount:qwiklabs-gcp-02-b762b5c5e46@qwiklabs-gcp-02-b762b5c5e46.iam.gserviceaccount.com
  role: roles/storage.admin
- members:
  - user:student-02-5fa746c9c6@qwiklabs.net
  role: roles/viewer
- members:
  - serviceaccount:service-299338749043gcp-sa-websecurityscanner.iam.gserviceaccount.com
  role: roles/websecurityscanner.servicemanage
  - user:student-02-5fa746c9c6@qwiklabs.net
  version: 1
  student_02_5fa746c9c6@cloudshell: (qwiklabs-gcp-02-b762b5c5e46) X + Gemini CLI X +
```

## Fourth Step: Sign in as the second user

I switched Google Cloud accounts by logging into the Google Cloud console using the second user account provided in the lab details. This user account was what I used to perform the remaining tasks.

## Fifth Step: View the Threat Finding in SCC

Here, I'll locate and examine the SCC finding generated by the Service Event Threat Detection. This finding is a false positive that was triggered by the activity I generated in steps 1-3.

I displayed the findings in the **Findings query results** panel and was able to answer the following questions

Severity of the alert—> medium

The threat finding class for the alert—> misconfiguration

When is it important to monitor for threats—> whenever your device is on

User managed service account key		
Take action 5 of 30		
Summary Source properties (7) JSON		
What was detected		
Description	Service accounts should not have user-managed keys because they can be easily leaked. <a href="#">Learn more</a>	
State	Active	state
Severity	Medium	severity
Event time	October 16, 2025 at 12:27:35 AM GMT+1	event_time
Create time	October 16, 2025 at 12:27:36 AM GMT+1	create_time
Affected resource		
Resource display name	projects/qwiklabs-gcp-01-17c9571a6b0c/serviceAccounts/test-account@qwiklabs-gcp-01-17c9571a6b0c.iam.gserviceaccount.com/keys/9575b1a42a9e40280b025af2b6000423e96e0a	resource_display_name
Resource full name	/iam.googleapis.com/projects/qwiklabs-gcp-01-17c9571a6b0c/serviceAccounts/1169953171328844861/keys/9575b1a42a9e40280b025af2b6000423e96e0a	resource_full_name
Resource type	google.iam.ServiceAccountKey	resource_type
Parent display name	projects/qwiklabs-gcp-01-17c9571a6b0c/serviceAccounts/test-account@qwiklabs-gcp-01-17c9571a6b0c.iam.gserviceaccount.com	resource_parent_display_name

## Final Step: Fix the Finding

Here, I remediated the false positive by deleting the JSON authentication key for the **test-account** service account.

## CONCLUSION

I used SCC to investigate a false positive and took action to remediate it. As a Cloud security analyst, I'll likely encounter false positive alerts. It's important to understand how and why false positive alerts are triggered and how I can take action to remediate them.