CIS Benchmarks

CIS Google Cloud Platform Foundation Benchmark

v1.2.0 - 05-01-2021

Terms of Use

Please see the below link for our current terms of use:

https://www.cisecurity.org/cis-securesuite/cis-securesuite-membership-terms-of-use/

Table of Contents

Overview

This security configuration benchmark covers foundational elements of Google Cloud Platform. The recommendations detailed here are important security considerations when designing your infrastructure on Google Cloud Platform. Most of the recommendations provided with this release of the benchmark covers security considerations only at individual Project level and not at the organization level.

To obtain the latest version of this guide, please visit http://benchmarks.cisecurity.org. If you have questions, comments, or have identified ways to improve this guide, please write us at BenchmarkInfo@cisecurity.org.

Intended Audience

This document is intended for system and application administrators, security specialists, auditors, help desk, platform deployment, and/or DevOps personnel who plan to develop, deploy, assess, or secure solutions on Google Cloud Platform.

Consensus Guidance

This benchmark was created using a consensus review process comprised of subject matter experts. Consensus participants provide perspective from a diverse set of backgrounds including consulting, software development, audit and compliance, security research, operations, government, and legal.

Each CIS benchmark undergoes two phases of consensus review. The first phase occurs during initial benchmark development. During this phase, subject matter experts convene to discuss, create, and test working drafts of the benchmark. This discussion occurs until consensus has been reached on benchmark recommendations. The second phase begins after the benchmark has been published. During this phase, all feedback provided by the Internet community is reviewed by the consensus team for incorporation in the benchmark. If you are interested in participating in the consensus process, please visit https://workbench.cisecurity.org/.

Typographical Conventions

The following typographical conventions are used throughout this guide:

Convention	Meaning
`Stylized Monospace font`	Used for blocks of code, command, and script examples. Text should be interpreted exactly as presented.
Monospace font	Used for inline code, commands, or examples. Text should be interpreted exactly as presented.
<italic font in brackets>	Italic texts set in angle brackets denote a variable requiring substitution for a real value.
Italic font	Used to denote the title of a book, article, or other publication.
Note	Additional information or caveats

Assessment Status

An assessment status is included for every recommendation. The assessment status indicates whether the given recommendation can be automated or requires manual steps to implement. Both statuses are equally important and are determined and supported as defined below:

Automated

Represents recommendations for which assessment of a technical control can be fully automated and validated to a pass/fail state. Recommendations will include the necessary information to implement automation.

Manual

Represents recommendations for which assessment of a technical control cannot be fully automated and requires all or some manual steps to validate that the configured state is set as expected. The expected state can vary depending on the environment.

Profile Definitions

The following configuration profiles are defined by this Benchmark:

- **Level 1**

  Items in this profile intend to:

  - be practical and prudent;
  - provide a clear security benefit; and
  - not inhibit the utility of the technology beyond acceptable means.

- **Level 2**

  This profile extends the "Level 1" profile. Items in this profile exhibit one or more of the following characteristics:

  - are intended for environments or use cases where security is paramount
  - acts as a defense in depth measure
  - may negatively inhibit the utility or performance of the technology.

Acknowledgements

This benchmark exemplifies the great things a community of users, vendors, and subject matter experts can accomplish through consensus collaboration. The CIS community thanks the entire consensus team with special recognition to the following individuals who contributed greatly to the creation of this guide:

Contributor
Shobha H D
Pravin Goyal
Aditi Sahasrabudhe
Mike Wicks
Jacqueline Kenny
Colin Estep
Anmol Baansal
Robin Drake
Nathael Leblanc
Geoff Uyleman
Jeremy Phillips
David Lu

Editor
Prabhu Angadi
Parag Patil
Pradeep R B
Iulia Ion
Andrew Kiggins

Recommendations

1 Identity and Access Management

This section covers recommendations addressing Identity and Access Management on Google Cloud Platform.

1.1 Ensure that corporate login credentials are used (Automated)

Profile Applicability:

- Level 1

Description:

Use corporate login credentials instead of personal accounts, such as Gmail accounts.

Rationale:

It is recommended fully-managed corporate Google accounts be used for increased visibility, auditing, and controlling access to Cloud Platform resources. Email accounts based outside of the user's organization, such as personal accounts, should not be used for business purposes.

Impact:

None.

Audit:

For each Google Cloud Platform project, list the accounts that have been granted access to that project:

```
gcloud projects get-iam-policy PROJECT_ID
```

Also list the accounts added on each folder:

```
gcloud resource-manager folders get-iam-policy FOLDER_ID
```

And list your organization's IAM policy:

```
gcloud organizations get-iam-policy ORGANIZATION_ID
```

No email accounts outside the organization domain should be granted permissions in the IAM policies. This excludes Google-owned service accounts.

Remediation:

Follow the documentation and setup corporate login accounts.

Prevention:

To ensure that no email addresses outside the organization can be granted IAM permissions to its Google Cloud projects, folders or organization, turn on the Organization

Policy for `Domain Restricted Sharing`. Learn more at: https://cloud.google.com/resource-manager/docs/organization-policy/restricting-domains

Default Value:

By default, no email addresses outside the organization's domain have access to its Google Cloud deployments, but any user email account can be added to the IAM policy for Google Cloud Platform projects, folders, or organizations.

References:

1. https://cloud.google.com/docs/enterprise/best-practices-for-enterprise-organizations#manage-identities
2. https://support.google.com/work/android/answer/6371476
3. https://cloud.google.com/sdk/gcloud/reference/organizations/get-iam-policy
4. https://cloud.google.com/sdk/gcloud/reference/beta/resource-manager/folders/get-iam-policy
5. https://cloud.google.com/sdk/gcloud/reference/projects/get-iam-policy
6. https://cloud.google.com/resource-manager/docs/organization-policy/org-policy-constraints
7. https://cloud.google.com/resource-manager/docs/organization-policy/restricting-domains

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16.2 Configure Centralized Point of Authentication Configure access for all accounts through as few centralized points of authentication as possible, including network, security, and cloud systems.		●	●

1.2 Ensure that multi-factor authentication is enabled for all non-service accounts (Manual)

Profile Applicability:

- Level 1

Description:

Setup multi-factor authentication for Google Cloud Platform accounts.

Rationale:

Multi-factor authentication requires more than one mechanism to authenticate a user. This secures user logins from attackers exploiting stolen or weak credentials.

Audit:

For each Google Cloud Platform project, folder, or organization:
Step 1: Identify non-service accounts.
Step 2: Manually verify that multi-factor authentication for each account is set.

Remediation:

For each Google Cloud Platform project:
Step 1: Identify non-service accounts.
Step 2: Setup multi-factor authentication for each account.

Default Value:

By default, multi-factor authentication is not set.

References:

1. https://cloud.google.com/solutions/securing-gcp-account-u2f
2. https://support.google.com/accounts/answer/185839

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16.3 Require Multi-factor Authentication Require multi-factor authentication for all user accounts, on all systems, whether managed onsite or by a third-party provider.		●	●

1.3 Ensure that Security Key Enforcement is enabled for all admin accounts (Manual)

Profile Applicability:

- Level 2

Description:

Setup Security Key Enforcement for Google Cloud Platform admin accounts.

Rationale:

Google Cloud Platform users with Organization Administrator roles have the highest level of privilege in the organization. These accounts should be protected with the strongest form of two-factor authentication: Security Key Enforcement. Ensure that admins use Security Keys to log in instead of weaker second factors like SMS or one-time passwords (OTP). Security Keys are actual physical keys used to access Google Organization Administrator Accounts. They send an encrypted signature rather than a code, ensuring that logins cannot be phished.

Impact:

If an organization administrator loses access to their security key, the user could lose access to their account. For this reason, it is important to set up backup security keys.

Audit:

Step 1: Identify users with Organization Administrator privileges:

```
gcloud organizations get-iam-policy ORGANIZATION_ID
```

Look for members granted the role "roles/resourcemanager.organizationAdmin".
Step 2: Manually verify that Security Key Enforcement has been enabled for each account.

Remediation:

Step 1: Identify users with the Organization Administrator role.
Step 2: Setup Security Key Enforcement for each account. Learn more at: https://cloud.google.com/security-key/

Default Value:

By default, Security Key Enforcement is not enabled for Organization Administrators.

References:

1. https://cloud.google.com/security-key/
2. https://gsuite.google.com/learn-more/key_for_working_smarter_faster_and_more_securely.html

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16.3 Require Multi-factor Authentication Require multi-factor authentication for all user accounts, on all systems, whether managed onsite or by a third-party provider.		●	●

1.4 Ensure that there are only GCP-managed service account keys for each service account (Automated)

Profile Applicability:

- Level 1

Description:

User managed service accounts should not have user-managed keys.

Rationale:

Anyone who has access to the keys will be able to access resources through the service account. GCP-managed keys are used by Cloud Platform services such as App Engine and Compute Engine. These keys cannot be downloaded. Google will keep the keys and automatically rotate them on an approximately weekly basis. User-managed keys are created, downloadable, and managed by users. They expire 10 years from creation.

For user-managed keys, the user has to take ownership of key management activities which include:

- Key storage
- Key distribution
- Key revocation
- Key rotation
- Protecting the keys from unauthorized users
- Key recovery

Even with key owner precautions, keys can be easily leaked by common development malpractices like checking keys into the source code or leaving them in the Downloads directory, or accidentally leaving them on support blogs/channels.

It is recommended to prevent user-managed service account keys.

Impact:

Deleting user-managed Service Account Keys may break communication with the applications using the corresponding keys.

Audit:

From Console:

1. Go to the IAM page in the GCP Console using `https://console.cloud.google.com/iam-admin/iam`
2. In the left navigation pane, click `Service accounts`. All service accounts and their corresponding keys are listed.
3. Click the service accounts and check if keys exist.

From Command Line:
List All the service accounts:

```
gcloud iam service-accounts list
```

Identify user-managed service accounts as such account `EMAIL` ends with `iam.gserviceaccount.com`
For each user-managed service account, list the keys managed by the user:

```
gcloud iam service-accounts keys list --iam-account=<Service Account> --
managed-by=user
```

No keys should be listed.

Remediation:

From Console:

1. Go to the IAM page in the GCP Console using `https://console.cloud.google.com/iam-admin/iam`
2. In the left navigation pane, click `Service accounts`. All service accounts and their corresponding keys are listed.
3. Click the service account.
4. Click the `edit` and delete the keys.

From CLI:
To delete a user managed Service Account Key,

```
gcloud iam service-accounts keys delete --iam-account=<user-managed-service-
account-EMAIL> <KEY-ID>
```

Prevention:
You can disable service account key creation through the `Disable service account key creation` Organization policy by visiting https://console.cloud.google.com/iam-admin/orgpolicies/iam-disableServiceAccountKeyCreation. Learn more at: https://cloud.google.com/resource-manager/docs/organization-policy/restricting-service-accounts
In addition, if you do not need to have service accounts in your project, you can also prevent the creation of service accounts through the `Disable service account creation`

Organization policy: https://console.cloud.google.com/iam-admin/orgpolicies/iam-disableServiceAccountCreation.

Default Value:

By default, there are no user-managed keys created for user-managed service accounts.

References:

1. https://cloud.google.com/iam/docs/understanding-service-accounts#managing_service_account_keys
2. https://cloud.google.com/resource-manager/docs/organization-policy/restricting-service-accounts

Additional Information:

A user-managed key cannot be created on GCP-Managed Service Accounts.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

1.5 Ensure that Service Account has no Admin privileges (Automated)

Profile Applicability:

- Level 1

Description:

A service account is a special Google account that belongs to an application or a VM, instead of to an individual end-user. The application uses the service account to call the service's Google API so that users aren't directly involved. It's recommended not to use admin access for ServiceAccount.

Rationale:

Service accounts represent service-level security of the Resources (application or a VM) which can be determined by the roles assigned to it. Enrolling ServiceAccount with Admin rights gives full access to an assigned application or a VM. A ServiceAccount Access holder can perform critical actions like delete, update change settings, etc. without user intervention. For this reason, it's recommended that service accounts not have Admin rights.

Impact:

Removing `*Admin or` *admin`or`Editor`or`Owner` role assignments from service accounts may break functionality that uses impacted service accounts. Required role(s) should be assigned to impacted service accounts in order to restore broken functionalities.

Audit:

From Console:

1. Go to `IAM & admin/IAM` using `https://console.cloud.google.com/iam-admin/iam`
2. Go to the `Members`
3. Ensure that there are no `User-Managed user created service account(s)` with roles containing `*Admin` or `*admin` or role matching `Editor` or role matching `Owner`

From Command Line:

1. Get the policy that you want to modify, and write it to a JSON file:
   gcloud projects get-iam-policy PROJECT_ID --format json > iam.json
2. The contents of the JSON file will look similar to the following. Note that `role` of members group associated with each `serviceaccount` does not contain `*Admin` or `*admin` or does not match `roles/editor` or does not match `roles/owner`.

This recommendation is only applicable to `User-Managed user-created` service accounts. These accounts have the nomenclature: `SERVICE_ACCOUNT_NAME@PROJECT_ID.iam.gserviceaccount.com`. Note that some Google-managed, Google-created service accounts have the same naming format, and should be excluded (e.g., `appsdev-apps-dev-script-auth@system.gserviceaccount.com` which needs the Owner role).

Sample Json output:

```
{
"bindings": [
{
   "members": [
      "serviceAccount:our-project-123@appspot.gserviceaccount.com",
    ],
    "role": "roles/appengine.appAdmin"
},
{
    "members": [
      "user:email1@gmail.com"
    ],
    "role": "roles/owner"
},
{
   "members": [
      "serviceAccount:our-project-123@appspot.gserviceaccount.com",
      "serviceAccount:123456789012-compute@developer.gserviceaccount.com"
    ],
    "role": "roles/editor"
}
],
"etag": "BwUjMhCsNvY=",
"version": 1
}
```

Remediation:

From Console

1. Go to `IAM & admin/IAM` using `https://console.cloud.google.com/iam-admin/iam`
2. Go to the `Members`
3. Identify `User-Managed user created` service account with roles containing `*Admin` or `*admin` or role matching `Editor` or role matching `Owner`
4. Click the `Delete bin` icon to remove the role from the member (service account in this case)

From Command Line:

gcloud projects get-iam-policy PROJECT_ID --format json > iam.json

1. Using a text editor, Remove `Role` which contains `roles/*Admin` or `roles/*admin` or matched `roles/editor` or matches 'roles/owner`. Add a role to the bindings array that defines the group members and the role for those members.

For example, to grant the role roles/appengine.appViewer to the `ServiceAccount` which is roles/editor, you would change the example shown below as follows:

```
{
"bindings": [
{
   "members": [
     "serviceAccount:our-project-123@appspot.gserviceaccount.com",
    ],
    "role": "roles/appengine.appViewer"
},
{
    "members": [
     "user:email1@gmail.com"
    ],
    "role": "roles/owner"
   },
{
    "members": [
      "serviceAccount:our-project-123@appspot.gserviceaccount.com",
      "serviceAccount:123456789012-compute@developer.gserviceaccount.com"
    ],
    "role": "roles/editor"
}
],
"etag": "BwUjMhCsNvY="
}
```

2. Update the project's IAM policy:
   gcloud projects set-iam-policy PROJECT_ID iam.json

Default Value:

User Managed (and not user-created) default service accounts have the `Editor (roles/editor)` role assigned to them to support GCP services they offer.

By default, there are no roles assigned to `User Managed User created` service accounts.

References:

1. https://cloud.google.com/sdk/gcloud/reference/iam/service-accounts/
2. https://cloud.google.com/iam/docs/understanding-roles
3. https://cloud.google.com/iam/docs/understanding-service-accounts

Additional Information:

Default (user-managed but not user-created) service accounts have the `Editor (roles/editor)` role assigned to them to support GCP services they offer. Such Service accounts are: `PROJECT_NUMBER-compute@developer.gserviceaccount.com`, `PROJECT_ID@appspot.gserviceaccount.com`.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

1.6 Ensure that IAM users are not assigned the Service Account User or Service Account Token Creator roles at project level (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to assign the `Service Account User (iam.serviceAccountUser)` and `Service Account Token Creator (iam.serviceAccountTokenCreator)` roles to a user for a specific service account rather than assigning the role to a user at project level.

Rationale:

A service account is a special Google account that belongs to an application or a virtual machine (VM), instead of to an individual end-user. Application/VM-Instance uses the service account to call the service's Google API so that users aren't directly involved. In addition to being an identity, a service account is a resource that has IAM policies attached to it. These policies determine who can use the service account.

Users with IAM roles to update the App Engine and Compute Engine instances (such as App Engine Deployer or Compute Instance Admin) can effectively run code as the service accounts used to run these instances, and indirectly gain access to all the resources for which the service accounts have access. Similarly, SSH access to a Compute Engine instance may also provide the ability to execute code as that instance/Service account.

Based on business needs, there could be multiple user-managed service accounts configured for a project. Granting the `iam.serviceAccountUser` or `iam.serviceAserviceAccountTokenCreatorccountUser` roles to a user for a project gives the user access to all service accounts in the project, including service accounts that may be created in the future. This can result in elevation of privileges by using service accounts and corresponding `Compute Engine instances`.

In order to implement `least privileges` best practices, IAM users should not be assigned the `Service Account User` or `Service Account Token Creator` roles at the project level. Instead, these roles should be assigned to a user for a specific service account, giving that user access to the service account. The `Service Account User` allows a user to bind a service account to a long-running job service, whereas the `Service Account Token Creator` role allows a user to directly impersonate (or assert) the identity of a service account.

Impact:

After revoking `Service Account User` or `Service Account Token Creator` roles at the project level from all impacted user account(s), these roles should be assigned to a user(s) for specific service account(s) according to business needs.

Audit:

From Console:

1. Go to the IAM page in the GCP Console by visiting https://console.cloud.google.com/iam-admin/iam
2. Click on the filter table text bar, Type `Role: Service Account User`.
3. Ensure no user is listed as a result of the filter.
4. Click on the filter table text bar, Type `Role: Service Account Token Creator`.
5. Ensure no user is listed as a result of the filter.

From Command Line:
To ensure IAM users are not assigned Service Account User role at the project level:

```
gcloud projects get-iam-policy PROJECT_ID --format json | jq
'.bindings[].role' | grep "roles/iam.serviceAccountUser"

gcloud projects get-iam-policy PROJECT_ID --format json | jq
'.bindings[].role' | grep "roles/iam.serviceAccountTokenCreator"
```

These commands should not return any output.

Remediation:

From Console:

1. Go to the IAM page in the GCP Console by visiting: https://console.cloud.google.com/iam-admin/iam.
2. Click on the filter table text bar. Type `Role: Service Account User`
3. Click the `Delete Bin` icon in front of the role `Service Account User` for every user listed as a result of a filter.
4. Click on the filter table text bar. Type `Role: Service Account Token Creator`
5. Click the `Delete Bin` icon in front of the role `Service Account Token Creator` for every user listed as a result of a filter.

From Command Line:

1. Using a text editor, remove the bindings with the `roles/iam.serviceAccountUser` or `roles/iam.serviceAccountTokenCreator`.

For example, you can use the iam.json file shown below as follows:

```
{
"bindings": [
{
   "members": [
     "serviceAccount:our-project-123@appspot.gserviceaccount.com",
    ],
    "role": "roles/appengine.appViewer"
},
{
    "members": [
     "user:email1@gmail.com"
    ],
    "role": "roles/owner"
   },
{
    "members": [
      "serviceAccount:our-project-123@appspot.gserviceaccount.com",
      "serviceAccount:123456789012-compute@developer.gserviceaccount.com"
    ],
    "role": "roles/editor"
}
],
"etag": "BwUjMhCsNvY="
    }
```

2. Update the project's IAM policy:

```
gcloud projects set-iam-policy PROJECT_ID iam.json
```

Default Value:

By default, users do not have the Service Account User or Service Account Token Creator role assigned at project level.

References:

1. https://cloud.google.com/iam/docs/service-accounts
2. https://cloud.google.com/iam/docs/granting-roles-to-service-accounts
3. https://cloud.google.com/iam/docs/understanding-roles
4. https://cloud.google.com/iam/docs/granting-changing-revoking-access
5. https://console.cloud.google.com/iam-admin/iam

Additional Information:

To assign the role `roles/iam.serviceAccountUser` or `roles/iam.serviceAccountTokenCreator` to a user role on a service account instead of a project:

1. Go to https://console.cloud.google.com/projectselector/iam-admin/serviceaccounts
2. Select `Target Project`
3. Select `target service account`. Click `Permissions` on the top bar. It will open permission pane on right side of the page
4. Add desired members with `Service Account User` or `Service Account Token Creator` role.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●
v7	16 Account Monitoring and Control Account Monitoring and Control			

1.7 Ensure user-managed/external keys for service accounts are rotated every 90 days or less (Automated)

Profile Applicability:

- Level 1

Description:

Service Account keys consist of a key ID (Private_key_Id) and Private key, which are used to sign programmatic requests users make to Google cloud services accessible to that particular service account. It is recommended that all Service Account keys are regularly rotated.

Rationale:

Rotating Service Account keys will reduce the window of opportunity for an access key that is associated with a compromised or terminated account to be used. Service Account keys should be rotated to ensure that data cannot be accessed with an old key that might have been lost, cracked, or stolen.

Each service account is associated with a key pair managed by Google Cloud Platform (GCP). It is used for service-to-service authentication within GCP. Google rotates the keys daily.

GCP provides the option to create one or more user-managed (also called external key pairs) key pairs for use from outside GCP (for example, for use with Application Default Credentials). When a new key pair is created, the user is required to download the private key (which is not retained by Google). With external keys, users are responsible for keeping the private key secure and other management operations such as key rotation. External keys can be managed by the IAM API, gcloud command-line tool, or the Service Accounts page in the Google Cloud Platform Console. GCP facilitates up to 10 external service account keys per service account to facilitate key rotation.

Impact:

Rotating service account keys will break communication for dependent applications. Dependent applications need to be configured manually with the new key `ID` displayed in the `Service account keys`section and the `private key` downloaded by the user.

Audit:

From Console:

1. Go to `APIs & Services\Credentials` using `https://console.cloud.google.com/apis/credentials`
2. In the section `Service Account Keys`, for every External (user-managed) service account key listed ensure the `creation date` is within the past 90 days.

From Command Line:

1. List all Service accounts from a project.

```
gcloud iam service-accounts list
```

2. For every service account list service account keys.

```
gcloud iam service-accounts keys list --iam-account
[Service_Account_Email_Id] --format=json
```

3. Ensure every service account key for a service account has a `"validAfterTime"` value within the past 90 days.

Remediation:

From Console:
Delete any external (user-managed) Service Account Key older than 90 days:

1. Go to `APIs & Services\Credentials` using `https://console.cloud.google.com/apis/credentials`
2. In the Section `Service Account Keys`, for every external (user-managed) service account key where `creation date` is greater than or equal to the past 90 days, click `Delete Bin Icon` to `Delete Service Account key`

Create a new external (user-managed) Service Account Key for a Service Account:

1. Go to `APIs & Services\Credentials` using `https://console.cloud.google.com/apis/credentials`
2. Click `Create Credentials` and Select `Service Account Key`.
3. Choose the service account in the drop-down list for which an External (user-managed) Service Account key needs to be created.
4. Select the desired key type format among `JSON` or `P12`.
5. Click `Create`. It will download the `private key`. Keep it safe.
6. Click `Close` if prompted.
7. The site will redirect to the `APIs & Services\Credentials` page. Make a note of the new `ID` displayed in the `Service account keys` section.

Default Value:

GCP does not provide an automation option for External (user-managed) Service key rotation.

References:

1. https://cloud.google.com/iam/docs/understanding-service-accounts#managing_service_account_keys
2. https://cloud.google.com/sdk/gcloud/reference/iam/service-accounts/keys/list
3. https://cloud.google.com/iam/docs/service-accounts

Additional Information:

For user-managed Service Account key(s), key management is entirely the user's responsibility.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

1.8 Ensure that Separation of duties is enforced while assigning service account related roles to users (Manual)

Profile Applicability:

- Level 2

Description:

It is recommended that the principle of 'Separation of Duties' is enforced while assigning service-account related roles to users.

