Guide to false positive analysis

False positives are something that you'll likely encounter at some point in your journey as a cloud security analyst. As a reminder, a **false positive** is an alert that incorrectly detects the presence of a threat. It's important to understand the context behind an alert when you're addressing false positive alerts. In the upcoming lab activity, you'll recreate and analyze an Identity and Access Management (IAM) false positive pertaining to a test service account that a Cymbal Bank team member created. In this reading, you'll explore how insecure IAM permissions and key management practices can trigger false positives alerts in Security Command Center (SCC).

Note: Security Command Center has two service tiers: Standard and Premium. The tier type determines which built-in services and their features are available to use. In this program, the labs provide you with access to the Premium tier.

Service accounts

In Google Cloud, projects use and depend on service accounts. A service account is an account that's used by an application or compute instance instead of a person. A service account has a unique email address and can only use cryptographic keys to authenticate. You'll primarily use service accounts to ensure safe and managed connections to APIs and Google Cloud services. For example, a Compute Engine virtual machine (VM) can run as a service account with permissions to access the cloud resources it needs. The service account acts as the identity for the service, and its permissions determine which resources the service can access.

In Google Cloud, there are several types of service accounts, and in this lab you'll focus on *user-managed service accounts*. You create user-managed service accounts in your projects using IAM. You can update, disable, enable, and delete these service accounts. You can also manage other principals' access to these service accounts.

Create a service account using Google Cloud console

Creating a service account is similar to adding a member to a project, but as mentioned earlier, the service account belongs to applications rather than an individual end user.

As a quick reminder, there are three types of roles in Cloud IAM:

- **Basic**: These roles are historically available in the Google Cloud console. These roles are **Owner, Editor,** and **Viewer.**

- **Predefined**: There are roles that provide granular access for a specific service than the basic roles.
- **Custom**: There are roles that you create to tailor permissions to the needs of your organization when predefined roles don't meet your needs.

In the lab activity, you'll create a service account named **test-account,** which you'll use to generate the false positive activity. To create a service account:

1. In the Google Cloud console, click the **Navigation menu.**
2. Click **IAM & Admin** > **Service Accounts**, and click **+ Create Service Account**.
3. Specify the service account name and role to grant to the service account.
4. Click **Done**.

Note: In the lab activity, you'll be using two student user accounts: username 1 and username 2. You'll use username 1 to trigger the false positive alert and username 2 to analyze the false positive in SCC.

Service account keys

Service accounts can't be used for browser-based sign-in and they don't require passwords. Instead, service accounts must authenticate with public / private key pairs. Each service account is associated with a public / private key pair. You can create a key pair, then upload and store the public key to Google Cloud while keeping the private key available only to you.

User-managed keys are extremely powerful credentials and could pose a security risk if compromised. As a preventative measure, it's highly recommended to avoid downloading service account keys and instead use secure key management services to protect keys from being compromised. This is why SCC triggers a security finding when user-managed service accounts use user-managed keys.

Create and upload a service account key

In the lab, you'll create a key pair for the **test-account**. To create a key pair for a service account:

1. Download a JSON file that contains the private key.
2. In the Google Cloud console, click the **Navigation** menu.
3. Click **IAM & Admin** > **Service Accounts,** and select a service account.
4. Click **Keys** > **Add Key** > **Create new key** > **JSON.**
5. Click **Create**.

The downloaded key has the following format, where PRIVATE_KEY is the private portion of the public / private key pair:

```
Unset
{
  "type": "service_account",
  "project_id": "PROJECT_ID",
  "private_key_id": "KEY_ID",
  "private_key": "-----BEGIN PRIVATE
KEY-----\nPRIVATE_KEY\n-----END PRIVATE KEY-----\n",
  "client_email": "SERVICE_ACCOUNT_EMAIL",
  "client_id": "CLIENT_ID",
  "auth_uri": "https://accounts.google.com/o/oauth2/auth",
  "token_uri": "https://accounts.google.com/o/oauth2/token",
  "auth_provider_x509_cert_url":
"https://www.googleapis.com/oauth2/v1/certs",
  "client_x509_cert_url":
"https://www.googleapis.com/robot/v1/metadata/x509/SERVICE_ACCOUN
T_EMAIL"
}
```

Trigger the false positive

To trigger the **User managed service account key** false positive finding for the service account, use the following command:

```
Unset
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
```

Here's what each line of this command does:

- `export PROJECT_ID=$(gcloud info --format='value(config.project)')`

This command retrieves the project ID of the Google Cloud project and stores it in an environment variable named `PROJECT_ID`. When you store values in environment variables you ensure that commands are concise, reusable, easier to maintain, and easier to read. Storing values also saves you the effort of typing out the full name of the project ID, helping you avoid potential typing errors caused from manually typing.

- `export SA_NAME="test-account@${PROJECT_ID}.iam.gserviceaccount.com"`

This command stores the full email address of the `test-account` service account within the current project `PROJECT_ID` and stores it in the `SA_NAME` environment variable.

- `gcloud auth activate-service-account ${SA_NAME} --key-file=test-account.json`

Finally, this command activates the `SA_NAME` test account using the service account's key `test-account.json`.

Analyze and remediate the finding

You'll notice that the **User managed service account key** false positive finding is now listed in SCC. The finding identifies a user-managed service account key that no application is using. This is a security risk because if this key were leaked, it could be used to access your Google Cloud resources.

To fix this issue, delete the user-managed service account key:

1. In the Google Cloud console, navigate to the **Service Accounts** page.
2. Locate the service account associated with the key.
3. Click the **Keys** tab.
4. Click **Delete** next to the key you want to delete.

Key takeaways

False positive alerts require deep investigation including context. In the upcoming lab, you'll investigate why the false positive alert was generated and recreate a false positive to understand how and why it was generated. Being able to analyze false positives will help you distinguish between alerts that require immediate attention and those that you can safely disregard.

Resources for more information

To learn more about service accounts, check out Best practices for using service accounts.