Rationale:

The built-in/predefined IAM role `Service Account admin` allows the user/identity to create, delete, and manage service account(s). The built-in/predefined IAM role `Service Account User` allows the user/identity (with adequate privileges on Compute and App Engine) to assign service account(s) to Apps/Compute Instances.

Separation of duties is the concept of ensuring that one individual does not have all necessary permissions to be able to complete a malicious action. In Cloud IAM - service accounts, this could be an action such as using a service account to access resources that user should not normally have access to.

Separation of duties is a business control typically used in larger organizations, meant to help avoid security or privacy incidents and errors. It is considered best practice.

No user should have `Service Account Admin` and `Service Account User`roles assigned at the same time.

Impact:

The removed role should be assigned to a different user based on business needs.

Audit:

From Console:

1. Go to `IAM & Admin/IAM` using `https://console.cloud.google.com/iam-admin/iam`.
2. Ensure no member has the roles `Service Account Admin` and `Service account User` assigned together.

From Command Line:

1. List all users and role assignments:

```
gcloud projects get-iam-policy [Project_ID]
```

2. Ensure that there are no common users found in the member section for roles `roles/iam.serviceAccountAdmin` and `roles/iam.serviceAccountUser`

Remediation:

From Console:

1. Go to `IAM & Admin/IAM` using `https://console.cloud.google.com/iam-admin/iam`.
2. For any member having both `Service Account Admin` and `Service account User` roles granted/assigned, click the `Delete Bin` icon to remove either role from the member.
   Removal of a role should be done based on the business requirements.

References:

1. https://cloud.google.com/iam/docs/service-accounts
2. https://cloud.google.com/iam/docs/understanding-roles
3. https://cloud.google.com/iam/docs/granting-roles-to-service-accounts

Additional Information:

Users granted with Owner (roles/owner) and Editor (roles/editor) have privileges equivalent to `Service Account Admin` and `Service Account User`. To avoid the misuse, Owner and Editor roles should be granted to very limited users and Use of these primitive privileges should be minimal. These requirements are addressed in separate recommendations.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

1.9 Ensure that Cloud KMS cryptokeys are not anonymously or publicly accessible (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended that the IAM policy on Cloud KMS `cryptokeys` should restrict anonymous and/or public access.

Rationale:

Granting permissions to `allUsers` or `allAuthenticatedUsers` allows anyone to access the dataset. Such access might not be desirable if sensitive data is stored at the location. In this case, ensure that anonymous and/or public access to a Cloud KMS `cryptokey` is not allowed.

Impact:

Removing the binding for `allUsers` and `allAuthenticatedUsers` members denies accessing `cryptokeys` to anonymous or public users.

Audit:

From Command Line:

1. List all Cloud KMS `Cryptokeys`.

```
gcloud kms keys list --keyring=[key_ring_name] --location=global --
format=json | jq '.[].name'
```

2. Ensure the below command's output does not contain `allUsers` and `allAuthenticatedUsers`.

```
gcloud kms keys get-iam-policy [key_name] --keyring=[key_ring_name] --
location=global --format=json | jq '.bindings[].members[]'
```

Remediation:

From Command Line:

1. List all Cloud KMS `Cryptokeys`.

```
gcloud kms keys list --keyring=[key_ring_name] --location=global --
format=json | jq '.[].name'
```

2. Remove IAM policy binding for a KMS key to remove access to `allUsers` and `allAuthenticatedUsers` using the below command.

```
gcloud kms keys remove-iam-policy-binding [key_name] --
keyring=[key_ring_name] --location=global --member='allAuthenticatedUsers' --
role='[role]'

gcloud kms keys remove-iam-policy-binding [key_name] --
keyring=[key_ring_name] --location=global --member='allUsers' --role='[role]'
```

Default Value:

By default Cloud KMS does not allow access to `allUsers` or `allAuthenticatedUsers`.

References:

1. https://cloud.google.com/sdk/gcloud/reference/kms/keys/remove-iam-policy-binding
2. https://cloud.google.com/sdk/gcloud/reference/kms/keys/set-iam-policy
3. https://cloud.google.com/sdk/gcloud/reference/kms/keys/get-iam-policy
4. https://cloud.google.com/kms/docs/object-hierarchy#key_resource_id

Additional Information:

[key_ring_name] : Is the resource ID of the key ring, which is the fully-qualified Key ring name. This value is case-sensitive and in the form: projects/PROJECT_ID/locations/LOCATION/keyRings/KEY_RING

You can retrieve the key ring resource ID using the Cloud Console:

1. Open the `Cryptographic Keys` page in the Cloud Console.
2. For the key ring whose resource ID you are retrieving, click the `More icon (3 vertical dots)`.
3. Click `Copy Resource ID`. The resource ID for the key ring is copied to your clipboard.

[key_name] : Is the resource ID of the key, which is the fully-qualified CryptoKey name. This value is case-sensitive and in the form: projects/PROJECT_ID/locations/LOCATION/keyRings/KEY_RING/cryptoKeys/KEY

You can retrieve the key resource ID using the Cloud Console:

1. Open the `Cryptographic Keys` page in the Cloud Console.

2. Click the name of the key ring that contains the key.
3. For the key whose resource ID you are retrieving, click the `More icon (3 vertical dots)`.
4. Click `Copy Resource ID`. The resource ID for the key is copied to your clipboard.

[role] : The role to remove the member from.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

1.10 Ensure KMS encryption keys are rotated within a period of 90 days (Automated)

Profile Applicability:

- Level 1

Description:

Google Cloud Key Management Service stores cryptographic keys in a hierarchical structure designed for useful and elegant access control management.

The format for the rotation schedule depends on the client library that is used. For the gcloud command-line tool, the next rotation time must be in `ISO` or `RFC3339` format, and the rotation period must be in the form `INTEGER[UNIT]`, where units can be one of seconds (s), minutes (m), hours (h) or days (d).

Rationale:

Set a key rotation period and starting time. A key can be created with a specified `rotation period`, which is the time between when new key versions are generated automatically. A key can also be created with a specified next rotation time. A key is a named object representing a `cryptographic key` used for a specific purpose. The key material, the actual bits used for `encryption`, can change over time as new key versions are created.

A key is used to protect some `corpus of data`. A collection of files could be encrypted with the same key and people with `decrypt` permissions on that key would be able to decrypt those files. Therefore, it's necessary to make sure the `rotation period` is set to a specific time.

Impact:

After a successful key rotation, the older key version is required in order to decrypt the data encrypted by that previous key version.

Audit:

From Console:

1. Go to `Cryptographic Keys` by visiting: https://console.cloud.google.com/security/kms.
2. Click on each key ring, then ensure each key in the keyring has `Next Rotation` set for less than 90 days from the current date.

From Command Line:

1. Ensure rotation is scheduled by `ROTATION_PERIOD` and `NEXT_ROTATION_TIME` for each key :

```
gcloud kms keys list --keyring=<KEY_RING> --location=<LOCATION> --
format=json'(rotationPeriod)'
```

Ensure outcome values for `rotationPeriod` and `nextRotationTime` satisfy the below criteria:

```
rotationPeriod is <= 129600m

rotationPeriod is <= 7776000s

rotationPeriod is <= 2160h

rotationPeriod is <= 90d
```

`nextRotationTime is <= 90days` from current DATE

Remediation:

From Console:

1. Go to `Cryptographic Keys` by visiting: https://console.cloud.google.com/security/kms.
2. Click on the specific key ring
3. From the list of keys, choose the specific key and Click on `Right side pop up the blade (3 dots)`.
4. Click on `Edit rotation period`.
5. On the pop-up window, `Select a new rotation period` in days which should be less than 90 and then choose `Starting on` date (date from which the rotation period begins).

From Command Line:

1. Update and schedule rotation by `ROTATION_PERIOD` and `NEXT_ROTATION_TIME` for each key:

```
gcloud kms keys update new --keyring=KEY_RING --location=LOCATION --next-
rotation-time=NEXT_ROTATION_TIME --rotation-period=ROTATION_PERIOD
```

Default Value:

By default, KMS encryption keys are rotated every 90 days.

References:

1. https://cloud.google.com/kms/docs/key-rotation#frequency_of_key_rotation
2. https://cloud.google.com/kms/docs/re-encrypt-data

Additional Information:

- Key rotation does NOT re-encrypt already encrypted data with the newly generated key version. If you suspect unauthorized use of a key, you should re-encrypt the data protected by that key and then disable or schedule destruction of the prior key version.
- It is not recommended to rely solely on irregular rotation, but rather to use irregular rotation if needed in conjunction with a regular rotation schedule.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

1.11 Ensure that Separation of duties is enforced while assigning KMS related roles to users (Automated)

Profile Applicability:

- Level 2

Description:

It is recommended that the principle of 'Separation of Duties' is enforced while assigning KMS related roles to users.

Rationale:

The built-in/predefined IAM role `Cloud KMS Admin` allows the user/identity to create, delete, and manage service account(s). The built-in/predefined IAM role `Cloud KMS CryptoKey Encrypter/Decrypter` allows the user/identity (with adequate privileges on concerned resources) to encrypt and decrypt data at rest using an encryption key(s).

The built-in/predefined IAM role `Cloud KMS CryptoKey Encrypter` allows the user/identity (with adequate privileges on concerned resources) to encrypt data at rest using an encryption key(s). The built-in/predefined IAM role `Cloud KMS CryptoKey Decrypter` allows the user/identity (with adequate privileges on concerned resources) to decrypt data at rest using an encryption key(s).

Separation of duties is the concept of ensuring that one individual does not have all necessary permissions to be able to complete a malicious action. In Cloud KMS, this could be an action such as using a key to access and decrypt data a user should not normally have access to. Separation of duties is a business control typically used in larger organizations, meant to help avoid security or privacy incidents and errors. It is considered best practice.

No user(s) should have `Cloud KMS Admin` and any of the `Cloud KMS CryptoKey Encrypter/Decrypter`, `Cloud KMS CryptoKey Encrypter`, `Cloud KMS CryptoKey Decrypter` roles assigned at the same time.

Impact:

Removed roles should be assigned to another user based on business needs.

Audit:

From Console:

1. Go to `IAM & Admin/IAM` by visiting: https://console.cloud.google.com/iam-admin/iam
2. Ensure no member has the roles `Cloud KMS Admin` and any of the `Cloud KMS CryptoKey Encrypter/Decrypter`, `Cloud KMS CryptoKey Encrypter`, `Cloud KMS CryptoKey Decrypter` assigned.

From Command Line:

1. List all users and role assignments:

```
gcloud projects get-iam-policy PROJECT_ID
```

2. Ensure that there are no common users found in the member section for roles `cloudkms.admin` and any one of `Cloud KMS CryptoKey Encrypter/Decrypter`, `Cloud KMS CryptoKey Encrypter`, `Cloud KMS CryptoKey Decrypter`

Remediation:

From Console:

1. Go to `IAM & Admin/IAM` using `https://console.cloud.google.com/iam-admin/iam`
2. For any member having `Cloud KMS Admin` and any of the `Cloud KMS CryptoKey Encrypter/Decrypter`, `Cloud KMS CryptoKey Encrypter`, `Cloud KMS CryptoKey Decrypter` roles granted/assigned, click the `Delete Bin` icon to remove the role from the member.

Note: Removing a role should be done based on the business requirement.

References:

1. https://cloud.google.com/kms/docs/separation-of-duties

Additional Information:

Users granted with Owner (roles/owner) and Editor (roles/editor) have privileges equivalent to `Cloud KMS Admin` and `Cloud KMS CryptoKey Encrypter/Decrypter`. To avoid misuse, Owner and Editor roles should be granted to a very limited group of users. Use of these primitive privileges should be minimal. These requirements are addressed in separate recommendations.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	4 Controlled Use of Administrative Privileges Controlled Use of Administrative Privileges			
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●
v7	16 Account Monitoring and Control Account Monitoring and Control			

1.12 Ensure API keys are not created for a project (Manual)

Profile Applicability:

- Level 2

Description:

Keys are insecure because they can be viewed publicly, such as from within a browser, or they can be accessed on a device where the key resides. It is recommended to use standard authentication flow instead.

Rationale:

Security risks involved in using API-Keys appear below:

- API keys are simple encrypted strings
- API keys do not identify the user or the application making the API request
- API keys are typically accessible to clients, making it easy to discover and steal an API key

To avoid the security risk in using API keys, it is recommended to use standard authentication flow instead.

Impact:

Deleting an API key will break dependent applications (if any).

Audit:

From Console:

1. Go to `APIs & Services\Credentials` using `https://console.cloud.google.com/apis/credentials`
2. In the section `API Keys`, no API key should be listed.

Remediation:

From Console:

1. Go to `APIs & Services\Credentials` using `https://console.cloud.google.com/apis/credentials`
2. In the section `API Keys`, to delete API Keys: Click the `Delete Bin Icon` in front of every `API Key Name`.

References:

1. https://cloud.google.com/docs/authentication/api-keys

Additional Information:

Google recommends using the standard authentication flow instead of using API keys. However, there are limited cases where API keys are more appropriate. For example, if there is a mobile application that needs to use the Google Cloud Translation API, but doesn't otherwise need a backend server, API keys are the simplest way to authenticate to that API.

If a business requires API keys to be used, then the API keys should be secured properly.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

1.13 Ensure API keys are restricted to use by only specified Hosts and Apps (Manual)

Profile Applicability:

- Level 1

Description:

Unrestricted keys are insecure because they can be viewed publicly, such as from within a browser, or they can be accessed on a device where the key resides. It is recommended to restrict API key usage to trusted hosts, HTTP referrers and apps.

Rationale:

Security risks involved in using API-Keys appear below:

- API keys are simple encrypted strings
- API keys do not identify the user or the application making the API request
- API keys are typically accessible to clients, making it easy to discover and steal an API key

In light of these potential risks, Google recommends using the standard authentication flow instead of API keys. However, there are limited cases where API keys are more appropriate. For example, if there is a mobile application that needs to use the Google Cloud Translation API, but doesn't otherwise need a backend server, API keys are the simplest way to authenticate to that API.

In order to reduce attack vectors, API-Keys can be restricted only to trusted hosts, HTTP referrers and applications.

Impact:

Setting `Application Restrictions` may break existing application functioning, if not done carefully.

Audit:

From Console:

1. Go to `APIs & Services\Credentials` using `https://console.cloud.google.com/apis/credentials`
2. In the section `API Keys`, Click the `API Key Name`. The API Key properties display on a new page.

3. For every API Key, ensure the section `Key restrictions` parameter `Application restrictions` is not set to `None`.

Or,
Ensure `Application restrictions` is set to `HTTP referrers` and the referrer is not set to wild-cards `(* or *.[TLD] or *.[TLD]/*) allowing access to any/wide HTTP referrer(s)`
Or,
Ensure `Application restrictions` is set to `IP addresses` and referrer is not set to `any host (0.0.0.0 or 0.0.0.0/0 or ::0)`

Remediation:

From Console:

1. Go to `APIs & Services\Credentials` using `https://console.cloud.google.com/apis/credentials`
2. In the section `API Keys`, Click the `API Key Name`. The API Key properties display on a new page.
3. In the `Key restrictions` section, set the application restrictions to any of `HTTP referrers, IP Adresses, Android Apps, iOs Apps`.
4. Click `Save`.
5. Repeat steps 2,3,4 for every unrestricted API key.
   Note: Do not set `HTTP referrers` to wild-cards (* or *.[TLD] or .*[TLD]/*) allowing access to any/wide HTTP referrer(s)
   Do not set `IP addresses` and referrer to `any host (0.0.0.0 or 0.0.0.0/0 or ::0)`

Default Value:

By default, `Application Restrictions` are set to `None`.

References:

1. https://cloud.google.com/docs/authentication/api-keys

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

1.14 Ensure API keys are restricted to only APIs that application needs access (Manual)

Profile Applicability:

- Level 1

Description:

API keys are insecure because they can be viewed publicly, such as from within a browser, or they can be accessed on a device where the key resides. It is recommended to restrict API keys to use (call) only APIs required by an application.

Rationale:

Security risks involved in using API-Keys are below:

- API keys are simple encrypted strings
- API keys do not identify the user or the application making the API request
- API keys are typically accessible to clients, making it easy to discover and steal an API key

In light of these potential risks, Google recommends using the standard authentication flow instead of API-Keys. However, there are limited cases where API keys are more appropriate. For example, if there is a mobile application that needs to use the Google Cloud Translation API, but doesn't otherwise need a backend server, API keys are the simplest way to authenticate to that API.

In order to reduce attack surfaces by providing `least privileges`, API-Keys can be restricted to use (call) only APIs required by an application.

Impact:

Setting `API restrictions` may break existing application functioning, if not done carefully.

Audit:

From Console:

1. Go to `APIs & Services\Credentials` using `https://console.cloud.google.com/apis/credentials`
2. In the section `API Keys`, Click the `API Key Name`. The API Key properties display on a new page.

3. For every API Key, ensure the section `Key restrictions` parameter `API restrictions` is not set to `None`.

Or,
Ensure `API restrictions` is not set to `Google Cloud APIs`
Note: `Google Cloud APIs` represents the API collection of all cloud services/APIs offered by Google cloud.

Remediation:

From Console:

1. Go to `APIs & Services\Credentials` using `https://console.cloud.google.com/apis/credentials`
2. In the section `API Keys`, Click the `API Key Name`. The API Key properties display on a new page.
3. In the `Key restrictions` section go to `API restrictions`.
4. Click the `Select API` drop-down to choose an API.
5. Click `Save`.
6. Repeat steps 2,3,4,5 for every unrestricted API key

Note: Do not set `API restrictions` to `Google Cloud APIs`, as this option allows access to all services offered by Google cloud.

Default Value:

By default, `API restrictions` are set to `None`.

References:

1. https://cloud.google.com/docs/authentication/api-keys
2. https://cloud.google.com/apis/docs/overview

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

1.15 Ensure API keys are rotated every 90 days (Manual)

Profile Applicability:

- Level 1

Description:

It is recommended to rotate API keys every 90 days.

Rationale:

Security risks involved in using API-Keys are listed below:

- API keys are simple encrypted strings
- API keys do not identify the user or the application making the API request
- API keys are typically accessible to clients, making it easy to discover and steal an API key

Because of these potential risks, Google recommends using the standard authentication flow instead of API Keys. However, there are limited cases where API keys are more appropriate. For example, if there is a mobile application that needs to use the Google Cloud Translation API, but doesn't otherwise need a backend server, API keys are the simplest way to authenticate to that API.

Once a key is stolen, it has no expiration, meaning it may be used indefinitely unless the project owner revokes or regenerates the key. Rotating API keys will reduce the window of opportunity for an access key that is associated with a compromised or terminated account to be used.

API keys should be rotated to ensure that data cannot be accessed with an old key that might have been lost, cracked, or stolen.

Impact:

`Regenerating Key` may break existing client connectivity as the client will try to connect with older API keys they have stored on devices.

Audit:

From Console:

1. Go to `APIs & Services\Credentials` using `https://console.cloud.google.com/apis/credentials`

2. In the section `API Keys`, for every key ensure the `creation date` is less than 90 days.

Remediation:

From Console:

1. Go to `APIs & Services\Credentials` using `https://console.cloud.google.com/apis/credentials`
2. In the section `API Keys`, Click the `API Key Name`. The API Key properties display on a new page.
3. Click `REGENERATE KEY` to rotate API key.
4. Click `Save`.
5. Repeat steps 2,3,4 for every API key that has not been rotated in the last 90 days.

Note: Do not set `HTTP referrers` to wild-cards (* or *.[TLD] or *.[TLD]/*) allowing access to any/wide HTTP referrer(s)
Do not set `IP addresses` and referrer to `any host (0.0.0.0 or 0.0.0.0/0 or ::0)`

References:

1. There is no option to automatically regenerate (rotate) API keys periodically.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

2 Logging and Monitoring

This section covers recommendations addressing Logging and Monitoring on Google Cloud Platform.

2.1 Ensure that Cloud Audit Logging is configured properly across all services and all users from a project (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended that Cloud Audit Logging is configured to track all admin activities and read, write access to user data.

Rationale:

Cloud Audit Logging maintains two audit logs for each project, folder, and organization: Admin Activity and Data Access.

1. Admin Activity logs contain log entries for API calls or other administrative actions that modify the configuration or metadata of resources. Admin Activity audit logs are enabled for all services and cannot be configured.
2. Data Access audit logs record API calls that create, modify, or read user-provided data. These are disabled by default and should be enabled.

   There are three kinds of Data Access audit log information:

   - Admin read: Records operations that read metadata or configuration information. Admin Activity audit logs record writes of metadata and configuration information that cannot be disabled.
   - Data read: Records operations that read user-provided data.
   - Data write: Records operations that write user-provided data.

It is recommended to have an effective default audit config configured in such a way that:

1. logtype is set to DATA_READ (to log user activity tracking) and DATA_WRITES (to log changes/tampering to user data).
2. audit config is enabled for all the services supported by the Data Access audit logs feature.
3. Logs should be captured for all users, i.e., there are no exempted users in any of the audit config sections. This will ensure overriding the audit config will not contradict the requirement.

Impact:

There is no charge for Admin Activity audit logs. Enabling the Data Access audit logs might result in your project being charged for the additional logs usage.

Audit:

From Console:

1. Go to `Audit Logs` by visiting https://console.cloud.google.com/iam-admin/audit.
2. Ensure that Admin Read, Data Write, and Data Read are enabled for all Google Cloud services and that no exemptions are allowed.

From Command Line:

1. List the Identity and Access Management (IAM) policies for the project, folder, or organization:

```
gcloud organizations get-iam-policy ORGANIZATION_ID
gcloud resource-manager folders get-iam-policy FOLDER_ID
gcloud projects get-iam-policy PROJECT_ID
```

2. Policy should have a default auditConfigs section which has the logtype set to DATA_WRITES and DATA_READ for all services. Note that projects inherit settings from folders, which in turn inherit settings from the organization. When called, projects get-iam-policy, the result shows only the policies set in the project, not the policies inherited from the parent folder or organization. Nevertheless, if the parent folder has Cloud Audit Logging enabled, the project does as well.

Sample output for default audit configs may look like this:

```
        auditConfigs:
        - auditLogConfigs:
        - logType: ADMIN_READ
        - logType: DATA_WRITE
        - logType: DATA_READ
         service: allServices
```

3. Any of the auditConfigs sections should not have parameter "exemptedMembers:" set, which will ensure that Logging is enabled for all users and no user is exempted.

Remediation:

From Console:

1. Go to `Audit Logs` by visiting https://console.cloud.google.com/iam-admin/audit.

2. Follow the steps at https://cloud.google.com/logging/docs/audit/configure-data-access to enable audit logs for all Google Cloud services. Ensure that no exemptions are allowed.

From Command Line:

1. To read the project's IAM policy and store it in a file run a command:

```
gcloud projects get-iam-policy PROJECT_ID > /tmp/project_policy.yaml
```

Alternatively, the policy can be set at the organization or folder level. If setting the policy at the organization level, it is not necessary to also set it for each folder or project.

```
gcloud organizations get-iam-policy ORGANIZATION_ID > /tmp/org_policy.yaml
gcloud resource-manager folders get-iam-policy FOLDER_ID >
/tmp/folder_policy.yaml
```

2. Edit policy in /tmp/policy.yaml, adding or changing only the audit logs configuration to:

```
auditConfigs:
- auditLogConfigs:
  - logType: DATA_WRITE
  - logType: DATA_READ
  service: allServices
```

Note: `exemptedMembers:` is not set as audit logging should be enabled for all the users

3. To write new IAM policy run command:

```
gcloud organizations set-iam-policy ORGANIZATION_ID /tmp/org_policy.yaml
gcloud resource-manager folders set-iam-policy FOLDER_ID
/tmp/folder_policy.yaml
gcloud projects set-iam-policy PROJECT_ID /tmp/project_policy.yaml
```

If the preceding command reports a conflict with another change, then repeat these steps, starting with the first step.

Default Value:

Admin Activity logs are always enabled. They cannot be disabled. Data Access audit logs are disabled by default because they can be quite large.

References:

1. https://cloud.google.com/logging/docs/audit/
2. https://cloud.google.com/logging/docs/audit/configure-data-access

Additional Information:

- Log type `DATA_READ` is equally important to that of `DATA_WRITE` to track detailed user activities.
- BigQuery Data Access logs are handled differently from other data access logs. BigQuery logs are enabled by default and cannot be disabled. They do not count against logs allotment and cannot result in extra logs charges.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.7 Regularly Review Logs On a regular basis, review logs to identify anomalies or abnormal events.		●	●

2.2 Ensure that sinks are configured for all log entries (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to create a sink that will export copies of all the log entries. This can help aggregate logs from multiple projects and export them to a Security Information and Event Management (SIEM).

Rationale:

Log entries are held in Cloud Logging. To aggregate logs, export them to a SIEM. To keep them longer, it is recommended to set up a log sink. Exporting involves writing a filter that selects the log entries to export, and choosing a destination in Cloud Storage, BigQuery, or Cloud Pub/Sub. The filter and destination are held in an object called a sink. To ensure all log entries are exported to sinks, ensure that there is no filter configured for a sink. Sinks can be created in projects, organizations, folders, and billing accounts.

Impact:

There are no costs or limitations in Cloud Logging for exporting logs, but the export destinations charge for storing or transmitting the log data.

Audit:

From Console:

1. Go to `Logging/Exports` by visiting https://console.cloud.google.com/logs/exports.
2. For every sink, click the 3-dot button for Menu options and select `View Filter`.
3. Ensure there is at least one sink with an `empty` sink filter.
4. Additionally, ensure that the resource configured as `Destination` exists.

From Command Line:

1. Ensure that a sink with an `empty filter` exists. List the sinks for the project, folder or organization. If sinks are configured at a folder or organization level, they do not need to be configured for each project:

```
gcloud logging sinks list --folder=FOLDER_ID | --organization=ORGANIZATION_ID
| --project=PROJECT_ID
```

The output should list at least one sink with an `empty filter`.

2. Additionally, ensure that the resource configured as `Destination` exists.

See https://cloud.google.com/sdk/gcloud/reference/beta/logging/sinks/list for more information.

Remediation:

From Console:

1. Go to `Logging/Logs` by visiting https://console.cloud.google.com/logs/viewer.
2. Click the down arrow symbol on `Filter Bar` at the rightmost corner and select `Convert to Advanced Filter`.
3. This step converts `Filter Bar` to `Advanced Filter Bar`.
4. Clear any text from the `Advanced Filter` field. This ensures that the `log-filter` is set to empty and captures all the logs.
5. Click `Submit Filter` and the result should display all logs.
6. Click `Create Sink`, which opens a menu on the right.
7. Fill out the fields and click `Create Sink`.

For more information, see https://cloud.google.com/logging/docs/export/configure_export_v2#dest-create.

From Command Line:

To create a sink to export all log entries in a Google Cloud Storage bucket:

```
gcloud logging sinks create <sink-name>
storage.googleapis.com/DESTINATION_BUCKET_NAME
```

Sinks can be created for a folder or organization, which will include all projects.

```
gcloud logging sinks create <sink-name>
storage.googleapis.com/DESTINATION_BUCKET_NAME --include-children --
folder=FOLDER_ID | --organization=ORGANIZATION_ID
```

Note:

1. A sink created by the command-line above will export logs in storage buckets. However, sinks can be configured to export logs into BigQuery, or Cloud Pub/Sub, or `Custom Destination`.
2. While creating a sink, the sink option `--log-filter` is not used to ensure the sink exports all log entries.

3. A sink can be created at a folder or organization level that collects the logs of all the projects underneath bypassing the option `--include-children` in the gcloud command.

Default Value:

By default, there are no sinks configured.

References:

1. https://cloud.google.com/logging/docs/reference/tools/gcloud-logging
2. https://cloud.google.com/logging/quotas
3. https://cloud.google.com/logging/docs/export/
4. https://cloud.google.com/logging/docs/export/using_exported_logs
5. https://cloud.google.com/logging/docs/export/configure_export_v2
6. https://cloud.google.com/logging/docs/export/aggregated_exports
7. https://cloud.google.com/sdk/gcloud/reference/beta/logging/sinks/list

Additional Information:

For Command-Line Audit and Remediation, the sink destination of type `Cloud Storage Bucket` is considered. However, the destination could be configured to `Cloud Storage Bucket` or `BigQuery` or `Cloud Pub\Sub` or `Custom Destination`. Command Line Interface commands would change accordingly.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.4 Ensure adequate storage for logs Ensure that all systems that store logs have adequate storage space for the logs generated.		●	●

2.3 Ensure that retention policies on log buckets are configured using Bucket Lock (Automated)

Profile Applicability:

- Level 1

Description:

Enabling retention policies on log buckets will protect logs stored in cloud storage buckets from being overwritten or accidentally deleted. It is recommended to set up retention policies and configure Bucket Lock on all storage buckets that are used as log sinks.

Rationale:

Logs can be exported by creating one or more sinks that include a log filter and a destination. As Cloud Logging receives new log entries, they are compared against each sink. If a log entry matches a sink's filter, then a copy of the log entry is written to the destination.

Sinks can be configured to export logs in storage buckets. It is recommended to configure a data retention policy for these cloud storage buckets and to lock the data retention policy; thus permanently preventing the policy from being reduced or removed. This way, if the system is ever compromised by an attacker or a malicious insider who wants to cover their tracks, the activity logs are definitely preserved for forensics and security investigations.

Impact:

Locking a bucket is an irreversible action. Once you lock a bucket, you cannot remove the retention policy from the bucket or decrease the retention period for the policy.

Audit:

From Console:

1. Open the Cloud Storage browser in the Google Cloud Console by visiting https://console.cloud.google.com/storage/browser.
2. In the Column display options menu, make sure `Retention policy` is checked.
3. In the list of buckets, the retention period of each bucket is found in the `Retention policy` column. If the retention policy is locked, an image of a lock appears directly to the left of the retention period.

From Command Line:

1. To list all sinks destined to storage buckets:

```
gcloud logging sinks list --folder=FOLDER_ID | --organization=ORGANIZATION_ID
| --project=PROJECT_ID
```

2. For every storage bucket listed above, verify that retention policies and Bucket Lock are enabled:

```
gsutil retention get gs://BUCKET_NAME
```

For more information, see https://cloud.google.com/storage/docs/using-bucket-lock#view-policy.

Remediation:

From Console:

1. If sinks are **not** configured, first follow the instructions in the recommendation: `Ensure that sinks are configured for all Log entries`.
2. For each storage bucket configured as a sink, go to the Cloud Storage browser at `https://console.cloud.google.com/storage/browser/<BUCKET_NAME>`.
3. Select the Bucket Lock tab near the top of the page.
4. In the Retention policy entry, click the Add Duration link. The `Set a retention policy` dialog box appears.
5. Enter the desired length of time for the retention period and click `Save policy`.
6. Set the `Lock status` for this retention policy to `Locked`.

From Command Line:

1. To list all sinks destined to storage buckets:

```
gcloud logging sinks list --folder=FOLDER_ID | --organization=ORGANIZATION_ID
| --project=PROJECT_ID
```

2. For each storage bucket listed above, set a retention policy and lock it:

```
gsutil retention set [TIME_DURATION] gs://[BUCKET_NAME]
gsutil retention lock gs://[BUCKET_NAME]
```

For more information, visit https://cloud.google.com/storage/docs/using-bucket-lock#set-policy.

Default Value:

By default, storage buckets used as log sinks do not have retention policies and Bucket Lock configured.

References:

1. https://cloud.google.com/storage/docs/bucket-lock
2. https://cloud.google.com/storage/docs/using-bucket-lock

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6 Maintenance, Monitoring and Analysis of Audit Logs Maintenance, Monitoring and Analysis of Audit Logs			

2.4 Ensure log metric filter and alerts exist for project ownership assignments/changes (Automated)

Profile Applicability:

- Level 1

Description:

In order to prevent unnecessary project ownership assignments to users/service-accounts and further misuses of projects and resources, all `roles/Owner` assignments should be monitored.

Members (users/Service-Accounts) with a role assignment to primitive role `roles/Owner` are project owners.

The project owner has all the privileges on the project the role belongs to. These are summarized below:

```
- All viewer permissions on all GCP Services within the project



- Permissions for actions that modify the state of all GCP services within
the project



- Manage roles and permissions for a project and all resources within the
project



- Set up billing for a project
```

Granting the owner role to a member (user/Service-Account) will allow that member to modify the Identity and Access Management (IAM) policy. Therefore, grant the owner role only if the member has a legitimate purpose to manage the IAM policy. This is because the project IAM policy contains sensitive access control data. Having a minimal set of users allowed to manage IAM policy will simplify any auditing that may be necessary.

Rationale:

Project ownership has the highest level of privileges on a project. To avoid misuse of project resources, the project ownership assignment/change actions mentioned above should be monitored and alerted to concerned recipients.

```
- Sending project ownership invites


- Acceptance/Rejection of project ownership invite by user


- Adding `role\Owner` to a user/service-account


- Removing a user/Service account from `role\Owner`
```

Impact:

Enabling of logging may result in your project being charged for the additional logs usage.

Audit:

From Console:
Ensure that the prescribed log metric is present:

1. Go to `Logging/Log-based Metrics` by visiting https://console.cloud.google.com/logs/metrics.
2. In the `User-defined Metrics` section, ensure that at least one metric `<Log_Metric_Name>` is present with filter text:

```
(protoPayload.serviceName="cloudresourcemanager.googleapis.com")
AND (ProjectOwnership OR projectOwnerInvitee)
OR (protoPayload.serviceData.policyDelta.bindingDeltas.action="REMOVE"
AND protoPayload.serviceData.policyDelta.bindingDeltas.role="roles/owner")
OR (protoPayload.serviceData.policyDelta.bindingDeltas.action="ADD"
AND protoPayload.serviceData.policyDelta.bindingDeltas.role="roles/owner")
```

Ensure that the prescribed Alerting Policy is present:

3. Go to `Alerting` by visiting https://console.cloud.google.com/monitoring/alerting.
4. Under the `Policies` section, ensure that at least one alert policy exists for the log metric above. Clicking on the policy should show that it is configured with a condition. For example, `Violates when: Any logging.googleapis.com/user/<Log Metric Name> stream is above a threshold of zero(0) for greater than zero(0) seconds` means that the alert will trigger for any new owner change. Verify that the chosen alerting thresholds make sense for your organization.
5. Ensure that the appropriate notifications channels have been set up.

From Command Line:
Ensure that the prescribed log metric is present:

1. List the log metrics:

```
gcloud beta logging metrics list --format json
```

2. Ensure that the output contains at least one metric with filter set to:

```
(protoPayload.serviceName="cloudresourcemanager.googleapis.com")
AND (ProjectOwnership OR projectOwnerInvitee)
OR (protoPayload.serviceData.policyDelta.bindingDeltas.action="REMOVE"
AND protoPayload.serviceData.policyDelta.bindingDeltas.role="roles/owner")
OR (protoPayload.serviceData.policyDelta.bindingDeltas.action="ADD"
AND protoPayload.serviceData.policyDelta.bindingDeltas.role="roles/owner")
```

3. Note the value of the property `metricDescriptor.type` for the identified metric, in the format `logging.googleapis.com/user/<Log Metric Name>`.

Ensure that the prescribed alerting policy is present:

4. List the alerting policies:

```
gcloud alpha monitoring policies list --format json
```

5. Ensure that the output contains an least one alert policy where:

- `conditions.conditionThreshold.filter` is set to `metric.type=\"logging.googleapis.com/user/<Log Metric Name>\"`
- AND `enabled` is set to `true`

Remediation:

From Console:
Create the prescribed log metric:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics and click "CREATE METRIC".
2. Click the down arrow symbol on the `Filter Bar` at the rightmost corner and select `Convert to Advanced Filter`.
3. Clear any text and add:

```
(protoPayload.serviceName="cloudresourcemanager.googleapis.com")
AND (ProjectOwnership OR projectOwnerInvitee)
OR (protoPayload.serviceData.policyDelta.bindingDeltas.action="REMOVE"
AND protoPayload.serviceData.policyDelta.bindingDeltas.role="roles/owner")
OR (protoPayload.serviceData.policyDelta.bindingDeltas.action="ADD"
AND protoPayload.serviceData.policyDelta.bindingDeltas.role="roles/owner")
```

4. Click `Submit Filter`. The logs display based on the filter text entered by the user.
5. In the `Metric Editor` menu on the right, fill out the name field. Set `Units` to `1` (default) and the `Type` to `Counter`. This ensures that the log metric counts the number of log entries matching the advanced logs query.
6. Click `Create Metric`.

Create the display prescribed Alert Policy:

1. Identify the newly created metric under the section `User-defined Metrics` at https://console.cloud.google.com/logs/metrics.
2. Click the 3-dot icon in the rightmost column for the desired metric and select `Create alert from Metric`. A new page opens.
3. Fill out the alert policy configuration and click `Save`. Choose the alerting threshold and configuration that makes sense for the user's organization. For example, a threshold of zero(0) for the most recent value will ensure that a notification is triggered for every owner change in the project:

```
Set `Aggregator` to `Count`

Set `Configuration`:

- Condition: above

- Threshold: 0

- For: most recent value
```

4. Configure the desired notifications channels in the section `Notifications`.
5. Name the policy and click `Save`.

From Command Line:

Create a prescribed Log Metric:

- Use the command: gcloud beta logging metrics create
- Reference for Command Usage: https://cloud.google.com/sdk/gcloud/reference/beta/logging/metrics/create

Create prescribed Alert Policy

- Use the command: gcloud alpha monitoring policies create
- Reference for Command Usage: https://cloud.google.com/sdk/gcloud/reference/alpha/monitoring/policies/create

References:

1. https://cloud.google.com/logging/docs/logs-based-metrics/
2. https://cloud.google.com/monitoring/custom-metrics/
3. https://cloud.google.com/monitoring/alerts/
4. https://cloud.google.com/logging/docs/reference/tools/gcloud-logging

Additional Information:

1. Project ownership assignments for a user cannot be done using the gcloud utility as assigning project ownership to a user requires sending, and the user accepting, an invitation.
2. Project Ownership assignment to a service account does not send any invites. SetIAMPolicy to `role/owner`is directly performed on service accounts.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

2.5 Ensure that the log metric filter and alerts exist for Audit Configuration changes (Automated)

Profile Applicability:

- Level 1

Description:

Google Cloud Platform (GCP) services write audit log entries to the Admin Activity and Data Access logs to help answer the questions of, "who did what, where, and when?" within GCP projects.

Cloud audit logging records information includes the identity of the API caller, the time of the API call, the source IP address of the API caller, the request parameters, and the response elements returned by GCP services. Cloud audit logging provides a history of GCP API calls for an account, including API calls made via the console, SDKs, command-line tools, and other GCP services.

Rationale:

Admin activity and data access logs produced by cloud audit logging enable security analysis, resource change tracking, and compliance auditing.

Configuring the metric filter and alerts for audit configuration changes ensures the recommended state of audit configuration is maintained so that all activities in the project are audit-able at any point in time.

Impact:

Enabling of logging may result in your project being charged for the additional logs usage.

Audit:

From Console:
Ensure the prescribed log metric is present:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics.
2. In the `User-defined Metrics` section, ensure that at least one metric `<Log_Metric_Name>` is present with the filter text:

```
protoPayload.methodName="SetIamPolicy" AND
protoPayload.serviceData.policyDelta.auditConfigDeltas:*
```

Ensure that the prescribed alerting policy is present:

3. Go to `Alerting` by visiting https://console.cloud.google.com/monitoring/alerting.
4. Under the `Policies` section, ensure that at least one alert policy exists for the log metric above. Clicking on the policy should show that it is configured with a condition. For example, `Violates when: Any logging.googleapis.com/user/<Log Metric Name> stream` `is above a threshold of 0 for greater than zero(0) seconds`, means that the alert will trigger for any new owner change. Verify that the chosen alerting thresholds make sense for the user's organization.
5. Ensure that appropriate notifications channels have been set up.

From Command Line:
Ensure that the prescribed log metric is present:

1. List the log metrics:

```
gcloud beta logging metrics list --format json
```

2. Ensure that the output contains at least one metric with the filter set to:

```
protoPayload.methodName="SetIamPolicy" AND
protoPayload.serviceData.policyDelta.auditConfigDeltas:*
```

3. Note the value of the property `metricDescriptor.type` for the identified metric, in the format `logging.googleapis.com/user/<Log Metric Name>`.

Ensure that the prescribed alerting policy is present:

4. List the alerting policies:

```
gcloud alpha monitoring policies list --format json
```

5. Ensure that the output contains at least one alert policy where:

- `conditions.conditionThreshold.filter` is set to `metric.type=\"logging.googleapis.com/user/<Log Metric Name>\"`
- AND `enabled` is set to `true`

Remediation:

From Console:
Create the prescribed log metric:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics and click "CREATE METRIC".
2. Click the down arrow symbol on the `Filter Bar` at the rightmost corner and select `Convert to Advanced Filter`.
3. Clear any text and add:

```
protoPayload.methodName="SetIamPolicy" AND
protoPayload.serviceData.policyDelta.auditConfigDeltas:*
```

4. Click `Submit Filter`. Display logs appear based on the filter text entered by the user.
5. In the `Metric Editor` menu on the right, fill out the name field. Set `Units` to `1` (default) and `Type` to `Counter`. This will ensure that the log metric counts the number of log entries matching the user's advanced logs query.
6. Click `Create Metric`.

Create a prescribed Alert Policy:

1. Identify the new metric the user just created, under the section `User-defined Metrics` at https://console.cloud.google.com/logs/metrics.
2. Click the 3-dot icon in the rightmost column for the new metric and select `Create alert from Metric`. A new page opens.
3. Fill out the alert policy configuration and click `Save`. Choose the alerting threshold and configuration that makes sense for the organization. For example, a threshold of zero(0) for the most recent value will ensure that a notification is triggered for every owner change in the project:

```
Set `Aggregator` to `Count`

Set `Configuration`:

- Condition: above

- Threshold: 0

- For: most recent value
```

4. Configure the desired notifications channels in the section `Notifications`.
5. Name the policy and click `Save`.

From Command Line:

Create a prescribed Log Metric:

- Use the command: gcloud beta logging metrics create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/beta/logging/metrics/create

Create prescribed Alert Policy

- Use the command: gcloud alpha monitoring policies create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/alpha/monitoring/policies/create

References:

1. https://cloud.google.com/logging/docs/logs-based-metrics/
2. https://cloud.google.com/monitoring/custom-metrics/
3. https://cloud.google.com/monitoring/alerts/
4. https://cloud.google.com/logging/docs/reference/tools/gcloud-logging
5. https://cloud.google.com/logging/docs/audit/configure-data-access#getiampolicy-setiampolicy

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

2.6 Ensure that the log metric filter and alerts exist for Custom Role changes (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended that a metric filter and alarm be established for changes to Identity and Access Management (IAM) role creation, deletion and updating activities.

Rationale:

Google Cloud IAM provides predefined roles that give granular access to specific Google Cloud Platform resources and prevent unwanted access to other resources. However, to cater to organization-specific needs, Cloud IAM also provides the ability to create custom roles. Project owners and administrators with the Organization Role Administrator role or the IAM Role Administrator role can create custom roles. Monitoring role creation, deletion and updating activities will help in identifying any over-privileged role at early stages.

Impact:

Enabling of logging may result in your project being charged for the additional logs usage.

Audit:

From Console:
Ensure that the prescribed log metric is present:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics.
2. In the `User-defined Metrics` section, ensure that at least one metric `<Log_Metric_Name>` is present with filter text:

```
resource.type="iam_role"
AND protoPayload.methodName =  "google.iam.admin.v1.CreateRole"
OR protoPayload.methodName="google.iam.admin.v1.DeleteRole"
OR protoPayload.methodName="google.iam.admin.v1.UpdateRole"
```

Ensure that the prescribed alerting policy is present:

3. Go to `Alerting` by visiting https://console.cloud.google.com/monitoring/alerting.

4. Under the `Policies` section, ensure that at least one alert policy exists for the log metric above. Clicking on the policy should show that it is configured with a condition. For example, `Violates when: Any logging.googleapis.com/user/<Log Metric Name> stream` `is above a threshold of zero(0) for greater than zero(0) seconds` means that the alert will trigger for any new owner change. Verify that the chosen alerting thresholds make sense for the user's organization.
5. Ensure that the appropriate notifications channels have been set up.

From Command Line:
Ensure that the prescribed log metric is present:

1. List the log metrics:

```
gcloud beta logging metrics list --format json
```

2. Ensure that the output contains at least one metric with the filter set to:

```
resource.type="iam_role" AND protoPayload.methodName =
"google.iam.admin.v1.CreateRole" OR
 protoPayload.methodName="google.iam.admin.v1.DeleteRole" OR
 protoPayload.methodName="google.iam.admin.v1.UpdateRole"
```

3. Note the value of the property `metricDescriptor.type` for the identified metric, in the format `logging.googleapis.com/user/<Log Metric Name>`.

Ensure that the prescribed alerting policy is present:

4. List the alerting policies:

```
gcloud alpha monitoring policies list --format json
```

5. Ensure that the output contains an least one alert policy where:

- `conditions.conditionThreshold.filter` is set to `metric.type=\"logging.googleapis.com/user/<Log Metric Name>\"`
- AND `enabled` is set to `true`.

Remediation:

From Console:
Create the prescribed log metric:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics and click "CREATE METRIC".
2. Click the down arrow symbol on the `Filter Bar` at the rightmost corner and select `Convert to Advanced Filter`.

3. Clear any text and add:

```
resource.type="iam_role"
AND protoPayload.methodName =  "google.iam.admin.v1.CreateRole"
OR protoPayload.methodName="google.iam.admin.v1.DeleteRole"
OR protoPayload.methodName="google.iam.admin.v1.UpdateRole"
```

4. Click `Submit Filter`. Display logs appear based on the filter text entered by the user.
5. In the `Metric Editor` menu on the right, fill out the name field. Set `Units` to `1` (default) and `Type` to `Counter`. This ensures that the log metric counts the number of log entries matching the advanced logs query.
6. Click `Create Metric`.

Create a prescribed Alert Policy:

1. Identify the new metric that was just created under the section `User-defined Metrics` at https://console.cloud.google.com/logs/metrics.
2. Click the 3-dot icon in the rightmost column for the metric and select `Create alert from Metric`. A new page displays.
3. Fill out the alert policy configuration and click `Save`. Choose the alerting threshold and configuration that makes sense for the user's organization. For example, a threshold of zero(0) for the most recent value ensures that a notification is triggered for every owner change in the project:

```
Set `Aggregator` to `Count`

Set `Configuration`:

- Condition: above

- Threshold: 0

- For: most recent value
```

4. Configure the desired notification channels in the section `Notifications`.
5. Name the policy and click `Save`.

From Command Line:

Create the prescribed Log Metric:

- Use the command: gcloud beta logging metrics create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/beta/logging/metrics/create

Create the prescribed Alert Policy:

- Use the command: gcloud alpha monitoring policies create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/alpha/monitoring/policies/create

References:

1. https://cloud.google.com/logging/docs/logs-based-metrics/
2. https://cloud.google.com/monitoring/custom-metrics/
3. https://cloud.google.com/monitoring/alerts/
4. https://cloud.google.com/logging/docs/reference/tools/gcloud-logging
5. https://cloud.google.com/iam/docs/understanding-custom-roles

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

2.7 Ensure that the log metric filter and alerts exist for VPC Network Firewall rule changes (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended that a metric filter and alarm be established for Virtual Private Cloud (VPC) Network Firewall rule changes.

Rationale:

Monitoring for Create or Update Firewall rule events gives insight to network access changes and may reduce the time it takes to detect suspicious activity.

Impact:

Enabling of logging may result in your project being charged for the additional logs usage.

Audit:

From Console:
Ensure that the prescribed log metric is present:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics.
2. In the `User-defined Metrics` section, ensure at least one metric `<Log_Metric_Name>` is present with this filter text:

```
resource.type="gce_firewall_rule"
AND protoPayload.methodName="v1.compute.firewalls.patch"
OR protoPayload.methodName="v1.compute.firewalls.insert"
```

Ensure that the prescribed alerting policy is present:

3. Go to `Alerting` by visiting https://console.cloud.google.com/monitoring/alerting.
4. Under the `Policies` section, ensure that at least one alert policy exists for the log metric above. Clicking on the policy should show that it is configured with a condition. For example, `Violates when: Any logging.googleapis.com/user/<Log Metric Name> stream` is `above a threshold of zero(0) for greater than zero(0) seconds` means that the alert will trigger for any new owner change. Verify that the chosen alerting thresholds make sense for the user's organization.
5. Ensure that appropriate notification channels have been set up.

From Command Line:
Ensure that the prescribed log metric is present:

1. List the log metrics:

```
gcloud beta logging metrics list --format json
```

2. Ensure that the output contains at least one metric with the filter set to:

```
resource.type="gce_firewall_rule"
AND protoPayload.methodName="v1.compute.firewalls.patch"
OR protoPayload.methodName="v1.compute.firewalls.insert"
```

3. Note the value of the property `metricDescriptor.type` for the identified metric, in the format `logging.googleapis.com/user/<Log Metric Name>`.

Ensure that the prescribed alerting policy is present:

4. List the alerting policies:

```
gcloud alpha monitoring policies list --format json
```

5. Ensure that the output contains an least one alert policy where:

- `conditions.conditionThreshold.filter` is set to `metric.type=\"logging.googleapis.com/user/<Log Metric Name>\"`
- AND `enabled` is set to `true`

Remediation:

From Console:
Create the prescribed log metric:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics and click "CREATE METRIC".
2. Click the down arrow symbol on the `Filter Bar` at the rightmost corner and select `Convert to Advanced Filter`.
3. Clear any text and add:

```
resource.type="gce_firewall_rule"
AND protoPayload.methodName="v1.compute.firewalls.patch"
OR protoPayload.methodName="v1.compute.firewalls.insert"
```

4. Click `Submit Filter`. Display logs appear based on the filter text entered by the user.
5. In the `Metric Editor` menu on the right, fill out the name field. Set `Units` to `1` (default) and `Type` to `Counter`. This ensures that the log metric counts the number of log entries matching the advanced logs query.
6. Click `Create Metric`.

Create the prescribed Alert Policy:

1. Identify the newly created metric under the section `User-defined Metrics` at https://console.cloud.google.com/logs/metrics.
2. Click the 3-dot icon in the rightmost column for the new metric and select `Create alert from Metric`. A new page displays.
3. Fill out the alert policy configuration and click `Save`. Choose the alerting threshold and configuration that makes sense for the user's organization. For example, a threshold of zero(0) for the most recent value ensures that a notification is triggered for every owner change in the project:

```
Set `Aggregator` to `Count`

Set `Configuration`:

- Condition: above

- Threshold: 0

- For: most recent value
```

4. Configure the desired notifications channels in the section `Notifications`.
5. Name the policy and click `Save`.

From Command Line:

Create the prescribed Log Metric

- Use the command: gcloud beta logging metrics create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/beta/logging/metrics/create

Create the prescribed alert policy:

- Use the command: gcloud alpha monitoring policies create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/alpha/monitoring/policies/create

References:

1. https://cloud.google.com/logging/docs/logs-based-metrics/
2. https://cloud.google.com/monitoring/custom-metrics/
3. https://cloud.google.com/monitoring/alerts/
4. https://cloud.google.com/logging/docs/reference/tools/gcloud-logging
5. https://cloud.google.com/vpc/docs/firewalls

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

2.8 Ensure that the log metric filter and alerts exist for VPC network route changes (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended that a metric filter and alarm be established for Virtual Private Cloud (VPC) network route changes.

Rationale:

Google Cloud Platform (GCP) routes define the paths network traffic takes from a VM instance to another destination. The other destination can be inside the organization VPC network (such as another VM) or outside of it. Every route consists of a destination and a next hop. Traffic whose destination IP is within the destination range is sent to the next hop for delivery.

Monitoring changes to route tables will help ensure that all VPC traffic flows through an expected path.

Impact:

Enabling of logging may result in your project being charged for the additional logs usage.

Audit:

From Console:
Ensure that the prescribed Log metric is present:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics.
2. In the `User-defined Metrics` section, ensure that at least one metric `<Log_Metric_Name>` is present with the filter text:

```
resource.type="gce_route"
AND protoPayload.methodName="beta.compute.routes.patch"
OR protoPayload.methodName="beta.compute.routes.insert"
```

Ensure the prescribed alerting policy is present:

3. Go to `Alerting` by visiting: https://console.cloud.google.com/monitoring/alerting.
4. Under the `Policies` section, ensure that at least one alert policy exists for the log metric above. Clicking on the policy should show that it is configured with a condition. For example, `Violates when: Any logging.googleapis.com/user/<Log Metric Name> stream` `is above a threshold of 0 for greater than zero(0) seconds` means that the alert will trigger for any new owner change. Verify that the chosen alert thresholds make sense for the user's organization.
5. Ensure that the appropriate notification channels have been set up.

From Command Line:
Ensure the prescribed log metric is present:

1. List the log metrics:

```
gcloud beta logging metrics list --format json
```

2. Ensure that the output contains at least one metric with the filter set to:

```
resource.type="gce_route"
AND protoPayload.methodName="beta.compute.routes.patch"
OR protoPayload.methodName="beta.compute.routes.insert"
```

3. Note the value of the property `metricDescriptor.type` for the identified metric, in the format `logging.googleapis.com/user/<Log Metric Name>`.

Ensure that the prescribed alerting policy is present:

4. List the alerting policies:

```
gcloud alpha monitoring policies list --format json
```

5. Ensure that the output contains an least one alert policy where:

- `conditions.conditionThreshold.filter` is set to `metric.type=\"logging.googleapis.com/user/<Log Metric Name>\"`
- AND `enabled` is set to `true`

Remediation:

From Console:
Create the prescribed Log Metric:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics and click "CREATE METRIC".
2. Click the down arrow symbol on the `Filter Bar` at the rightmost corner and select `Convert to Advanced Filter`

3. Clear any text and add:

```
resource.type="gce_route"
AND protoPayload.methodName="beta.compute.routes.patch"
OR protoPayload.methodName="beta.compute.routes.insert"
```

4. Click `Submit Filter`. Display logs appear based on the filter text entered by the user.
5. In the `Metric Editor` menu on the right, fill out the name field. Set `Units` to `1` (default) and `Type` to `Counter`. This ensures that the log metric counts the number of log entries matching the user's advanced logs query.
6. Click `Create Metric`.

Create the prescribed alert policy:

1. Identify the newly created metric under the section `User-defined Metrics` at https://console.cloud.google.com/logs/metrics.
2. Click the 3-dot icon in the rightmost column for the new metric and select `Create alert from Metric`. A new page displays.
3. Fill out the alert policy configuration and click `Save`. Choose the alerting threshold and configuration that makes sense for the user's organization. For example, a threshold of zero(0) for the most recent value ensures that a notification is triggered for every owner change in the project:

```
Set `Aggregator` to `Count`

Set `Configuration`:

- Condition: above

- Threshold: 0

- For: most recent value
```

4. Configure the desired notification channels in the section `Notifications`.
5. Name the policy and click `Save`.

From Command Line:

Create the prescribed Log Metric:

- Use the command: gcloud beta logging metrics create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/beta/logging/metrics/create

Create the prescribed the alert policy:

- Use the command: gcloud alpha monitoring policies create

- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/alpha/monitoring/policies/create

References:

1. https://cloud.google.com/logging/docs/logs-based-metrics/
2. https://cloud.google.com/monitoring/custom-metrics/
3. https://cloud.google.com/monitoring/alerts/
4. https://cloud.google.com/logging/docs/reference/tools/gcloud-logging
5. https://cloud.google.com/storage/docs/access-control/iam

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

2.9 Ensure that the log metric filter and alerts exist for VPC network changes (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended that a metric filter and alarm be established for Virtual Private Cloud (VPC) network changes.

Rationale:

It is possible to have more than one VPC within a project. In addition, it is also possible to create a peer connection between two VPCs enabling network traffic to route between VPCs.

Monitoring changes to a VPC will help ensure VPC traffic flow is not getting impacted.

Impact:

Enabling of logging may result in your project being charged for the additional logs usage.

Audit:

From Console:
Ensure the prescribed log metric is present:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics.
2. In the `User-defined Metrics` section, ensure at least one metric `<Log_Metric_Name>` is present with filter text:

```
resource.type=gce_network
AND protoPayload.methodName="beta.compute.networks.insert"
OR protoPayload.methodName="beta.compute.networks.patch"
OR protoPayload.methodName="v1.compute.networks.delete"
OR protoPayload.methodName="v1.compute.networks.removePeering"
OR protoPayload.methodName="v1.compute.networks.addPeering"
```

Ensure the prescribed alerting policy is present:

3. Go to `Alerting` by visiting https://console.cloud.google.com/monitoring/alerting.

4. Under the `Policies` section, ensure that at least one alert policy exists for the log metric above. Clicking on the policy should show that it is configured with a condition. For example, `Violates when: Any logging.googleapis.com/user/<Log Metric Name> stream` `is above a threshold of 0 for greater than 0 seconds` means that the alert will trigger for any new owner change. Verify that the chosen alerting thresholds make sense for the user's organization.
5. Ensure that appropriate notification channels have been set up.

From Command Line:
Ensure the log metric is present:

1. List the log metrics:

```
gcloud beta logging metrics list --format json
```

2. Ensure that the output contains at least one metric with filter set to:

```
resource.type=gce_network
AND protoPayload.methodName="beta.compute.networks.insert"
OR protoPayload.methodName="beta.compute.networks.patch"
OR protoPayload.methodName="v1.compute.networks.delete"
OR protoPayload.methodName="v1.compute.networks.removePeering"
OR protoPayload.methodName="v1.compute.networks.addPeering"
```

3. Note the value of the property `metricDescriptor.type` for the identified metric, in the format `logging.googleapis.com/user/<Log Metric Name>`.

Ensure the prescribed alerting policy is present:

4. List the alerting policies:

```
gcloud alpha monitoring policies list --format json
```

5. Ensure that the output contains at least one alert policy where:

- `conditions.conditionThreshold.filter` is set to `metric.type=\"logging.googleapis.com/user/<Log Metric Name>\"`
- AND `enabled` is set to `true`

Remediation:

From Console:
Create the prescribed log metric:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics and click "CREATE METRIC".

2. Click the down arrow symbol on `Filter Bar` at the rightmost corner and select `Convert to Advanced Filter`.
3. Clear any text and add:

```
resource.type=gce_network
AND protoPayload.methodName="beta.compute.networks.insert"
OR protoPayload.methodName="beta.compute.networks.patch"
OR protoPayload.methodName="v1.compute.networks.delete"
OR protoPayload.methodName="v1.compute.networks.removePeering"
OR protoPayload.methodName="v1.compute.networks.addPeering"
```

4. Click `Submit Filter`. Display logs appear based on the filter text entered by the user.
5. In the `Metric Editor` menu on the right, fill out the name field. Set `Units` to `1` (default) and `Type` to `Counter`. This ensures that the log metric counts the number of log entries matching the user's advanced logs query.
6. Click `Create Metric`.

Create the prescribed alert policy:

1. Identify the newly created metric under the section `User-defined Metrics` at https://console.cloud.google.com/logs/metrics.
2. Click the 3-dot icon in the rightmost column for the new metric and select `Create alert from Metric`. A new page appears.
3. Fill out the alert policy configuration and click `Save`. Choose the alerting threshold and configuration that makes sense for the user's organization. For example, a threshold of 0 for the most recent value will ensure that a notification is triggered for every owner change in the project:

```
Set `Aggregator` to `Count`

Set `Configuration`:

- Condition: above

- Threshold: 0

- For: most recent value
```

4. Configure the desired notification channels in the section `Notifications`.
5. Name the policy and click `Save`.

From Command Line:

Create the prescribed Log Metric:

- Use the command: gcloud beta logging metrics create

- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/beta/logging/metrics/create

Create the prescribed alert policy:

- Use the command: gcloud alpha monitoring policies create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/alpha/monitoring/policies/create

References:

1. https://cloud.google.com/logging/docs/logs-based-metrics/
2. https://cloud.google.com/monitoring/custom-metrics/
3. https://cloud.google.com/monitoring/alerts/
4. https://cloud.google.com/logging/docs/reference/tools/gcloud-logging
5. https://cloud.google.com/vpc/docs/overview

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

2.10 Ensure that the log metric filter and alerts exist for Cloud Storage IAM permission changes (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended that a metric filter and alarm be established for Cloud Storage Bucket IAM changes.

Rationale:

Monitoring changes to cloud storage bucket permissions may reduce the time needed to detect and correct permissions on sensitive cloud storage buckets and objects inside the bucket.

Impact:

Enabling of logging may result in your project being charged for the additional logs usage.

Audit:

From Console:
Ensure the prescribed log metric is present:

1. For each project that contains cloud storage buckets, go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics.
2. In the `User-defined Metrics` section, ensure at least one metric `<Log_Metric_Name>` is present with the filter text:

```
resource.type=gcs_bucket
AND protoPayload.methodName="storage.setIamPermissions"
```

Ensure that the prescribed alerting policy is present:

3. Go to `Alerting` by visiting https://console.cloud.google.com/monitoring/alerting.
4. Under the `Policies` section, ensure that at least one alert policy exists for the log metric above. Clicking on the policy should show that it is configured with a condition. For example, `Violates when: Any logging.googleapis.com/user/<Log Metric Name> stream` `is above a threshold of 0 for greater than 0 seconds` means that the alert will trigger for any new owner change. Verify that the chosen alerting thresholds make sense for the user's organization.
5. Ensure that the appropriate notifications channels have been set up.

From Command Line:
Ensure that the prescribed log metric is present:

1. List the log metrics:

```
gcloud beta logging metrics list --format json
```

2. Ensure that the output contains at least one metric with the filter set to:

```
resource.type=gcs_bucket
AND protoPayload.methodName="storage.setIamPermissions"
```

3. Note the value of the property `metricDescriptor.type` for the identified metric, in the format `logging.googleapis.com/user/<Log Metric Name>`.

Ensure the prescribed alerting policy is present:

4. List the alerting policies:

```
gcloud alpha monitoring policies list --format json
```

5. Ensure that the output contains an least one alert policy where:

- `conditions.conditionThreshold.filter` is set to `metric.type=\"logging.googleapis.com/user/<Log Metric Name>\"`
- AND `enabled` is set to `true`

Remediation:

From Console:
Create the prescribed log metric:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics and click "CREATE METRIC".
2. Click the down arrow symbol on the `Filter Bar` at the rightmost corner and select `Convert to Advanced Filter`.
3. Clear any text and add:

```
resource.type=gcs_bucket
AND protoPayload.methodName="storage.setIamPermissions"
```

4. Click `Submit Filter`. Display logs appear based on the filter text entered by the user.

5. In the `Metric Editor` menu on right, fill out the name field. Set `Units` to `1` (default) and `Type` to `Counter`. This ensures that the log metric counts the number of log entries matching the user's advanced logs query.
6. Click `Create Metric`.

Create the prescribed Alert Policy:

1. Identify the newly created metric under the section `User-defined Metrics` at https://console.cloud.google.com/logs/metrics.
2. Click the 3-dot icon in the rightmost column for the new metric and select `Create alert from Metric`. A new page appears.
3. Fill out the alert policy configuration and click `Save`. Choose the alerting threshold and configuration that makes sense for the user's organization. For example, a threshold of zero(0) for the most recent value will ensure that a notification is triggered for every owner change in the project:

```
Set `Aggregator` to `Count`

Set `Configuration`:

- Condition: above

- Threshold: 0

- For: most recent value
```

4. Configure the desired notifications channels in the section `Notifications`.
5. Name the policy and click `Save`.

From Command Line:
Create the prescribed Log Metric:

- Use the command: gcloud beta logging metrics create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/beta/logging/metrics/create

Create the prescribed alert policy:

- Use the command: gcloud alpha monitoring policies create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/alpha/monitoring/policies/create

References:

1. https://cloud.google.com/logging/docs/logs-based-metrics/
2. https://cloud.google.com/monitoring/custom-metrics/
3. https://cloud.google.com/monitoring/alerts/
4. https://cloud.google.com/logging/docs/reference/tools/gcloud-logging
5. https://cloud.google.com/storage/docs/overview
6. https://cloud.google.com/storage/docs/access-control/iam-roles

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

2.11 Ensure that the log metric filter and alerts exist for SQL instance configuration changes (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended that a metric filter and alarm be established for SQL instance configuration changes.

Rationale:

Monitoring changes to SQL instance configuration changes may reduce the time needed to detect and correct misconfigurations done on the SQL server.

Below are a few of the configurable options which may the impact security posture of an SQL instance:

- Enable auto backups and high availability: Misconfiguration may adversely impact business continuity, disaster recovery, and high availability
- Authorize networks: Misconfiguration may increase exposure to untrusted networks

Impact:

Enabling of logging may result in your project being charged for the additional logs usage.

Audit:

From Console:
Ensure the prescribed log metric is present:

1. For each project that contains Cloud SQL instances, go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics.
2. In the `User-defined Metrics` section, ensure that at least one metric `<Log_Metric_Name>` is present with the filter text:

```
protoPayload.methodName="cloudsql.instances.update"
```

Ensure that the prescribed alerting policy is present:

3. Go to `Alerting` by visiting https://console.cloud.google.com/monitoring/alerting.
4. Under the `Policies` section, ensure that at least one alert policy exists for the log metric above. Clicking on the policy should show that it is configured with a condition. For example, `Violates when: Any logging.googleapis.com/user/<Log Metric Name> stream` `is above a threshold of zero(0) for greater than zero(0) seconds` means that the alert will trigger for any new owner change. Verify that the chosen alerting thresholds make sense for the user's organization.
5. Ensure that the appropriate notifications channels have been set up.

From Command Line:
Ensure that the prescribed log metric is present:

1. List the log metrics:

```
gcloud beta logging metrics list --format json
```

2. Ensure that the output contains at least one metric with the filter set to

```
protoPayload.methodName="cloudsql.instances.update"
```

3. Note the value of the property `metricDescriptor.type` for the identified metric, in the format `logging.googleapis.com/user/<Log Metric Name>`.

Ensure that the prescribed alerting policy is present:

4. List the alerting policies:

```
gcloud alpha monitoring policies list --format json
```

5. Ensure that the output contains at least one alert policy where:

- `conditions.conditionThreshold.filter` is set to `metric.type=\"logging.googleapis.com/user/<Log Metric Name>\"`
- AND `enabled` is set to `true`

Remediation:

From Console:
Create the prescribed Log Metric:

1. Go to `Logging/Logs-based Metrics` by visiting https://console.cloud.google.com/logs/metrics and click "CREATE METRIC".
2. Click the down arrow symbol on the `Filter Bar` at the rightmost corner and select `Convert to Advanced Filter`.
3. Clear any text and add:

```
protoPayload.methodName="cloudsql.instances.update"
```

4. Click `Submit Filter`. Display logs appear based on the filter text entered by the user.
5. In the `Metric Editor` menu on right, fill out the name field. Set `Units` to `1` (default) and `Type` to `Counter`. This ensures that the log metric counts the number of log entries matching the user's advanced logs query.
6. Click `Create Metric`.

Create the prescribed alert policy:

1. Identify the newly created metric under the section `User-defined Metrics` at https://console.cloud.google.com/logs/metrics.
2. Click the 3-dot icon in the rightmost column for the new metric and select `Create alert from Metric`. A new page appears.
3. Fill out the alert policy configuration and click `Save`. Choose the alerting threshold and configuration that makes sense for the user's organization. For example, a threshold of zero(0) for the most recent value will ensure that a notification is triggered for every owner change in the user's project:

```
Set `Aggregator` to `Count`

Set `Configuration`:

- Condition: above

- Threshold: 0

- For: most recent value
```

4. Configure the desired notification channels in the section `Notifications`.
5. Name the policy and click `Save`.

From Command Line:

Create the prescribed log metric:

- Use the command: gcloud beta logging metrics create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/beta/logging/metrics/create

Create the prescribed alert policy:

- Use the command: gcloud alpha monitoring policies create
- Reference for command usage: https://cloud.google.com/sdk/gcloud/reference/alpha/monitoring/policies/create

References:

1. https://cloud.google.com/logging/docs/logs-based-metrics/
2. https://cloud.google.com/monitoring/custom-metrics/
3. https://cloud.google.com/monitoring/alerts/
4. https://cloud.google.com/logging/docs/reference/tools/gcloud-logging
5. https://cloud.google.com/storage/docs/overview
6. https://cloud.google.com/sql/docs/
7. https://cloud.google.com/sql/docs/mysql/
8. https://cloud.google.com/sql/docs/postgres/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

2.12 Ensure that Cloud DNS logging is enabled for all VPC networks (Automated)

Profile Applicability:

- Level 1

Description:

Cloud DNS logging records the queries from the name servers within your VPC to Stackdriver. Logged queries can come from Compute Engine VMs, GKE containers, or other GCP resources provisioned within the VPC.

Rationale:

Security monitoring and forensics cannot depend solely on IP addresses from VPC flow logs, especially when considering the dynamic IP usage of cloud resources, HTTP virtual host routing, and other technology that can obscure the DNS name used by a client from the IP address. Monitoring of Cloud DNS logs provides visibility to DNS names requested by the clients within the VPC. These logs can be monitored for anomalous domain names, evaluated against threat intelligence, and

Note: For full capture of DNS, firewall must block egress UDP/53 (DNS) and TCP/443 (DNS over HTTPS) to prevent client from using external DNS name server for resolution.

Impact:

Enabling of Cloud DNS logging might result in your project being charged for the additional logs usage.

Audit:

From Command Line:

1. List all VPCs networks in a project:

```
gcloud compute networks list --format="table[box,title='All VPC
Networks'](name:label='VPC Network Name')"
```

2. List all DNS policies, logging enablement, and associated VPC networks:

```
gcloud dns policies list --flatten="networks[]" --
format="table[box,title='All DNS Policies By VPC Network'](name:label='Policy
Name',enableLogging:label='Logging
Enabled':align=center,networks.networkUrl.basename():label='VPC Network
Name')"
```

Each VPC Network should be associated with a DNS policy with logging enabled.

Remediation:

From Command Line:
Add New DNS Policy With Logging Enabled
For each VPC network that needs a DNS policy with logging enabled:

```
gcloud dns policies create enable-dns-logging --enable-logging --
description="Enable DNS Logging" --networks=VPC_NETWORK_NAME
```

The VPC_NETWORK_NAME can be one or more networks in comma-separated list
Enable Logging for Existing DNS Policy
For each VPC network that has an existing DNS policy that needs logging enabled:

```
gcloud dns policies update POLICY_NAME --enable-logging --
networks=VPC_NETWORK_NAME
```

The VPC_NETWORK_NAME can be one or more networks in comma-separated list

Default Value:

Cloud DNS logging is disabled by default on each network.

References:

1. https://cloud.google.com/dns/docs/monitoring

Additional Information:

Additional Info

- Only queries that reach a name server are logged. Cloud DNS resolvers cache responses, queries answered from caches, or direct queries to an external DNS resolver outside the VPC are not logged.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.7 Regularly Review Logs On a regular basis, review logs to identify anomalies or abnormal events.		●	●
v7	8.7 Enable DNS Query Logging Enable Domain Name System (DNS) query logging to detect hostname lookups for known malicious domains.		●	●

3 Networking

This section covers recommendations addressing networking on Google Cloud Platform.

3.1 Ensure that the default network does not exist in a project (Automated)

Profile Applicability:

- Level 2

Description:

To prevent use of `default` network, a project should not have a `default` network.

Rationale:

The `default` network has a preconfigured network configuration and automatically generates the following insecure firewall rules:

- default-allow-internal: Allows ingress connections for all protocols and ports among instances in the network.
- default-allow-ssh: Allows ingress connections on TCP port 22(SSH) from any source to any instance in the network.
- default-allow-rdp: Allows ingress connections on TCP port 3389(RDP) from any source to any instance in the network.
- default-allow-icmp: Allows ingress ICMP traffic from any source to any instance in the network.

These automatically created firewall rules do not get audit logged and cannot be configured to enable firewall rule logging.

Furthermore, the default network is an auto mode network, which means that its subnets use the same predefined range of IP addresses, and as a result, it's not possible to use Cloud VPN or VPC Network Peering with the default network.

Based on organization security and networking requirements, the organization should create a new network and delete the `default` network.

Impact:

When an organization deletes the default network, it may need to migrate or service onto a new network.

Audit:

From Console:

1. Go to the `VPC networks` page by visiting: https://console.cloud.google.com/networking/networks/list.
2. Ensure that a network with the name `default` is not present.

From Command Line:

1. Set the project name in the Google Cloud Shell:

```
gcloud config set project PROJECT_ID
```

2. List the networks configured in that project:

```
gcloud compute networks list
```

It should not list `default` as one of the available networks in that project.

Remediation:

From Console:

1. Go to the `VPC networks` page by visiting: https://console.cloud.google.com/networking/networks/list.
2. Click the network named `default`.
3. On the network detail page, click `EDIT`.
4. Click `DELETE VPC NETWORK`.
5. If needed, create a new network to replace the default network.

From Command Line:
For each Google Cloud Platform project,

1. Delete the default network:

```
gcloud compute networks delete default
```

2. If needed, create a new network to replace it:

```
gcloud compute networks create NETWORK_NAME
```

Prevention:
The user can prevent the default network and its insecure default firewall rules from being created by setting up an Organization Policy to `Skip default network creation` at https://console.cloud.google.com/iam-admin/orgpolicies/compute-skipDefaultNetworkCreation.

Default Value:

By default, for each project, a `default` network is created.

References:

1. https://cloud.google.com/compute/docs/networking#firewall_rules
2. https://cloud.google.com/compute/docs/reference/latest/networks/insert
3. https://cloud.google.com/compute/docs/reference/latest/networks/delete
4. https://cloud.google.com/vpc/docs/firewall-rules-logging
5. https://cloud.google.com/vpc/docs/vpc#default-network
6. https://cloud.google.com/sdk/gcloud/reference/compute/networks/delete

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	11.1 Maintain Standard Security Configurations for Network Devices Maintain standard, documented security configuration standards for all authorized network devices.		●	●

3.2 Ensure legacy networks do not exist for a project (Automated)

Profile Applicability:

- Level 1

Description:

In order to prevent use of legacy networks, a project should not have a legacy network configured.

Rationale:

Legacy networks have a single network IPv4 prefix range and a single gateway IP address for the whole network. The network is global in scope and spans all cloud regions. Subnetworks cannot be created in a legacy network and are unable to switch from legacy to auto or custom subnet networks. Legacy networks can have an impact for high network traffic projects and are subject to a single point of contention or failure.

Impact:

None.

Audit:

For each Google Cloud Platform project,

1. Set the project name in the Google Cloud Shell:

```
gcloud config set project <Project-ID>
```

2. List the networks configured in that project:

```
gcloud compute networks list
```

None of the listed networks should be in the `legacy` mode.

Remediation:

For each Google Cloud Platform project,

1. Follow the documentation and create a non-legacy network suitable for the organization's requirements.

2. Follow the documentation and delete the networks in the `legacy` mode.

Default Value:

By default, networks are not created in the `legacy` mode.

References:

1. https://cloud.google.com/vpc/docs/using-legacy#creating_a_legacy_network
2. https://cloud.google.com/vpc/docs/using-legacy#deleting_a_legacy_network

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	11.1 Maintain Standard Security Configurations for Network Devices Maintain standard, documented security configuration standards for all authorized network devices.		●	●

3.3 Ensure that DNSSEC is enabled for Cloud DNS (Automated)

Profile Applicability:

- Level 1

Description:

Cloud Domain Name System (DNS) is a fast, reliable and cost-effective domain name system that powers millions of domains on the internet. Domain Name System Security Extensions (DNSSEC) in Cloud DNS enables domain owners to take easy steps to protect their domains against DNS hijacking and man-in-the-middle and other attacks.

Rationale:

Domain Name System Security Extensions (DNSSEC) adds security to the DNS protocol by enabling DNS responses to be validated. Having a trustworthy DNS that translates a domain name like www.example.com into its associated IP address is an increasingly important building block of today's web-based applications. Attackers can hijack this process of domain/IP lookup and redirect users to a malicious site through DNS hijacking and man-in-the-middle attacks. DNSSEC helps mitigate the risk of such attacks by cryptographically signing DNS records. As a result, it prevents attackers from issuing fake DNS responses that may misdirect browsers to nefarious websites.

Audit:

From Console:

1. Go to `Cloud DNS` by visiting https://console.cloud.google.com/net-services/dns/zones.
2. For each zone of `Type Public`, ensure that `DNSSEC` is set to `On`.

From Command Line:

1. List all the Managed Zones in a project:

```
gcloud dns managed-zones list
```

2. For each zone of `VISIBILITY public`, get its metadata:

```
gcloud dns managed-zones describe ZONE_NAME
```

3. Ensure that `dnssecConfig.state` property is `on`.

Remediation:

From Console:

1. Go to `Cloud DNS` by visiting https://console.cloud.google.com/net-services/dns/zones.
2. For each zone of `Type Public`, set `DNSSEC` to `On`.

From Command Line:
Use the below command to enable `DNSSEC` for Cloud DNS Zone Name.

```
gcloud dns managed-zones update ZONE_NAME --dnssec-state on
```

Default Value:

By default DNSSEC is not enabled.

References:

1. https://cloudplatform.googleblog.com/2017/11/DNSSEC-now-available-in-Cloud-DNS.html
2. https://cloud.google.com/dns/dnssec-config#enabling
3. https://cloud.google.com/dns/dnssec

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	11.1 Maintain Standard Security Configurations for Network Devices Maintain standard, documented security configuration standards for all authorized network devices.		●	●

3.4 Ensure that RSASHA1 is not used for the key-signing key in Cloud DNS DNSSEC (Manual)

Profile Applicability:

- Level 1

Description:

DNSSEC algorithm numbers in this registry may be used in CERT RRs. Zone signing (DNSSEC) and transaction security mechanisms (SIG(0) and TSIG) make use of particular subsets of these algorithms. The algorithm used for key signing should be a recommended one and it should be strong.

Rationale:

Domain Name System Security Extensions (DNSSEC) algorithm numbers in this registry may be used in CERT RRs. Zonesigning (DNSSEC) and transaction security mechanisms (SIG(0) and TSIG) make use of particular subsets of these algorithms.

The algorithm used for key signing should be a recommended one and it should be strong. When enabling DNSSEC for a managed zone, or creating a managed zone with DNSSEC, the user can select the DNSSEC signing algorithms and the denial-of-existence type. Changing the DNSSEC settings is only effective for a managed zone if DNSSEC is not already enabled. If there is a need to change the settings for a managed zone where it has been enabled, turn DNSSEC off and then re-enable it with different settings.

Audit:

Currently there is no support to audit this setting through console.
From Command Line:
Ensure the property algorithm for keyType keySigning is not using `RSASHA1`.

```
gcloud dns managed-zones describe ZONENAME --
format="json(dnsName,dnssecConfig.state,dnssecConfig.defaultKeySpecs)"
```

Remediation:

1. If it is necessary to change the settings for a managed zone where it has been enabled, NSSEC must be turned off and re-enabled with different settings. To turn off DNSSEC, run the following command:

```
gcloud dns managed-zones update ZONE_NAME --dnssec-state off
```

2. To update key-signing for a reported managed DNS Zone, run the following command:

```
gcloud dns managed-zones update ZONE_NAME --dnssec-state on --ksk-algorithm
KSK_ALGORITHM --ksk-key-length KSK_KEY_LENGTH --zsk-algorithm ZSK_ALGORITHM -
-zsk-key-length ZSK_KEY_LENGTH --denial-of-existence DENIAL_OF_EXISTENCE
```

Supported algorithm options and key lengths are as follows.

```
Algorithm                        KSK Length               ZSK Length
---------                        ----------               ----------
RSASHA1                          1024,2048                1024,2048
RSASHA256                        1024,2048                1024,2048
RSASHA512                        1024,2048                1024,2048
ECDSAP256SHA256                  256                      256
ECDSAP384SHA384                  384                      384
```

References:

1. https://cloud.google.com/dns/dnssec-advanced#advanced_signing_options

Additional Information:

1. RSASHA1 key-signing support may be required for compatibility reasons.
2. Remediation CLI works well with gcloud-cli version 221.0.0 and later.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	11.1 Maintain Standard Security Configurations for Network Devices Maintain standard, documented security configuration standards for all authorized network devices.		●	●

3.5 Ensure that RSASHA1 is not used for the zone-signing key in Cloud DNS DNSSEC (Manual)

Profile Applicability:

- Level 1

Description:

DNSSEC algorithm numbers in this registry may be used in CERT RRs. Zone signing (DNSSEC) and transaction security mechanisms (SIG(0) and TSIG) make use of particular subsets of these algorithms. The algorithm used for key signing should be a recommended one and it should be strong.

Rationale:

DNSSEC algorithm numbers in this registry may be used in CERT RRs. Zone signing (DNSSEC) and transaction security mechanisms (SIG(0) and TSIG) make use of particular subsets of these algorithms.

The algorithm used for key signing should be a recommended one and it should be strong. When enabling DNSSEC for a managed zone, or creating a managed zone with DNSSEC, the DNSSEC signing algorithms and the denial-of-existence type can be selected. Changing the DNSSEC settings is only effective for a managed zone if DNSSEC is not already enabled. If the need exists to change the settings for a managed zone where it has been enabled, turn DNSSEC off and then re-enable it with different settings.

Audit:

Currently there is no support to audit this setting through the console.
From Command Line:
Ensure the property algorithm for keyType zone signing is not using RSASHA1.

```
gcloud dns managed-zones describe --
format="json(dnsName,dnssecConfig.state,dnssecConfig.defaultKeySpecs)"
```

Remediation:

1. If the need exists to change the settings for a managed zone where it has been enabled, DNSSEC must be turned off and then re-enabled with different settings. To turn off DNSSEC, run following command:

```
gcloud dns managed-zones update ZONE_NAME --dnssec-state off
```

2. To update zone-signing for a reported managed DNS Zone, run the following command:

```
gcloud dns managed-zones update ZONE_NAME --dnssec-state on --ksk-algorithm
KSK_ALGORITHM --ksk-key-length KSK_KEY_LENGTH --zsk-algorithm ZSK_ALGORITHM -
-zsk-key-length ZSK_KEY_LENGTH --denial-of-existence DENIAL_OF_EXISTENCE
```

Supported algorithm options and key lengths are as follows.

```
Algorithm                KSK Length           ZSK Length
---------                ----------           ----------
RSASHA1                  1024,2048            1024,2048
RSASHA256                1024,2048            1024,2048
RSASHA512                1024,2048            1024,2048
ECDSAP256SHA256          256                  384
ECDSAP384SHA384          384                  384
```

References:

1. https://cloud.google.com/dns/dnssec-advanced#advanced_signing_options

Additional Information:

1. RSASHA1 zone-signing support may be required for compatibility reasons.
2. The remediation CLI works well with gcloud-cli version 221.0.0 and later.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	11.1 Maintain Standard Security Configurations for Network Devices Maintain standard, documented security configuration standards for all authorized network devices.		●	●

3.6 Ensure that SSH access is restricted from the internet (Automated)

Profile Applicability:

- Level 2

Description:

GCP `Firewall Rules` are specific to a `VPC Network`. Each rule either `allows` or `denies` traffic when its conditions are met. Its conditions allow the user to specify the type of traffic, such as ports and protocols, and the source or destination of the traffic, including IP addresses, subnets, and instances.

Firewall rules are defined at the VPC network level and are specific to the network in which they are defined. The rules themselves cannot be shared among networks. Firewall rules only support IPv4 traffic. When specifying a source for an ingress rule or a destination for an egress rule by address, only an `IPv4` address or `IPv4 block in CIDR` notation can be used. Generic `(0.0.0.0/0)` incoming traffic from the internet to VPC or VM instance using `SSH` on `Port 22` can be avoided.

Rationale:

GCP `Firewall Rules` within a `VPC Network` apply to outgoing (egress) traffic from instances and incoming (ingress) traffic to instances in the network. Egress and ingress traffic flows are controlled even if the traffic stays within the network (for example, instance-to-instance communication). For an instance to have outgoing Internet access, the network must have a valid Internet gateway route or custom route whose destination IP is specified. This route simply defines the path to the Internet, to avoid the most general `(0.0.0.0/0)` destination `IP Range` specified from the Internet through `SSH` with the default `Port 22`. Generic access from the Internet to a specific IP Range needs to be restricted.

Impact:

All Secure Shell (SSH) connections from outside of the network to the concerned VPC(s) will be blocked. There could be a business need where SSH access is required from outside of the network to access resources associated with the VPC. In that case, specific source IP(s) should be mentioned in firewall rules to white-list access to SSH port for the concerned VPC(s).

Audit:

From the Console:

1. Go to `VPC network`.
2. Go to the `Firewall Rules`.
3. Ensure that `Port` is not equal to `22` and `Action` is not set to `Allow`.
4. Ensure `IP Ranges` is not equal to `0.0.0.0/0` under `Source filters`.

From Command Line:

```
gcloud compute firewall-rules list --
format=table'(name,direction,sourceRanges,allowed)'
```

Ensure that there is no rule matching the below criteria:

- `SOURCE_RANGES` is `0.0.0.0/0`
- AND `DIRECTION` is `INGRESS`
- AND IPProtocol is `tcp` or `ALL`
- AND `PORTS` is set to `22` or `range containing 22` or `Null (not set)`

Note:

- When ALL TCP ports are allowed in a rule, PORT does not have any value set (`NULL`)
- When ALL Protocols are allowed in a rule, PORT does not have any value set (`NULL`)

Remediation:

From the Console:

1. Go to `VPC Network`.
2. Go to the `Firewall Rules`.
3. Click the `Firewall Rule` you want to modify.
4. Click `Edit`.
5. Modify `Source IP ranges` to specific `IP`.
6. Click `Save`.

From Command Line:

1.Update the Firewall rule with the new `SOURCE_RANGE` from the below command:

```
gcloud compute firewall-rules update FirewallName --allow=[PROTOCOL[:PORT[-
PORT]],...] --source-ranges=[CIDR_RANGE,...]
```

References:

1. https://cloud.google.com/vpc/docs/firewalls#blockedtraffic

Additional Information:

Currently, GCP VPC only supports IPV4; however, Google is already working on adding IPV6 support for VPC. In that case along with source IP range `0.0.0.0`, the rule should be checked for IPv6 equivalent `::0` as well.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●
v7	12.4 Deny Communication over Unauthorized Ports Deny communication over unauthorized TCP or UDP ports or application traffic to ensure that only authorized protocols are allowed to cross the network boundary in or out of the network at each of the organization's network boundaries.	●	●	●

3.7 Ensure that RDP access is restricted from the Internet (Automated)

Profile Applicability:

- Level 2

Description:

GCP `Firewall Rules` are specific to a `VPC Network`. Each rule either `allows` or `denies` traffic when its conditions are met. Its conditions allow users to specify the type of traffic, such as ports and protocols, and the source or destination of the traffic, including IP addresses, subnets, and instances.

Firewall rules are defined at the VPC network level and are specific to the network in which they are defined. The rules themselves cannot be shared among networks. Firewall rules only support IPv4 traffic. When specifying a source for an ingress rule or a destination for an egress rule by address, an `IPv4` address or `IPv4 block in CIDR` notation can be used. Generic `(0.0.0.0/0)` incoming traffic from the Internet to a VPC or VM instance using `RDP` on `Port 3389` can be avoided.

Rationale:

GCP `Firewall Rules` within a `VPC Network`. These rules apply to outgoing (egress) traffic from instances and incoming (ingress) traffic to instances in the network. Egress and ingress traffic flows are controlled even if the traffic stays within the network (for example, instance-to-instance communication). For an instance to have outgoing Internet access, the network must have a valid Internet gateway route or custom route whose destination IP is specified. This route simply defines the path to the Internet, to avoid the most general `(0.0.0.0/0)` destination `IP Range` specified from the Internet through `RDP` with the default `Port 3389`. Generic access from the Internet to a specific IP Range should be restricted.

Impact:

All Remote Desktop Protocol (RDP) connections from outside of the network to the concerned VPC(s) will be blocked. There could be a business need where secure shell access is required from outside of the network to access resources associated with the VPC. In that case, specific source IP(s) should be mentioned in firewall rules to white-list access to RDP port for the concerned VPC(s).

Audit:

From the Console:

1. Go to `VPC network`.
2. Go to the `Firewall Rules`.
3. Ensure `Port` is not equal to `3389` and `Action` is not `Allow`.
4. Ensure `IP Ranges` is not equal to `0.0.0.0/0` under `Source filters`.

From Command Line:

```
gcloud compute firewall-rules list --
format=table'(name,direction,sourceRanges,allowed.ports)'
```

Ensure that there is no rule matching the below criteria:

- `SOURCE_RANGES` is `0.0.0.0/0`
- AND `DIRECTION` is `INGRESS`
- AND IPProtocol is `TCP` or `ALL`
- AND `PORTS` is set to `3389` or `range containing 3389` or `Null (not set)`

Note:

- When ALL TCP ports are allowed in a rule, PORT does not have any value set (`NULL`)
- When ALL Protocols are allowed in a rule, PORT does not have any value set (`NULL`)

Remediation:

From the Console:

1. Go to `VPC Network`.
2. Go to the `Firewall Rules`.
3. Click the `Firewall Rule` to be modified.
4. Click `Edit`.
5. Modify `Source IP ranges` to specific `IP`.
6. Click `Save`.

From Command Line:

1.Update RDP Firewall rule with new `SOURCE_RANGE` from the below command:

```
gcloud compute firewall-rules update FirewallName --allow=[PROTOCOL[:PORT[-
PORT]],...] --source-ranges=[CIDR_RANGE,...]
```

References:

1. https://cloud.google.com/vpc/docs/firewalls#blockedtraffic

Additional Information:

Currently, GCP VPC only supports IPV4; however, Google is already working on adding IPV6 support for VPC. In that case along with source IP range `0.0.0.0`, the rule should be checked for IPv6 equivalent `::0` as well.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●
v7	12.4 Deny Communication over Unauthorized Ports Deny communication over unauthorized TCP or UDP ports or application traffic to ensure that only authorized protocols are allowed to cross the network boundary in or out of the network at each of the organization's network boundaries.	●	●	●

3.8 Ensure that VPC Flow Logs is enabled for every subnet in a VPC Network (Automated)

Profile Applicability:

- Level 1

Description:

Flow Logs is a feature that enables users to capture information about the IP traffic going to and from network interfaces in the organization's VPC Subnets. Once a flow log is created, the user can view and retrieve its data in Stackdriver Logging. It is recommended that Flow Logs be enabled for every business-critical VPC subnet.

Rationale:

VPC networks and subnetworks not reserved for internal HTTP(S) load balancing provide logically isolated and secure network partitions where GCP resources can be launched. When Flow Logs are enabled for a subnet, VMs within that subnet start reporting on all Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) flows. Each VM samples the TCP and UDP flows it sees, inbound and outbound, whether the flow is to or from another VM, a host in the on-premises datacenter, a Google service, or a host on the Internet. If two GCP VMs are communicating, and both are in subnets that have VPC Flow Logs enabled, both VMs report the flows.

Flow Logs supports the following use cases:

- Network monitoring
- Understanding network usage and optimizing network traffic expenses
- Network forensics
- Real-time security analysis

Flow Logs provide visibility into network traffic for each VM inside the subnet and can be used to detect anomalous traffic or provide insight during security workflows.

Note: Subnets reserved for use by internal HTTP(S) load balancers do not support VPC flow logs.

Impact:

Standard pricing for Stackdriver Logging, BigQuery, or Cloud Pub/Sub applies. VPC Flow Logs generation will be charged starting in GA as described in reference: https://cloud.google.com/vpc/

Audit:

From Console:

1. Go to the VPC network GCP Console visiting `https://console.cloud.google.com/networking/networks/list`
2. From the list of network subnets,
   make sure for each subnet `Flow Logs` is set to `On`

From Command Line:

```
gcloud compute networks list --format json | \
  jq -r '.[].subnetworks | .[]' | \
  xargs -I {} gcloud compute networks subnets describe {} --format json | \
  jq -r '. | "Subnet: \(.name)    Purpose: \(.purpose)    VPC Flow Log 
Enabled: \(has("enableFlowLogs"))"'
```

The output of the above command will list each subnet, the subnet's purpose, and a `true` or `false` value if `Flow Logs` are enabled.
If the subnet's purpose is `PRIVATE` then `Flow Logs` should be `true`.

Remediation:

From Console:

1. Go to the VPC network GCP Console visiting `https://console.cloud.google.com/networking/networks/list`
2. Click the name of a subnet, The `Subnet details` page displays.
3. Click the `EDIT` button.
4. Set `Flow Logs` to `On`.
5. Click Save.

From Command Line:
To set Private Google access for a network subnet, run the following command:

```
gcloud compute networks subnets update [SUBNET_NAME] --region [REGION] --
enable-flow-logs
```

Default Value:

By default, Flow Logs is set to Off when a new VPC network subnet is created.

References:

1. https://cloud.google.com/vpc/docs/using-flow-logs#enabling_vpc_flow_logging
2. https://cloud.google.com/vpc/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	12.8 Deploy NetFlow Collection on Networking Boundary Devices Enable the collection of NetFlow and logging data on all network boundary devices.		●	●

3.9 Ensure no HTTPS or SSL proxy load balancers permit SSL policies with weak cipher suites (Manual)

Profile Applicability:

- Level 1

Description:

Secure Sockets Layer (SSL) policies determine what port Transport Layer Security (TLS) features clients are permitted to use when connecting to load balancers. To prevent usage of insecure features, SSL policies should use (a) at least TLS 1.2 with the MODERN profile; or (b) the RESTRICTED profile, because it effectively requires clients to use TLS 1.2 regardless of the chosen minimum TLS version; or (3) a CUSTOM profile that does not support any of the following features:

```
TLS_RSA_WITH_AES_128_GCM_SHA256

TLS_RSA_WITH_AES_256_GCM_SHA384

TLS_RSA_WITH_AES_128_CBC_SHA

TLS_RSA_WITH_AES_256_CBC_SHA

TLS_RSA_WITH_3DES_EDE_CBC_SHA
```

Rationale:

Load balancers are used to efficiently distribute traffic across multiple servers. Both SSL proxy and HTTPS load balancers are external load balancers, meaning they distribute traffic from the Internet to a GCP network. GCP customers can configure load balancer SSL policies with a minimum TLS version (1.0, 1.1, or 1.2) that clients can use to establish a connection, along with a profile (Compatible, Modern, Restricted, or Custom) that specifies permissible cipher suites. To comply with users using outdated protocols, GCP load balancers can be configured to permit insecure cipher suites. In fact, the GCP default SSL policy uses a minimum TLS version of 1.0 and a Compatible profile, which allows the widest range of insecure cipher suites. As a result, it is easy for customers to configure a load balancer without even knowing that they are permitting outdated cipher suites.

Impact:

Creating more secure SSL policies can prevent clients using older TLS versions from establishing a connection.

Audit:

From Console:

1. See all load balancers by visiting https://console.cloud.google.com/net-services/loadbalancing/loadBalancers/list.
2. For each load balancer for `SSL (Proxy)` or `HTTPS`, click on its name to go the `Load balancer details` page.
3. Ensure that each target proxy entry in the `Frontend` table has an `SSL Policy` configured.
4. Click on each SSL policy to go to its `SSL policy details` page.
5. Ensure that the SSL policy satisfies one of the following conditions:

- has a `Min TLS` set to `TLS 1.2` and `Profile` set to `Modern` profile, or
- has `Profile` set to `Restricted`. Note that a Restricted profile effectively requires clients to use TLS 1.2 regardless of the chosen minimum TLS version, or
- has `Profile` set to `Custom` and the following features are all disabled:

```
TLS_RSA_WITH_AES_128_GCM_SHA256
TLS_RSA_WITH_AES_256_GCM_SHA384
TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA
```

From Command Line:

1. List all TargetHttpsProxies and TargetSslProxies.

```
gcloud compute target-https-proxies list
gcloud compute target-ssl-proxies list
```

2. For each target proxy, list its properties:

```
gcloud compute target-https-proxies describe TARGET_HTTPS_PROXY_NAME
gcloud compute target-ssl-proxies describe TARGET_SSL_PROXY_NAME
```

3. Ensure that the `sslPolicy` field is present and identifies the name of the SSL policy:

```
sslPolicy:
https://www.googleapis.com/compute/v1/projects/PROJECT_ID/global/sslPolicies/
SSL_POLICY_NAME
```

If the `sslPolicy` field is missing from the configuration, it means that the GCP default policy is used, which is insecure.

4. Describe the SSL policy:

```
gcloud compute ssl-policies describe SSL_POLICY_NAME
```

5. Ensure that the policy satisfies one of the following conditions:

- has `Profile` set to `Modern` and `minTlsVersion` set to `TLS_1_2`, or
- has `Profile` set to `Restricted`, or
- has `Profile` set to `Custom` and `enabledFeatures` does not contain any of the following values:

```
TLS_RSA_WITH_AES_128_GCM_SHA256
TLS_RSA_WITH_AES_256_GCM_SHA384
TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA
```

Remediation:

From Console:
If the TargetSSLProxy or TargetHttpsProxy does not have an SSL policy configured, create a new SSL policy. Otherwise, modify the existing insecure policy.

1. Navigate to the `SSL Policies` page by visiting: https://console.cloud.google.com/net-security/sslpolicies
2. Click on the name of the insecure policy to go to its `SSL policy details` page.
3. Click `EDIT`.
4. Set `Minimum TLS version` to `TLS 1.2`.
5. Set `Profile` to `Modern` or `Restricted`.
6. Alternatively, if teh user selects the profile `Custom`, make sure that the following features are disabled:

```
TLS_RSA_WITH_AES_128_GCM_SHA256
TLS_RSA_WITH_AES_256_GCM_SHA384
TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA
```

From Command Line:

1. For each insecure SSL policy, update it to use secure cyphers:

```
gcloud compute ssl-policies update NAME [--profile
COMPATIBLE|MODERN|RESTRICTED|CUSTOM] --min-tls-version 1.2 [--custom-features
FEATURES]
```

2. If the target proxy has a GCP default SSL policy, use the following command corresponding to the proxy type to update it.

```
gcloud compute target-ssl-proxies update TARGET_SSL_PROXY_NAME --ssl-policy
SSL_POLICY_NAME
gcloud compute target-https-proxies update TARGET_HTTPS_POLICY_NAME --ssl-
policy SSL_POLICY_NAME
```

Default Value:

The GCP default SSL policy is the least secure setting: Min TLS 1.0 and Compatible profile

References:

1. https://cloud.google.com/load-balancing/docs/use-ssl-policies
2. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r2.pdf

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			

3.10 Ensure Firewall Rules for instances behind Identity Aware Proxy (IAP) only allow the traffic from Google Cloud Loadbalancer (GCLB) Health Check and Proxy Addresses (Manual)

Profile Applicability:

- Level 2

Description:

Access to VMs should be restricted by firewall rules that allow only IAP traffic by ensuring only connections proxied by the IAP are allowed. To ensure that load balancing works correctly health checks should also be allowed.

Rationale:

IAP ensure that access to VMs is controlled by authenticating incoming requests. However if the VM is still accessible from IP addresses other than the IAP it may still be possible to send unauthenticated requests to the instance. Care must be taken to ensure that loadblancer health checks are not blocked as this would stop the loadbalancer from correctly knowing the health of the VM and loadbalancing correctly.

Impact:

If firewall rules are not configured correctly, legitimate business services could be negatively impacted.

Audit:

From the Console:

1. Go to the Cloud Console VPC network > Firewall rules.
2. Verify that the only rules correspond to the following values:
    - **Targets**: All instances in the network
    - **Source IP ranges** (press Enter after you paste each value in the box):
        - 130.211.0.0/22
        - 35.191.0.0/16
    - **Protocols and ports**:
        - Specified protocols and ports
        - tcp:80

Remediation:

From the Console:

1. Go to the Cloud Console VPC network > Firewall rules.
2. Select the checkbox next to the following rules:
   - default-allow-http
   - default-allow-https
   - default-allow-internal
3. Click **Delete.**
4. Click **Create firewall rule** and set the following values:
   - **Name**: allow-iap-traffic
   - **Targets**: All instances in the network
   - **Source IP ranges** (press Enter after you paste each value in the box):
     - 130.211.0.0/22
     - 35.191.0.0/16
   - **Protocols and ports**:
     - Specified protocols and ports
     - tcp:80
5. When you're finished updating values, click **Create**.

Default Value:

By default all traffic is allowed.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	9 Limitation and Control of Network Ports, Protocols, and Services Limitation and Control of Network Ports, Protocols, and Services			

4 Virtual Machines

This section covers recommendations addressing virtual machines on Google Cloud Platform.

4.1 Ensure that instances are not configured to use the default service account (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to configure your instance to not use the default Compute Engine service account because it has the Editor role on the project.

Rationale:

The default Compute Engine service account has the Editor role on the project, which allows read and write access to most Google Cloud Services. To defend against privilege escalations if your VM is compromised and prevent an attacker from gaining access to all of your project, it is recommended to not use the default Compute Engine service account. Instead, you should create a new service account and assigning only the permissions needed by your instance.

The default Compute Engine service account is named `[PROJECT_NUMBER]-compute@developer.gserviceaccount.com`.

Impact:

None.

Audit:

From Console:

1. Go to the `VM instances` page by visiting: https://console.cloud.google.com/compute/instances.
2. Click on each instance name to go to its `VM instance details` page.
3. Under the section `Service Account`, ensure that the default Compute Engine service account is not used. This account is named `[PROJECT_NUMBER]-compute@developer.gserviceaccount.com`.

From Command Line:

1. List the instances in your project:

```
gcloud compute instances list
```

2. Get the details on each instance:

```
gcloud compute instances describe INSTANCE_NAME --zone ZONE
```

3. Ensure that the service account section does not have an email that matches the pattern used does not match the pattern `[PROJECT_NUMBER]-compute@developer.gserviceaccount.com`.

Exception:
VMs created by GKE should be excluded. These VMs have names that start with `gke-` and are labeled `goog-gke-node`.

Remediation:

From Console:

1. Go to the `VM instances` page by visiting: https://console.cloud.google.com/compute/instances.
2. Click on the instance name to go to its `VM instance details` page.
3. Click `STOP` and then click `EDIT`.
4. Under the section `Service Account`, select a service account other than the default Compute Engine service account. You may first need to create a new service account.
5. Click `Save` and then click `START`.

From Command Line:

1. Stop the instance:

```
gcloud compute instances stop INSTANCE_NAME
```

2. Update the instance:

```
gcloud compute instances set-service-account INSTANCE_NAME --service-
account=SERVICE_ACCOUNT
```

3. Restart the instance:

```
gcloud compute instances start INSTANCE_NAME
```

Default Value:

By default, Compute instances are configured to use the default Compute Engine service account.

References:

1. https://cloud.google.com/compute/docs/access/service-accounts
2. https://cloud.google.com/compute/docs/access/create-enable-service-accounts-for-instances
3. https://cloud.google.com/sdk/gcloud/reference/compute/instances/set-service-account

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	4.7 Limit Access to Script Tools Limit access to scripting tools (such as Microsoft PowerShell and Python) to only administrative or development users with the need to access those capabilities.		●	●
v7	16 Account Monitoring and Control Account Monitoring and Control			

4.2 Ensure that instances are not configured to use the default service account with full access to all Cloud APIs (Automated)

Profile Applicability:

- Level 1

Description:

To support principle of least privileges and prevent potential privilege escalation it is recommended that instances are not assigned to default service account `Compute Engine default service account` with Scope `Allow full access to all Cloud APIs`.

Rationale:

Along with ability to optionally create, manage and use user managed custom service accounts, Google Compute Engine provides default service account `Compute Engine default service account` for an instances to access necessary cloud services. `Project Editor` role is assigned to `Compute Engine default service account` hence, This service account has almost all capabilities over all cloud services except billing. However, when `Compute Engine default service account` assigned to an instance it can operate in 3 scopes.

```
1. Allow default access: Allows only minimum access required to run an
Instance (Least Privileges)



2. Allow full access to all Cloud APIs: Allow full access to all the cloud
APIs/Services (Too much access)



3. Set access for each API: Allows Instance administrator to choose only
those APIs that are needed to perform specific business functionality
expected by instance
```

When an instance is configured with `Compute Engine default service account` with Scope `Allow full access to all Cloud APIs`, based on IAM roles assigned to the user(s) accessing Instance, it may allow user to perform cloud operations/API calls that user is not supposed to perform leading to successful privilege escalation.

Impact:

In order to change service account or scope for an instance, it needs to be stopped.

Audit:

From Console:

1. Go to the `VM instances` page by visiting: https://console.cloud.google.com/compute/instances.
2. Click on each instance name to go to its `VM instance details` page.
3. If the `Default Compute Engine service account` is selected under `Service Account`, ensure that `Cloud API access scopes` is not set to `Allow full access to all Cloud APIs`.

From Command Line:

1. List Instances from project

```
gcloud compute instances list
```

2. Get the details on each instance:

```
gcloud compute instances describe INSTANCE_NAME --zone ZONE
```

3. Ensure that the instance is not configured to allow the `https://www.googleapis.com/auth/cloud-platform` scope for the default Compute Engine service account:

```
serviceAccounts:
- email: [PROJECT_NUMBER]-compute@developer.gserviceaccount.com
  scopes:
  - https://www.googleapis.com/auth/cloud-platform
```

Exception: Instances created by GKE should be excluded. These instances have names that start with "gke-" and are labeled "goog-gke-node"

Remediation:

From Console:

1. Go to the `VM instances` page by visiting: https://console.cloud.google.com/compute/instances.
2. Click on the impacted VM instance.
3. If the instance is not stopped, click the `Stop` button. Wait for the instance to be stopped.
4. Next, click the `Edit` button.
5. Scroll down to the `Service Account` section.
6. Select a different service account or ensure that `Allow full access to all Cloud APIs` is not selected.

7. Click the `Save` button to save your changes and then click `START`.

From Command Line:

1. Stop the instance:

```
gcloud compute instances stop INSTANCE_NAME
```

2. Update the instance:

```
gcloud compute instances set-service-account INSTANCE_NAME --service-
account=SERVICE_ACCOUNT --scopes [SCOPE1, SCOPE2...]
```

3. Restart the instance:

```
gcloud compute instances start INSTANCE_NAME
```

Default Value:

While creating an VM instance, default service account is used with scope `Allow default access`.

References:

1. https://cloud.google.com/compute/docs/access/create-enable-service-accounts-for-instances
2. https://cloud.google.com/compute/docs/access/service-accounts

Additional Information:

- User IAM roles will override service account scope but configuring minimal scope ensures defense in depth
- Non-default service accounts do not offer selection of access scopes like default service account. IAM roles with non-default service accounts should be used to control VM access.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	4.7 Limit Access to Script Tools Limit access to scripting tools (such as Microsoft PowerShell and Python) to only administrative or development users with the need to access those capabilities.		●	●

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

4.3 Ensure "Block Project-wide SSH keys" is enabled for VM instances (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to use Instance specific SSH key(s) instead of using common/shared project-wide SSH key(s) to access Instances.

Rationale:

Project-wide SSH keys are stored in Compute/Project-meta-data. Project wide SSH keys can be used to login into all the instances within project. Using project-wide SSH keys eases the SSH key management but if compromised, poses the security risk which can impact all the instances within project. It is recommended to use Instance specific SSH keys which can limit the attack surface if the SSH keys are compromised.

Impact:

Users already having Project-wide ssh key pairs and using third party SSH clients will lose access to the impacted Instances. For Project users using gcloud or GCP Console based SSH option, no manual key creation and distribution is required and will be handled by GCE (Google Compute Engine) itself. To access Instance using third party SSH clients Instance specific SSH key pairs need to be created and distributed to the required users.

Audit:

From Console:

1. Go to the `VM instances` page by visiting https://console.cloud.google.com/compute/instances. It will list all the instances in your project.
2. For every instance, click on the name of the instance.
3. Under `SSH Keys`, ensure `Block project-wide SSH keys` is selected.

From Command Line:

1. List all instances in a project:

```
gcloud compute instances list
```

2. For every instance, get the instance metadata:

```
gcloud compute instances describe INSTANCE_NAME
```

3. Ensure `key: block-project-ssh-keys` set to `value: 'true'`.

Exception:
Instances created by GKE should be excluded. These instances have names that start with "gke-" and are labeled "goog-gke-node".

Remediation:

From Console:

1. Go to the `VM instances` page by visiting: https://console.cloud.google.com/compute/instances. It will list all the instances in your project.
2. Click on the name of the Impacted instance
3. Click `Edit` in the toolbar
4. Under SSH Keys, go to the `Block project-wide SSH keys` checkbox
5. To block users with project-wide SSH keys from connecting to this instance, select `Block project-wide SSH keys`
6. Click `Save` at the bottom of the page
7. Repeat steps for every impacted Instance

From Command Line:
Block project-wide public SSH keys, set the metadata value to `TRUE`:

```
gcloud compute instances add-metadata INSTANCE_NAME --metadata block-project-
ssh-keys=TRUE
```

Default Value:

By Default `Block Project-wide SSH keys` is not enabled.

References:

1. https://cloud.google.com/compute/docs/instances/adding-removing-ssh-keys

Additional Information:

If OS Login is enabled, SSH keys in instance metadata are ignored, and therefore blocking project-wide SSH keys is not necessary.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

4.4 Ensure oslogin is enabled for a Project (Automated)

Profile Applicability:

- Level 1

Description:

Enabling OS login binds SSH certificates to IAM users and facilitates effective SSH certificate management.

Rationale:

Enabling osLogin ensures that SSH keys used to connect to instances are mapped with IAM users. Revoking access to IAM user will revoke all the SSH keys associated with that particular user. It facilitates centralized and automated SSH key pair management which is useful in handling cases like response to compromised SSH key pairs and/or revocation of external/third-party/Vendor users.

Impact:

Enabling OS Login on project disables metadata-based SSH key configurations on all instances from a project. Disabling OS Login restores SSH keys that you have configured in project or instance meta-data.

Audit:

From Console:

1. Go to the VM compute metadata page by visiting https://console.cloud.google.com/compute/metadata.
2. Ensure that key `enable-oslogin` is present with value set to `TRUE`.
3. Because instances can override project settings, ensure that no instance has custom metadata with key `enable-oslogin` and value `FALSE`.

From Command Line:

1. Ensure that OS login is enabled on the project:

```
gcloud compute project-info describe
```

2. Verify that the section `commonInstanceMetadata` has a key `enable-oslogin` set to value `TRUE`.
3. Ensure that no instance in the project overrides the project setting:

```
gcloud compute instances describe INSTANCE_NAME
```

Remediation:

From Console:

1. Go to the VM compute metadata page by visiting: https://console.cloud.google.com/compute/metadata.
2. Click `Edit`.
3. Add a metadata entry where the key is `enable-oslogin` and the value is `TRUE`.
4. Click `Save` to apply the changes.
5. For every instances that overrides the project setting, go to the `VM Instances` page at https://console.cloud.google.com/compute/instances.
6. Click the name of the instance on which you want to remove the metadata value.
7. At the top of the instance details page, click `Edit` to edit the instance settings.
8. Under `Custom metadata`, remove any entry with key `enable-oslogin` and the value is `FALSE`
9. At the bottom of the instance details page, click `Save` to apply your changes to the instance.

From Command Line:

1. Configure oslogin on the project:

```
gcloud compute project-info add-metadata --metadata enable-oslogin=TRUE
```

2. Remove instance metadata that overrides the project setting.

```
gcloud compute instances remove-metadata INSTANCE_NAME --keys=enable-oslogin
```

Optionally, you can enable two factor authentication fir OS login. For more information, see: https://cloud.google.com/compute/docs/oslogin/setup-two-factor-authentication.

Default Value:

By default, parameter `enable-oslogin` is not set, which is equivalent to setting it to `FALSE`.

References:

1. https://cloud.google.com/compute/docs/instances/managing-instance-access

2. https://cloud.google.com/compute/docs/instances/managing-instance-access#enable_oslogin
3. https://cloud.google.com/sdk/gcloud/reference/compute/instances/remove-metadata
4. https://cloud.google.com/compute/docs/oslogin/setup-two-factor-authentication

Additional Information:

1. In order to use osLogin, instance using Custom Images must have the latest version of the Linux Guest Environment installed. The following image families do not yet support OS Login:

```
Project cos-cloud (Container-Optimized OS) image family cos-stable.


All project coreos-cloud (CoreOS) image families


Project suse-cloud (SLES) image family sles-11


All Windows Server and SQL Server image families
```

2. Project enable-oslogin can be over-ridden by setting enable-oslogin parameter to an instance metadata individually.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

4.5 Ensure 'Enable connecting to serial ports' is not enabled for VM Instance (Automated)

Profile Applicability:

- Level 1

Description:

Interacting with a serial port is often referred to as the serial console, which is similar to using a terminal window, in that input and output is entirely in text mode and there is no graphical interface or mouse support.

If you enable the interactive serial console on an instance, clients can attempt to connect to that instance from any IP address. Therefore interactive serial console support should be disabled.

Rationale:

A virtual machine instance has four virtual serial ports. Interacting with a serial port is similar to using a terminal window, in that input and output is entirely in text mode and there is no graphical interface or mouse support. The instance's operating system, BIOS, and other system-level entities often write output to the serial ports, and can accept input such as commands or answers to prompts. Typically, these system-level entities use the first serial port (port 1) and serial port 1 is often referred to as the serial console.

The interactive serial console does not support IP-based access restrictions such as IP whitelists. If you enable the interactive serial console on an instance, clients can attempt to connect to that instance from any IP address. This allows anybody to connect to that instance if they know the correct SSH key, username, project ID, zone, and instance name.

Therefore interactive serial console support should be disabled.

Audit:

From Console:

1. Login to Google Cloud console
2. Go to Computer Engine
3. Go to VM instances
4. Click on the Specific VM
5. Ensure `Enable connecting to serial ports` below `Remote access` block is unselected.

From Command Line:
Ensure the below command's output shows `null`:

```
gcloud compute instances describe <vmName> --zone=<region> --
format="json(metadata.items[].key,metadata.items[].value)"
```

or `key` and `value` properties from below command's json response are equal to `serial-port-enable` and `0` or `false` respectively.

```
    {
      "metadata": {
        "items": [
          {
           "key": "serial-port-enable",
            "value": "0"
          }
        ]
     }
    }
```

Remediation:

From Console:

1. Login to Google Cloud console
2. Go to Computer Engine
3. Go to VM instances
4. Click on the Specific VM
5. Click `EDIT`
6. Unselect `Enable connecting to serial ports` below `Remote access` block.
7. Click `Save`

From Command Line:
Use the below command to disable

```
gcloud compute instances add-metadata INSTANCE_NAME --zone=ZONE --
metadata=serial-port-enable=false
```

or

```
gcloud compute instances add-metadata INSTANCE_NAME --zone=ZONE --
metadata=serial-port-enable=0
```

Prevention:
You can prevent VMs from having serial port access enable by `Disable VM serial port access` organization policy:

https://console.cloud.google.com/iam-admin/orgpolicies/compute-disableSerialPortAccess.

Default Value:

By default, connecting to serial ports is not enabled.

References:

1. https://cloud.google.com/compute/docs/instances/interacting-with-serial-console

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

4.6 Ensure that IP forwarding is not enabled on Instances (Automated)

Profile Applicability:

- Level 1

Description:

Compute Engine instance cannot forward a packet unless the source IP address of the packet matches the IP address of the instance. Similarly, GCP won't deliver a packet whose destination IP address is different than the IP address of the instance receiving the packet. However, both capabilities are required if you want to use instances to help route packets.

Forwarding of data packets should be disabled to prevent data loss or information disclosure.

Rationale:

Compute Engine instance cannot forward a packet unless the source IP address of the packet matches the IP address of the instance. Similarly, GCP won't deliver a packet whose destination IP address is different than the IP address of the instance receiving the packet. However, both capabilities are required if you want to use instances to help route packets. To enable this source and destination IP check, disable the `canIpForward` field, which allows an instance to send and receive packets with non-matching destination or source IPs.

Impact:

Deleting instance(s) acting as routers/packet forwarders may break the network connectivity.

Audit:

From Console:

1. Go to the `VM Instances` page by visiting: https://pantheon.corp.google.com/compute/instances.
2. For every instance, click on its name to go to the `VM instance details` page.
3. Under the `Network interfaces` section, ensure that `IP forwarding` is set to `Off` for every network interface.

From Command Line:

1. List all instances:

```
gcloud compute instances list --format='table(name,canIpForward)'
```

2. Ensure that `CAN_IP_FORWARD` column in the output of above command does not contain `True` for any VM instance.

Exception:
Instances created by GKE should be excluded because they need to have IP forwarding enabled and cannot be changed. Instances created by GKE have names that start with "gke-".

Remediation:

You only edit the `canIpForward` setting at instance creation time. Therefore, you need to delete the instance and create a new one where `canIpForward` is set to `false`.
From Console:

1. Go to the `VM Instances` page by visiting: https://pantheon.corp.google.com/compute/instances.
2. Select the `VM Instance` you want to remediate.
3. Click the `Delete` button.
4. On the 'VM Instances' page, click `CREATE INSTANCE'.
5. Create a new instance with the desired configuration. By default, the instance is configured to not allow IP forwarding.

From Command Line:

1. Delete the instance:

```
gcloud compute instances delete INSTANCE_NAME
```

2. Create a new instance to replace it, with `IP forwarding` set to `Off`

```
gcloud compute instances create
```

Default Value:

By default, instances are not configured to allow IP forwarding.

References:

1. https://cloud.google.com/vpc/docs/using-routes#canipforward

Additional Information:

You can only set the `canIpForward` field at instance creation time. After an instance is created, the field becomes read-only.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	11.1 Maintain Standard Security Configurations for Network Devices Maintain standard, documented security configuration standards for all authorized network devices.		●	●
v7	11.2 Document Traffic Configuration Rules All configuration rules that allow traffic to flow through network devices should be documented in a configuration management system with a specific business reason for each rule, a specific individual's name responsible for that business need, and an expected duration of the need.		●	●

4.7 Ensure VM disks for critical VMs are encrypted with Customer-Supplied Encryption Keys (CSEK) (Automated)

Profile Applicability:

- Level 2

Description:

Customer-Supplied Encryption Keys (CSEK) are a feature in Google Cloud Storage and Google Compute Engine. If you supply your own encryption keys, Google uses your key to protect the Google-generated keys used to encrypt and decrypt your data. By default, Google Compute Engine encrypts all data at rest. Compute Engine handles and manages this encryption for you without any additional actions on your part. However, if you wanted to control and manage this encryption yourself, you can provide your own encryption keys.

Rationale:

By default, Google Compute Engine encrypts all data at rest. Compute Engine handles and manages this encryption for you without any additional actions on your part. However, if you wanted to control and manage this encryption yourself, you can provide your own encryption keys.

If you provide your own encryption keys, Compute Engine uses your key to protect the Google-generated keys used to encrypt and decrypt your data. Only users who can provide the correct key can use resources protected by a customer-supplied encryption key.

Google does not store your keys on its servers and cannot access your protected data unless you provide the key. This also means that if you forget or lose your key, there is no way for Google to recover the key or to recover any data encrypted with the lost key.

At least business critical VMs should have VM disks encrypted with CSEK.

Impact:

If you lose your encryption key, you will not be able to recover the data.

Audit:

From Console:

1. Go to Compute Engine `Disks` by visiting: https://console.cloud.google.com/compute/disks.
2. Click on the disk for your critical VMs to see its configuration details.
3. Ensure that `Encryption type` is set to `Customer supplied`.

From Command Line:

Ensure `diskEncryptionKey` property in the below command's response is not null, and contains key `sha256` with corresponding value

```
gcloud compute disks describe DISK_NAME --zone ZONE --
format="json(diskEncryptionKey,name)"
```

Remediation:

Currently there is no way to update the encryption of an existing disk. Therefore you should create a new disk with `Encryption` set to `Customer supplied`.

From Console:

1. Go to Compute Engine `Disks` by visiting: https://console.cloud.google.com/compute/disks.
2. Click `CREATE DISK`.
3. Set `Encryption type` to `Customer supplied`,
4. Provide the `Key` in the box.
5. Select `Wrapped key`.
6. Click `Create`.

From Command Line:

In the gcloud compute tool, encrypt a disk using the --csek-key-file flag during instance creation. If you are using an RSA-wrapped key, use the gcloud beta component:

```
gcloud (beta) compute instances create INSTANCE_NAME --csek-key-file
<example-file.json>
```

To encrypt a standalone persistent disk:

```
gcloud (beta) compute disks create DISK_NAME --csek-key-file <example-
file.json>
```

Default Value:

By default, VM disks are encrypted with Google-managed keys. They are not encrypted with Customer-Supplied Encryption Keys.

References:

1. https://cloud.google.com/compute/docs/disks/customer-supplied-encryption#encrypt_a_new_persistent_disk_with_your_own_keys
2. https://cloud.google.com/compute/docs/reference/rest/v1/disks/get
3. https://cloud.google.com/compute/docs/disks/customer-supplied-encryption#key_file

Additional Information:

`Note 1`: When you delete a persistent disk, Google discards the cipher keys, rendering the data irretrievable. This process is irreversible.

`Note 2`: It is up to you to generate and manage your key. You must provide a key that is a 256-bit string encoded in RFC 4648 standard base64 to Compute Engine.

`Note 3`: An example key file looks like this.

```
[

  {

  "uri": "https://www.googleapis.com/compute/v1/projects/myproject/zones/us-
central1-a/disks/example-disk",

  "key": "acXTX3rxrKAFTF0tYVLvydU1riRZTvUNC4g5I11NY-c=",

  "key-type": "raw"

  },

  {

  "uri":
"https://www.googleapis.com/compute/v1/projects/myproject/global/snapshots/my
-private-snapshot",

  "key":
"ieCx/NcW06PcT7Ep1X6LUTc/hLvUDYyzSZPPVCVPTVEohpeHASqC8uw5TzyO9U+Fka9JFHz0mBib
XUInrC/jEk014kCK/NPjYgEMOyssZ4ZINPKxlUh2zn1bV+MCaTICrdmuSBTWlUUiFoDD6PYznLwh8
ZNdaheCeZ8ewEXgFQ8V+sDroLaN3Xs3MDTXQEMMoNUXMCZEIpg9Vtp9x2oeQ5lAbtt7bYAAHf5l+g
JWw3sUfs0/Glw5fpdjT8Uggrr+RMZezGrltJEF293rvTIjWOEB3z5OHyHwQkvdrPDFcTqsLfh+8Hr
8g+mf+7zVPEC8nEbqpdl3GPv3A7AwpFp7MA=="

  "key-type": "rsa-encrypted"
```

```
    }

]
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			

4.8 Ensure Compute instances are launched with Shielded VM enabled (Automated)

Profile Applicability:

- Level 2

Description:

To defend against against advanced threats and ensure that the boot loader and firmware on your VMs are signed and untampered, it is recommended that Compute instances are launched with Shielded VM enabled.

Rationale:

Shielded VMs are virtual machines (VMs) on Google Cloud Platform hardened by a set of security controls that help defend against rootkits and bootkits.

Shielded VM offers verifiable integrity of your Compute Engine VM instances, so you can be confident your instances haven't been compromised by boot- or kernel-level malware or rootkits. Shielded VM's verifiable integrity is achieved through the use of Secure Boot, virtual trusted platform module (vTPM)-enabled Measured Boot, and integrity monitoring.

Shielded VM instances run firmware which is signed and verified using Google's Certificate Authority, ensuring that the instance's firmware is unmodified and establishing the root of trust for Secure Boot.

Integrity monitoring helps you understand and make decisions about the state of your VM instances and the Shielded VM vTPM enables Measured Boot by performing the measurements needed to create a known good boot baseline, called the integrity policy baseline. The integrity policy baseline is used for comparison with measurements from subsequent VM boots to determine if anything has changed.

Secure Boot helps ensure that the system only runs authentic software by verifying the digital signature of all boot components, and halting the boot process if signature verification fails.

Impact:

None.

Audit:

From Console:

1. Go to the `VM instances` page by visiting: https://console.cloud.google.com/compute/instances.
2. Click on the instance name to see its `VM instance details` page.
3. Under the section `Shielded VM`, ensure that `Turn on vTPM` and `Turn on Integrity Monitoring` are enabled.

From Command Line:

1. For each instance in your project, get its metadata:

```
gcloud compute instances describe INSTANCE_NAME
```

2. Ensure that there is a `shieldedInstanceConfig` configuration and that configuration has the `enableIntegrityMonitoring` and `enableVtpm` set to `true`. If the VM is not a Shield VM image, you will not see a shieldedInstanceConfig` in the output.

Remediation:

To be able turn on `Shielded VM` on an instance, your instance must use an image with Shielded VM support.
From Console:

1. Go to the `VM instances` page by visiting: https://console.cloud.google.com/compute/instances.
2. Click on the instance name to see its `VM instance details` page.
3. Click `STOP` to stop the instance.
4. When the instance has stopped, click `EDIT`.
5. In the Shielded VM section, select `Turn on vTPM` and `Turn on Integrity Monitoring`.
6. Optionally, if you do not use any custom or unsigned drivers on the instance, also select `Turn on Secure Boot`.
7. Click the `Save` button to modify the instance and then click `START` to restart it.

From Command Line:
You can only enable Shielded VM options on instances that have Shielded VM support. For a list of Shielded VM public images, run the gcloud compute images list command with the following flags:

```
gcloud compute images list --project gce-uefi-images --no-standard-images
```

1. Stop the instance:

```
gcloud compute instances stop INSTANCE_NAME
```

2. Update the instance:

```
gcloud compute instances update INSTANCE_NAME --shielded-vtpm --shielded-vm-
integrity-monitoring
```

3. Optionally, if you do not use any custom or unsigned drivers on the instance, also turn on secure boot.

```
gcloud compute instances update INSTANCE_NAME --shielded-vm-secure-boot
```

4. Restart the instance:

```
gcloud compute instances start INSTANCE_NAME
```

Prevention:
You can ensure that all new VMs will be created with Shielded VM enabled by setting up an Organization Policy to for `Shielded VM` at https://console.cloud.google.com/iam-admin/orgpolicies/compute-requireShieldedVm. Learn more at: https://cloud.google.com/security/shielded-cloud/shielded-vm#organization-policy-constraint.

Default Value:

By default, Compute Instances do not have Shielded VM enabled.

References:

1. https://cloud.google.com/compute/docs/instances/modifying-shielded-vm
2. https://cloud.google.com/shielded-vm
3. https://cloud.google.com/security/shielded-cloud/shielded-vm#organization-policy-constraint

Additional Information:

If you do use custom or unsigned drivers on the instance, enabling Secure Boot will cause the machine to no longer boot. Turn on Secure Boot only on instances that have been verified to not have any custom drivers installed.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			

4.9 Ensure that Compute instances do not have public IP addresses (Automated)

Profile Applicability:

- Level 2

Description:

Compute instances should not be configured to have external IP addresses.

Rationale:

To reduce your attack surface, Compute instances should not have public IP addresses. Instead, instances should be configured behind load balancers, to minimize the instance's exposure to the internet.

Impact:

Removing the external IP address from your Compute instance may cause some applications to stop working.

Audit:

From Console:

1. Go to the `VM instances` page by visiting: https://console.cloud.google.com/compute/instances.
2. For every VM, ensure that there is no `External IP` configured.

From Command Line:

1. List the instances in your project:

```
gcloud compute instances list
```

2. For every instance, list its configuration:

```
gcloud compute instances describe INSTANCE_NAME --zone=ZONE
```

3. The output should not contain an `accessConfigs` section under `networkInterfaces`. Note that the `natIP` value is present only for instances that are running or for instances that are stoped but have a static IP address. For instances that are stopped

and are configured to have an ephemeral public IP address, the `natIP` field will not be present. Example output:

```
networkInterfaces:
- accessConfigs:
  - kind: compute#accessConfig
    name: External NAT
    networkTier: STANDARD
    type: ONE_TO_ONE_NAT
```

Exception:
Instances created by GKE should be excluded because some of them have external IP addresses and cannot be changed by editing the instance settings. Instances created by GKE should be excluded. These instances have names that start with "gke-" and are labeled "goog-gke-node".

Remediation:

From Console:

1. Go to the `VM instances` page by visiting: https://console.cloud.google.com/compute/instances.
2. Click on the instance name to go the the `Instance detail page`.
3. Click `Edit`.
4. For each Network interface, ensure that `External IP` is set to `None`.
5. Click `Done` and then click `Save`.

From Command Line:

1. Describe the instance properties:

```
gcloud compute instances describe INSTANCE_NAME --zone=ZONE
```

2. Identify the access config name that contains the external IP address. This access config appears in the following format:

```
networkInterfaces:
- accessConfigs:
 - kind: compute#accessConfig
   name: External NAT
   natIP: 130.211.181.55
   type: ONE_TO_ONE_NAT
```

2. Delete the access config.

```
gcloud compute instances delete-access-config INSTANCE_NAME --zone=ZONE --
access-config-name "ACCESS_CONFIG_NAME"
```

In the above example, the `ACCESS_CONFIG_NAME` is `External NAT`. The name of your access config might be different.

Prevention:

You can configure the `Define allowed external IPs for VM instances` Organization Policy to prevent VMs from being configured with public IP addresses. Learn more at: https://console.cloud.google.com/orgpolicies/compute-vmExternalIpAccess

Default Value:

By default, Compute instances have a public IP address.

References:

1. https://cloud.google.com/load-balancing/docs/backend-service#backends_and_external_ip_addresses
2. https://cloud.google.com/compute/docs/instances/connecting-advanced#sshbetweeninstances
3. https://cloud.google.com/compute/docs/instances/connecting-to-instance
4. https://cloud.google.com/compute/docs/ip-addresses/reserve-static-external-ip-address#unassign_ip
5. https://cloud.google.com/resource-manager/docs/organization-policy/org-policy-constraints

Additional Information:

You can connect to Linux VMs that do not have public IP addresses by using Identity-Aware Proxy for TCP forwarding. Learn more at https://cloud.google.com/compute/docs/instances/connecting-advanced#sshbetweeninstances

For Windows VMs, see https://cloud.google.com/compute/docs/instances/connecting-to-instance.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			

4.10 Ensure that App Engine applications enforce HTTPS connections (Manual)

Profile Applicability:

- Level 2

Description:

In order to maintain the highest level of security all connections to an application should be secure by default.

Rationale:

Insecure HTTP connections maybe subject to eavesdropping which can expose sensitive data.

Impact:

All connections to appengine will automatically be redirected to the HTTPS endpoint ensuring that all connections are secured by TLS.

Audit:

Verify that the app.yaml file controlling the application contains a line which enforces secure connections. For example

```
handlers:
- url: /.*
  secure: always
  redirect_http_response_code: 301
  script: auto
```

https://cloud.google.com/appengine/docs/standard/python3/config/appref

Remediation:

Add a line to the app.yaml file controlling the application which enforces secure connections. For example

```
handlers:
- url: /.*
  **secure: always**
  redirect_http_response_code: 301
  script: auto
```

[https://cloud.google.com/appengine/docs/standard/python3/config/appref]

Default Value:

By default both HTTP and HTTP are supported

References:

1. https://cloud.google.com/appengine/docs/standard/python3/config/appref
2. https://cloud.google.com/appengine/docs/flexible/nodejs/configuring-your-app-with-app-yaml

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	18.5 Use Only Standardized and Extensively Reviewed Encryption Algorithms Use only standardized and extensively reviewed encryption algorithms.		●	●

4.11 Ensure that Compute instances have Confidential Computing enabled (Automated)

Profile Applicability:

- Level 2

Description:

Google Cloud encrypts data at-rest and in-transit, but customer data must be decrypted for processing. Confidential Computing is a breakthrough technology which encrypts data in-use—while it is being processed. Confidential Computing environments keep data encrypted in memory and elsewhere outside the central processing unit (CPU).

Confidential VMs leverage the Secure Encrypted Virtualization (SEV) feature of AMD EPYC™ CPUs. Customer data will stay encrypted while it is used, indexed, queried, or trained on. Encryption keys are generated in hardware, per VM, and not exportable. Thanks to built-in hardware optimizations of both performance and security, there is no significant performance penalty to Confidential Computing workloads.

Rationale:

Confidential Computing enables customers' sensitive code and other data encrypted in memory during processing. Google does not have access to the encryption keys. Confidential VM can help alleviate concerns about risk related to either dependency on Google infrastructure or Google insiders' access to customer data in the clear.

Impact:

- Confidential Computing for Compute instances does not support live migration. Unlike regular Compute instances, Confidential VMs experience disruptions during maintenance events like a software or hardware update.
- Additional charges may be incurred when enabling this security feature. See https://cloud.google.com/compute/confidential-vm/pricing for more info.

Audit:

Note: Confidential Computing is currently only supported on N2D machines. To learn more about types of N2D machines, visit https://cloud.google.com/compute/docs/machine-types#n2d_machine_types
From Console:

1. Go to the `VM instances` page by visiting: https://console.cloud.google.com/compute/instances.
2. Click on the instance name to see its VM instance details page.
3. Ensure that `Confidential VM service` is `Enabled`.

From Command Line:

1. For each instance in your project, get its metadata:

```
gcloud compute instances describe INSTANCE_NAME --zone ZONE
```

2. Ensure that `enableConfidentialCompute` is set to `true` for all instances with machine type starting with "n2d-".

```
confidentialInstanceConfig:
  enableConfidentialCompute: true
```

Remediation:

Confidential Computing can only be enabled when an instance is created. You must delete the current instance and create a new one.

From Console:

1. Go to the VM instances page by visiting: https://console.cloud.google.com/compute/instances.
2. Click `CREATE INSTANCE`.
3. Fill out the desired configuration for your instance.
4. Under the `Confidential VM service` section, check the option `Enable the Confidential Computing service on this VM instance`.
5. Click `Create`.

From Command Line:

Create a new instance with Confidential Compute enabled.

```
gcloud beta compute instances create INSTANCE_NAME   --zone ZONE   --
confidential-compute  --maintenance-policy=TERMINATE
```

Default Value:

By default, Confidential Computing is disabled for Compute instances.

References:

1. https://cloud.google.com/compute/confidential-vm/docs/creating-cvm-instance
2. https://cloud.google.com/compute/confidential-vm/docs/about-cvm
3. https://cloud.google.com/confidential-computing

4. https://cloud.google.com/blog/products/identity-security/introducing-google-cloud-confidential-computing-with-confidential-vms

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			

5 Storage

This section covers recommendations addressing storage on Google Cloud Platform.

5.1 Ensure that Cloud Storage bucket is not anonymously or publicly accessible (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended that IAM policy on Cloud Storage bucket does not allows anonymous or public access.

Rationale:

Allowing anonymous or public access grants permissions to anyone to access bucket content. Such access might not be desired if you are storing any sensitive data. Hence, ensure that anonymous or public access to a bucket is not allowed.

Impact:

No storage buckets would be publicly accessible. You would have to explicitly administer bucket access.

Audit:

From Console:

1. Go to `Storage browser` by visiting https://console.cloud.google.com/storage/browser.
2. Click on each bucket name to go to its `Bucket details` page.
3. Click on the `Permissions` tab.
4. Ensure that `allUsers` and `allAuthenticatedUsers` are not in the `Members` list.

From Command Line:

1. List all buckets in a project

```
gsutil ls
```

2. Check the IAM Policy for each bucket:

```
gsutil iam get gs://BUCKET_NAME
```

No role should contain `allUsers` and/or `allAuthenticatedUsers` as a member.
Using Rest API

1. List all buckets in a project

```
Get https://www.googleapis.com/storage/v1/b?project=<ProjectName>
```

2. Check the IAM Policy for each bucket

```
GET https://www.googleapis.com/storage/v1/b/<bucketName>/iam
```

No role should contain `allUsers` and/or `allAuthenticatedUsers` as a member.

Remediation:

From Console:

1. Go to `Storage browser` by visiting https://console.cloud.google.com/storage/browser.
2. Click on the bucket name to go to its `Bucket details` page.
3. Click on the `Permissions` tab.
4. Click `Delete` button in front of `allUsers` and `allAuthenticatedUsers` to remove that particular role assignment.

From Command Line:
Remove `allUsers` and `allAuthenticatedUsers` access.

```
gsutil iam ch -d allUsers gs://BUCKET_NAME
gsutil iam ch -d allAuthenticatedUsers gs://BUCKET_NAME
```

Prevention:
You can prevent Storage buckets from becoming publicly accessible by setting up the `Domain restricted sharing` organization policy at: https://console.cloud.google.com/iam-admin/orgpolicies/iam-allowedPolicyMemberDomains .

Default Value:

By Default, Storage buckets are not publicly shared.

References:

1. https://cloud.google.com/storage/docs/access-control/iam-reference
2. https://cloud.google.com/storage/docs/access-control/making-data-public
3. https://cloud.google.com/storage/docs/gsutil/commands/iam

Additional Information:

To implement Access restrictions on buckets, configuring Bucket IAM is preferred way than configuring Bucket ACL. On GCP console, "Edit Permissions" for bucket exposes IAM configurations only. Bucket ACLs are configured automatically as per need in order to implement/support User enforced Bucket IAM policy. In-case administrator changes bucket ACL using command-line(gsutils)/API bucket IAM also gets updated automatically.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	12.4 Deny Communication over Unauthorized Ports Deny communication over unauthorized TCP or UDP ports or application traffic to ensure that only authorized protocols are allowed to cross the network boundary in or out of the network at each of the organization's network boundaries.	●	●	●
v7	16 Account Monitoring and Control Account Monitoring and Control			

5.2 Ensure that Cloud Storage buckets have uniform bucket-level access enabled (Automated)

Profile Applicability:

- Level 2

Description:

It is recommended that uniform bucket-level access is enabled on Cloud Storage buckets.

Rationale:

It is recommended to use uniform bucket-level access to unify and simplify how you grant access to your Cloud Storage resources.

Cloud Storage offers two systems for granting users permission to access your buckets and objects: Cloud Identity and Access Management (Cloud IAM) and Access Control Lists (ACLs). These systems act in parallel - in order for a user to access a Cloud Storage resource, only one of the systems needs to grant the user permission. Cloud IAM is used throughout Google Cloud and allows you to grant a variety of permissions at the bucket and project levels. ACLs are used only by Cloud Storage and have limited permission options, but they allow you to grant permissions on a per-object basis.

In order to support a uniform permissioning system, Cloud Storage has uniform bucket-level access. Using this feature disables ACLs for all Cloud Storage resources: access to Cloud Storage resources then is granted exclusively through Cloud IAM. Enabling uniform bucket-level access guarantees that if a Storage bucket is not publicly accessible, no object in the bucket is publicly accessible either.

Impact:

If you enable uniform bucket-level access, you revoke access from users who gain their access solely through object ACLs.

Certain Google Cloud services, such as Stackdriver, Cloud Audit Logs, and Datastore, cannot export to Cloud Storage buckets that have uniform bucket-level access enabled.

Audit:

From Console:

1. Open the Cloud Storage browser in the Google Cloud Console by visiting: https://console.cloud.google.com/storage/browser

2. For each bucket, make sure that `Access control` column has the value `Uniform`.

From Command Line:

1. List all buckets in a project

```
gsutil ls
```

2. For each bucket, verify that uniform bucket-level access is enabled.

```
gsutil uniformbucketlevelaccess get gs://BUCKET_NAME/
```

If uniform bucket-level access is enabled, the response looks like:

```
Uniform bucket-level access setting for gs://BUCKET_NAME/:
    Enabled: True
    LockedTime: LOCK_DATE
```

Remediation:

From Console:

1. Open the Cloud Storage browser in the Google Cloud Console by visiting: https://console.cloud.google.com/storage/browser
2. In the list of buckets, click on the name of the desired bucket.
3. Select the `Permissions` tab near the top of the page.
4. In the text box that starts with `This bucket uses fine-grained access control...`, click `Edit`.
5. In the pop-up menu that appears, select `Uniform`.
6. Click `Save`.

From Command Line:
Use the on option in a uniformbucketlevelaccess set command:

```
gsutil uniformbucketlevelaccess set on gs://BUCKET_NAME/
```

Prevention
You can set up an Organization Policy to enforce that any new bucket has uniform bucket level access enabled. Learn more at: https://cloud.google.com/storage/docs/setting-org-policies#uniform-bucket

Default Value:

By default, Cloud Storage buckets do not have uniform bucket-level access enabled.

References:

1. https://cloud.google.com/storage/docs/uniform-bucket-level-access
2. https://cloud.google.com/storage/docs/using-uniform-bucket-level-access
3. https://cloud.google.com/storage/docs/setting-org-policies#uniform-bucket

Additional Information:

Uniform bucket-level access can no longer be disabled if it has been active on a bucket for 90 consecutive days.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

6 Cloud SQL Database Services

This section covers security recommendations to follow to secure Cloud SQL database services.

The recommendations in this section on setting up database flags are also present in the CIS Oracle MySQL Community Server 5.7 Benchmarks and in the CIS PostgreSQL 12 Benchmarks. We, nevertheless, include them here as well, the remediation instructions are different on Cloud SQL. Settings these flags require superuser privileges and can only be configured through GCP controls.

Learn more at: https://cloud.google.com/sql/docs/postgres/users and https://cloud.google.com/sql/docs/mysql/flags.

6.1 MySQL Database

This section covers recommendations addressing Cloud SQL for MySQL on Google Cloud Platform.

6.1.1 Ensure that a MySQL database instance does not allow anyone to connect with administrative privileges (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to set a password for the administrative user (`root` by default) to prevent unauthorized access to the SQL database instances.

This recommendation is applicable only for MySQL Instances. PostgreSQL does not offer any setting for No Password from the cloud console.

Rationale:

At the time of MySQL Instance creation, not providing an administrative password allows anyone to connect to the SQL database instance with administrative privileges. The root password should be set to ensure only authorized users have these privileges.

Impact:

Connection strings for administrative clients need to be reconfigured to use a password.

Audit:

From Command Line:

1. List All SQL database instances of type MySQL.

```
gcloud sql instances list --filter='DATABASE_VERSION:MYSQL*'
```

2. For every MySQL instance try to connect from an `authorized network`:

```
mysql -u root -h <Instance_IP>
```

The command should return either an error message or a password prompt.
Sample Error message:

```
ERROR 1045 (28000): Access denied for user 'root'@'[Inatance_IP]' (using
password: NO)
```

If a command produces the `mysql prompt`, the SQL instance allows anyone to connect with administrative privileges without needing a password.

Note: The `No Password` setting is exposed only at the time of MySQL instance creation. Once the instance is created, the Google Cloud Platform Console does not expose the set to confirm whether a password for an administrative user is set to a MySQL instance.

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Platform Console using `https://console.cloud.google.com/sql/`
2. Select the instance to open its Overview page.
3. Select `Access Control > Users`.
4. Click the `more actions icon` for the user to be updated.
5. Select `Change password`, specify a `new password`, and click `OK`.

From Command Line:
Set a password to a MySql instance:

```
gcloud sql users set-password [USER_NAME] [HOST] --instance=[INSTANCE_NAME] -
-password=[PASSWORD]
```

Default Value:

From the Google Cloud Platform Console, the `Create Instance` workflow enforces the rule to enter the root password unless the option `No Password` is selected explicitly.

References:

1. https://cloud.google.com/sql/docs/mysql/create-manage-users
2. https://cloud.google.com/sql/docs/mysql/create-instance

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	4.2 Change Default Passwords Before deploying any new asset, change all default passwords to have values consistent with administrative level accounts.	●	●	●

6.1.2 Ensure 'skip_show_database' database flag for Cloud SQL Mysql instance is set to 'on' (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to set `skip_show_database` database flag for Cloud SQL Mysql instance to `on`

Rationale:

'skip_show_database' database flag prevents people from using the SHOW DATABASES statement if they do not have the SHOW DATABASES privilege. This can improve security if you have concerns about users being able to see databases belonging to other users. Its effect depends on the SHOW DATABASES privilege: If the variable value is ON, the SHOW DATABASES statement is permitted only to users who have the SHOW DATABASES privilege, and the statement displays all database names. If the value is OFF, SHOW DATABASES is permitted to all users, but displays the names of only those databases for which the user has the SHOW DATABASES or other privilege. This recommendation is applicable to Mysql database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Ensure the database flag `skip_show_database` that has been set is listed under the `Database flags` section.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Ensure the below command returns `on` for every Cloud SQL Mysql database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="skip_show_database")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the Mysql instance for which you want to enable to database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `skip_show_database` from the drop-down menu, and set its value to `on`.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `skip_show_database` database flag for every Cloud SQL Mysql database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags
skip_show_database=on

Note :

This command will overwrite all database flags previously set. To keep those
and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

References:

1. https://cloud.google.com/sql/docs/mysql/flags
2. https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_skip_show_database

Additional Information:

```
"WARNING: This patch modifies database flag values, which may require

your instance to be restarted. Check the list of supported flags -
```

```
https://cloud.google.com/sql/docs/mysql/flags - to see if your

instance will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability, and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines."



Note: Configuring the above flag restarts the Cloud SQL instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

6.1.3 Ensure that the 'local_infile' database flag for a Cloud SQL Mysql instance is set to 'off' (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to set the `local_infile` database flag for a Cloud SQL MySQL instance to `off`.

Rationale:

The `local_infile` flag controls the server-side LOCAL capability for LOAD DATA statements. Depending on the `local_infile` setting, the server refuses or permits local data loading by clients that have LOCAL enabled on the client side.

To explicitly cause the server to refuse LOAD DATA LOCAL statements (regardless of how client programs and libraries are configured at build time or runtime), start mysqld with local_infile disabled. local_infile can also be set at runtime.

Due to security issues associated with the `local_infile` flag, it is recommended to disable it. This recommendation is applicable to MySQL database instances.

Impact:

Disabling `local_infile` makes the server refuse local data loading by clients that have LOCAL enabled on the client side.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Ensure the database flag `local_infile` that has been set is listed under the `Database flags` section.

From Command Line:

1. List all Cloud SQL database instances:

```
gcloud sql instances list
```

2. Ensure the below command returns `off` for every Cloud SQL MySQL database instance.

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="local_infile")|.value'
```

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the MySQL instance where the database flag needs to be enabled.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `local_infile` from the drop-down menu, and set its value to `off`.
6. Click `Save`.
7. Confirm the changes under `Flags` on the Overview page.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Configure the `local_infile` database flag for every Cloud SQL Mysql database instance using the below command:

```
gcloud sql instances patch INSTANCE_NAME --database-flags local_infile=off

Note :

This command will overwrite all database flags that were previously set. To
keep those and add new ones, include the values for all flags to be set on
the instance; any flag not specifically included is set to its default value.
For flags that do not take a value, specify the flag name followed by an
equals sign ("=").
```

Default Value:

By default `local_infile` is `on`.

References:

1. https://cloud.google.com/sql/docs/mysql/flags
2. https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_local_infile
3. https://dev.mysql.com/doc/refman/5.7/en/load-data-local.html

Additional Information:

```
"WARNING: This patch modifies database flag values, which may require

the instance to be restarted. Check the list of supported flags -

https://cloud.google.com/sql/docs/mysql/flags - to see if your instance will
be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability, and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines."



Note: Configuring the above flag restarts the Cloud SQL instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			

6.2 PostgreSQL Database

This section covers recommendations addressing Cloud SQL for PostgreSQL on Google Cloud Platform.

6.2.1 Ensure that the 'log_checkpoints' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Automated)

Profile Applicability:

- Level 1

Description:

Ensure that the `log_checkpoints` database flag for the Cloud SQL PostgreSQL instance is set to `on`.

Rationale:

Enabling `log_checkpoints` causes checkpoints and restart points to be logged in the server log. Some statistics are included in the log messages, including the number of buffers written and the time spent writing them. This parameter can only be set in the postgresql.conf file or on the server command line. This recommendation is applicable to PostgreSQL database instances.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page.
3. Ensure that the database flag `log_checkpoints` that has been set is listed under the `Database flags` section.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Ensure that the below command returns `on` for every Cloud SQL PostgreSQL database instance.

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_checkpoints")|.value'
```

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance where the database flag needs to be enabled.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_checkpoints` from the drop-down menu, and set its value.
6. Click `Save`.
7. Confirm the changes under `Flags` on the Overview page.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Configure the `log_checkpoints` database flag for every Cloud SQL PosgreSQL database instance using the below command:

```
gcloud sql instances patch INSTANCE_NAME --database-flags log_checkpoints=on

Note:

This command will overwrite all previously set database flags. To keep those
and add new ones, include the values for all flags to be set on the instance.
Any flag not specifically included is set to its default value. For flags
that do not take a value, specify the flag name followed by an equals sign
("=").
```

Default Value:

By default `log_checkpoints` is `off`.

References:

1. https://www.postgresql.org/docs/9.6/runtime-config-logging.html#RUNTIME-CONFIG-LOGGING-WHAT
2. https://cloud.google.com/sql/docs/postgres/flags#setting_a_database_flag

Additional Information:

```
WARNING: This patch modifies database flag values, which may require

your instance to be restarted. Check the list of supported flags -
```

```
https://cloud.google.com/sql/docs/postgres/flags - to see if your

instance will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability, and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag restarts the Cloud SQL instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.2 Ensure 'log_error_verbosity' database flag for Cloud SQL PostgreSQL instance is set to 'DEFAULT' or stricter (Manual)

Profile Applicability:

- Level 2

Description:

The `log_error_verbosity` flag controls the verbosity/details of messages logged. Valid values are:

- `TERSE`
- `DEFAULT`
- `VERBOSE`

`TERSE` excludes the logging of `DETAIL`, `HINT`, `QUERY`, and `CONTEXT` error information.

`VERBOSE` output includes the `SQLSTATE` error code, source code file name, function name, and line number that generated the error.

Ensure an appropriate value is set to 'DEFAULT' or stricter.

Rationale:

Auditing helps in troubleshooting operational problems and also permits forensic analysis. If `log_error_verbosity` is not set to the correct value, too many details or too few details may be logged. This flag should be configured with a value of 'DEFAULT' or stricter. This recommendation is applicable to PostgreSQL database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Go to `Configuration` card
4. Under `Database flags`, check the value of `log_error_verbosity` flag is set to 'DEFAULT' or stricter.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Use the below command for every Cloud SQL PostgreSQL database instance to verify the value of `log_error_verbosity`

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_error_verbosity")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance for which you want to enable the database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_error_verbosity` from the drop-down menu and set appropriate value.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the log_error_verbosity database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags
log_error_verbosity=<TERSE|DEFAULT|VERBOSE>

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_error_verbosity` is `DEFAULT`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags

2. https://www.postgresql.org/docs/current/runtime-config-logging.html#RUNTIME-CONFIG-LOGGING-WHAT

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.3 Ensure that the 'log_connections' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Automated)

Profile Applicability:

- Level 1

Description:

Enabling the `log_connections` setting causes each attempted connection to the server to be logged, along with successful completion of client authentication. This parameter cannot be changed after the session starts.

Rationale:

PostgreSQL does not log attempted connections by default. Enabling the `log_connections` setting will create log entries for each attempted connection as well as successful completion of client authentication which can be useful in troubleshooting issues and to determine any unusual connection attempts to the server. This recommendation is applicable to PostgreSQL database instances.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page.
3. Go to the `Configuration` card.
4. Under `Database flags`, check the value of `log_connections` flag to determine if it is configured as expected.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Ensure the below command returns `on` for every Cloud SQL PostgreSQL database instance:

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_connections")|.value'
```

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance for which you want to enable the database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_connections` from the drop-down menu and set the value as `on`.
6. Click `Save`.
7. Confirm the changes under `Flags` on the Overview page.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Configure the `log_connections` database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags log_connections=on

Note:
This command will overwrite all previously set database flags. To keep those
and add new ones, include the values for all flags to be set on the instance;
any flag not specifically included is set to its default value. For flags
that do not take a value, specify the flag name followed by an equals sign
("=").
```

Default Value:

By default `log_connections` is `off`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/9.6/runtime-config-logging.html#RUNTIME-CONFIG-LOGGING-WHAT

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.
```

```
Note: some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see the Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.4 Ensure that the 'log_disconnections' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Automated)

Profile Applicability:

- Level 1

Description:

Enabling the `log_disconnections` setting logs the end of each session, including the session duration.

Rationale:

PostgreSQL does not log session details such as duration and session end by default. Enabling the `log_disconnections` setting will create log entries at the end of each session which can be useful in troubleshooting issues and determine any unusual activity across a time period. The `log_disconnections` and `log_connections` work hand in hand and generally, the pair would be enabled/disabled together. This recommendation is applicable to PostgreSQL database instances.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Go to the `Configuration` card.
4. Under `Database flags`, check the value of `log_disconnections` flag is configured as expected.

From Command Line:

1. List all Cloud SQL database Instances using the following command:

```
gcloud sql instances list
```

2. Ensure the below command returns `on` for every Cloud SQL PostgreSQL database instance:

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_disconnections")|.value'
```

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance where the database flag needs to be enabled.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_disconnections` from the drop-down menu and set the value as `on`.
6. Click `Save`.
7. Confirm the changes under `Flags` on the Overview page.

From Command Line:

1. List all Cloud SQL database Instances using the following command:

```
gcloud sql instances list
```

2. Configure the `log_disconnections` database flag for every Cloud SQL PosgreSQL database instance using the below command:

```
gcloud sql instances patch INSTANCE_NAME --database-flags
log_disconnections=on

Note: This command will overwrite all previously setdatabase flags. To keep
those and add new ones, include the values for all flags to be set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_disconnections` is off.

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/9.6/runtime-config-logging.html#RUNTIME-CONFIG-LOGGING-WHAT

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.
```

```
Note: some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.5 Ensure 'log_duration' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Manual)

Profile Applicability:

- Level 1

Description:

Enabling the `log_duration` setting causes the duration of each completed statement to be logged. This does not logs the text of the query and thus behaves different from the `log_min_duration_statement` flag. This parameter cannot be changed after session start.

Rationale:

Monitoring the time taken to execute the queries can be crucial in identifying any resource hogging queries and assessing the performance of the server. Further steps such as load balancing and use of optimized queries can be taken to ensure the performance and stability of the server. This recommendation is applicable to PostgreSQL database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its Instance Overview page
3. Go to `Configuration` card
4. Under `Database flags`, check the value of `log_duration` flag is configured as expected.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Ensure the below command returns `on` for every Cloud SQL PostgreSQL database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_duration")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance for which you want to enable the database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_duration` from the drop-down menu and set the value as `on`.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `log_duration` database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags log_duration=on

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_duration` is `off`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/9.6/runtime-config-logging.html#GUC-LOG-MIN-DURATION-STATEMENT

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.
```

Note: some database flag settings can affect instance availability or stability and remove the instance from the Cloud SQL SLA. For information about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL instance.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.6 Ensure that the 'log_lock_waits' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Automated)

Profile Applicability:

- Level 1

Description:

Enabling the `log_lock_waits` flag for a PostgreSQL instance creates a log for any session waits that take longer than the alloted `deadlock_timeout` time to acquire a lock.

Rationale:

The deadlock timeout defines the time to wait on a lock before checking for any conditions. Frequent run overs on deadlock timeout can be an indication of an underlying issue. Logging such waits on locks by enabling the `log_lock_waits` flag can be used to identify poor performance due to locking delays or if a specially-crafted SQL is attempting to starve resources through holding locks for excessive amounts of time. This recommendation is applicable to PostgreSQL database instances.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its Instance Overview page.
3. Go to the `Configuration` card.
4. Under `Database flags`, check if the value of the `log_lock_waits` flag is configured as expected.

From Command Line:

1. List all Cloud SQL database Instances using the following command:

```
gcloud sql instances list
```

2. Ensure the below command returns `on` for every Cloud SQL PostgreSQL database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_lock_waits")|.value'
```

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance where the database flag needs to be enabled.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_lock_waits` from the drop-down menu and set the value as `on`.
6. Click `Save`.
7. Confirm the changes under `Flags` on the Overview page.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Configure the `log_lock_waits` database flag for every Cloud SQL PosgreSQL database instance using the below command:

```
gcloud sql instances patch INSTANCE_NAME --database-flags log_lock_waits=on

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags to be set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_lock_waits` is `off`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/9.6/runtime-config-logging.html#GUC-LOG-MIN-DURATION-STATEMENT

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.
```

```
Note: Some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see the Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.7 Ensure 'log_statement' database flag for Cloud SQL PostgreSQL instance is set appropriately (Manual)

Profile Applicability:

- Level 1

Description:

The value of `log_statement` flag determined the SQL statements that are logged. Valid values are:

- `none`
- `ddl`
- `mod`
- `all`

The value `ddl` logs all data definition statements. The value `mod` logs all ddl statements, plus data-modifying statements.

The statements are logged after a basic parsing is done and statement type is determined, thus this does not logs statements with errors. When using extended query protocol, logging occurs after an Execute message is received and values of the Bind parameters are included.

A value of 'ddl' is recommended unless otherwise directed by your organization's logging policy.

Rationale:

Auditing helps in troubleshooting operational problems and also permits forensic analysis. If `log_statement` is not set to the correct value, too many statements may be logged leading to issues in finding the relevant information from the logs, or too few statements may be logged with relevant information missing from the logs. Setting log_statement to align with your organization's security and logging policies facilitates later auditing and review of database activities. This recommendation is applicable to PostgreSQL database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page

3. Go to `Configuration` card
4. Under `Database flags`, check the value of `log_statement` flag is set to appropriately.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Use the below command for every Cloud SQL PostgreSQL database instance to verify the value of `log_statement`

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_statement")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance for which you want to enable the database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_statement` from the drop-down menu and set appropriate value.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `log_statement` database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags
log_statement=<ddl|mod|all|none>

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/current/runtime-config-logging.html#RUNTIME-CONFIG-LOGGING-WHAT

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.8 Ensure 'log_hostname' database flag for Cloud SQL PostgreSQL instance is set appropriately (Automated)

Profile Applicability:

- Level 1

Description:

PostgreSQL logs only the IP address of the connecting hosts. The `log_hostname` flag controls the logging of `hostnames` in addition to the IP addresses logged. The performance hit is dependent on the configuration of the environment and the host name resolution setup. This parameter can only be set in the `postgresql.conf` file or on the server command line.

Rationale:

Logging hostnames can incur overhead on server performance as for each statement logged, DNS resolution will be required to convert IP address to hostname. Depending on the setup, this may be non-negligible. Additionally, the IP addresses that are logged can be resolved to their DNS names later when reviewing the logs excluding the cases where dynamic hostnames are used. This recommendation is applicable to PostgreSQL database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Go to `Configuration` card
4. Under `Database flags`, check the value of `log_hostname` flag is set to appropriately.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Use the below command for every Cloud SQL PostgreSQL database instance to verify the value of `log_hostname`

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_hostname")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance for which you want to enable the database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_hostname` from the drop-down menu and set appropriate value.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `log_hostname` database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags
log_hostname=<off|on>

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_hostname` is `off`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/current/runtime-config-logging.html#RUNTIME-CONFIG-LOGGING-WHAT

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.9 Ensure 'log_parser_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Automated)

Profile Applicability:

- Level 2

Description:

The PostgreSQL planner/optimizer is responsible to parse and verify the syntax of each query received by the server. If the syntax is correct a `parse tree` is built up else an error is generated. The `log_parser_stats` flag controls the inclusion of parser performance statistics in the PostgreSQL logs for each query.

Rationale:

The `log_parser_stats` flag enables a crude profiling method for logging parser performance statistics which even though can be useful for troubleshooting, it may increase the amount of logs significantly and have performance overhead. This recommendation is applicable to PostgreSQL database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Go to `Configuration` card
4. Under `Database flags`, check the value of `log_parser_stats` flag is set to 'off'.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Esure the below command returns `off` for every Cloud SQL PostgreSQL database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_parser_stats")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance for which you want to enable the database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_parser_stats` from the drop-down menu and set appropriate value.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `log_parser_stats` database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags
log_parser_stats=off

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_parser_stats` is `off`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/current/runtime-config-logging.html#RUNTIME-CONFIG-LOGGING-WHAT
3. https://www.postgresql.org/docs/10/parser-stage.html

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
```

```
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.10 Ensure 'log_planner_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Automated)

Profile Applicability:

- Level 2

Description:

The same SQL query can be excuted in multiple ways and still produce different results. The PostgreSQL planner/optimizer is responsible to create an optimal execution plan for each query. The `log_planner_stats` flag controls the inclusion of PostgreSQL planner performance statistics in the PostgreSQL logs for each query.

Rationale:

The `log_planner_stats` flag enables a crude profiling method for logging PostgreSQL planner performance statistics which even though can be useful for troubleshooting, it may increase the amount of logs significantly and have performance overhead. This recommendation is applicable to PostgreSQL database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Go to `Configuration` card
4. Under `Database flags`, check the value of `log_planner_stats` flag is set to 'off'.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Ensure the below command returns `off` for every Cloud SQL PostgreSQL database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_planner_stats")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance for which you want to enable the database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_planner_stats` from the drop-down menu and set appropriate value.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `log_planner_stats` database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags
log_planner_stats=off

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_planner_stats` is `off`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/9.6/runtime-config-statistics.html#RUNTIME-CONFIG-STATISTICS-MONITOR
3. https://www.postgresql.org/docs/9.5/planner-optimizer.html

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
```

```
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.11 Ensure 'log_executor_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Automated)

Profile Applicability:

- Level 2

Description:

The PostgreSQL executor is responsible to execute the plan handed over by the PostgreSQL planner. The executor processes the plan recursively to extract the required set of rows. The `log_executor_stats` flag controls the inclusion of PostgreSQL executor performance statistics in the PostgreSQL logs for each query.

Rationale:

The `log_executor_stats` flag enables a crude profiling method for logging PostgreSQL executor performance statistics which even though can be useful for troubleshooting, it may increase the amount of logs significantly and have performance overhead. This recommendation is applicable to PostgreSQL database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Go to `Configuration` card
4. Under `Database flags`, check the value of `log_executor_stats` flag is set to 'off'.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Esure the below command returns `off` for every Cloud SQL PostgreSQL database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_executor_stats")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance for which you want to enable the database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_executor_stats` from the drop-down menu and set appropriate value.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `log_executor_stats` database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags
log_executor_stats=off

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_executor_stats` is `off`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/current/runtime-config-logging.html#RUNTIME-CONFIG-LOGGING-WHAT
3. https://www.postgresql.org/docs/8.2/executor.html

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
```

```
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.12 Ensure 'log_statement_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Automated)

Profile Applicability:

- Level 2

Description:

The `log_statement_stats` flag controls the inclusion of end to end performance statistics of a SQL query in the PostgreSQL logs for each query. This **cannot** be enabled with other module statistics (`log_parser_stats`, `log_planner_stats`, `log_executor_stats`).

Rationale:

The `log_statement_stats` flag enables a crude profiling method for logging end to end performance statistics of a SQL query. This can be useful for troubleshooting but may increase the amount of logs significantly and have performance overhead. This recommendation is applicable to PostgreSQL database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Go to `Configuration` card
4. Under `Database flags`, check the value of `log_statement_stats` flag is set to 'off'.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Ensure the below command returns `off` for every Cloud SQL PostgreSQL database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_statement_stats")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance for which you want to enable the database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_statement_stats` from the drop-down menu and set appropriate value.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `log_statement_stats` database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags
log_statement_stats=off

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_statement_stats` is `off`.

References:

1. https://www.postgresql.org/docs/9.6/runtime-config-statistics.html#RUNTIME-CONFIG-STATISTICS-MONITOR
2. https://cloud.google.com/sql/docs/postgres/flags

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.
```

```
Note: some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.13 Ensure that the 'log_min_messages' database flag for Cloud SQL PostgreSQL instance is set appropriately (Manual)

Profile Applicability:

- Level 1

Description:

The `log_min_messages` flag defines the minimum message severity level that is considered as an error statement. Messages for error statements are logged with the SQL statement. Valid values include `DEBUG5`, `DEBUG4`, `DEBUG3`, `DEBUG2`, `DEBUG1`, `INFO`, `NOTICE`, `WARNING`, `ERROR`, `LOG`, `FATAL`, and `PANIC`. Each severity level includes the subsequent levels mentioned above.

Note: To effectively turn off logging failing statements, set this parameter to PANIC.

ERROR is considered the best practice setting. Changes should only be made in accordance with the organization's logging policy.

Rationale:

Auditing helps in troubleshooting operational problems and also permits forensic analysis. If `log_min_error_statement` is not set to the correct value, messages may not be classified as error messages appropriately. Considering general log messages as error messages would make it difficult to find actual errors, while considering only stricter severity levels as error messages may skip actual errors to log their SQL statements. The `log_min_messages` flag should be set in accordance with the organization's logging policy. This recommendation is applicable to PostgreSQL database instances.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page.
3. Go to the `Configuration` card.
4. Under `Database flags`, check the value of `log_min_messages` flag is in accordance with the organization's logging policy.

From Command Line:

1. List all Cloud SQL database Instances using the following command:

```
gcloud sql instances list
```

2. Use the below command for every Cloud SQL PostgreSQL database instance to verify that the value of `log_min_messages` is in accordance with the organization's logging policy.

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_min_messages")|.value'
```

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance for which you want to enable the database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_min_messages` from the drop-down menu and set appropriate value.
6. Click `Save` to save the changes.
7. Confirm the changes under `Flags` on the Overview page.

From Command Line:

1. List all Cloud SQL database Instances using the following command:

```
gcloud sql instances list
```

2. Configure the `log_min_messages` database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags
log_min_messages=<DEBUG5|DEBUG4|DEBUG3|DEBUG2|DEBUG1|INFO|NOTICE|WARNING|ERRO
R|LOG|FATAL|PANIC>

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags to be set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_min_error_statement` is `ERROR`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/9.6/runtime-config-logging.html#RUNTIME-CONFIG-LOGGING-WHEN

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.

Note: Some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.14 Ensure 'log_min_error_statement' database flag for Cloud SQL PostgreSQL instance is set to 'Error' or stricter (Automated)

Profile Applicability:

- Level 1

Description:

The `log_min_error_statement` flag defines the minimum message severity level that are considered as an error statement. Messages for error statements are logged with the SQL statement. Valid values include `DEBUG5`, `DEBUG4`, `DEBUG3`, `DEBUG2`, `DEBUG1`, `INFO`, `NOTICE`, `WARNING`, `ERROR`, `LOG`, `FATAL`, and `PANIC`. Each severity level includes the subsequent levels mentioned above. Ensure a value of `ERROR` or stricter is set.

Rationale:

Auditing helps in troubleshooting operational problems and also permits forensic analysis. If `log_min_error_statement` is not set to the correct value, messages may not be classified as error messages appropriately. Considering general log messages as error messages would make is difficult to find actual errors and considering only stricter severity levels as error messages may skip actual errors to log their SQL statements. The `log_min_error_statement` flag should be set to `ERROR` or stricter. This recommendation is applicable to PostgreSQL database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Go to `Configuration` card
4. Under `Database flags`, check the value of `log_min_error_statement` flag is configured as to `ERROR` or stricter.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Use the below command for every Cloud SQL PostgreSQL database instance to verify the value of `log_min_error_statement` is set to `ERROR` or stricter.

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_min_error_statement")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance for which you want to enable the database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_min_error_statement` from the drop-down menu and set appropriate value.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `log_min_error_statement` database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags
log_min_error_statement=<DEBUG5|DEBUG4|DEBUG3|DEBUG2|DEBUG1|INFO|NOTICE|WARNI
NG|ERROR>

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_min_error_statement` is `ERROR`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags

2. https://www.postgresql.org/docs/9.6/runtime-config-logging.html#RUNTIME-CONFIG-LOGGING-WHEN

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.15 Ensure that the 'log_temp_files' database flag for Cloud SQL PostgreSQL instance is set to '0' (on) (Automated)

Profile Applicability:

- Level 1

Description:

PostgreSQL can create a temporary file for actions such as sorting, hashing and temporary query results when these operations exceed `work_mem`. The `log_temp_files` flag controls logging names and the file size when it is deleted. Configuring `log_temp_files` to `0` causes all temporary file information to be logged, while positive values log only files whose size is greater than or equal to the specified number of kilobytes. A value of `-1` disables temporary file information logging.

Rationale:

If all temporary files are not logged, it may be more difficult to identify potential performance issues that may be due to either poor application coding or deliberate resource starvation attempts.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Go to the `Configuration` card.
4. Under `Database flags`, check that the value of `log_temp_files` flag is set to `0`.

From Command Line:

1. List all Cloud SQL database Instances using the following command:

```
gcloud sql instances list
```

2. Ensure that the below command returns `0` for every Cloud SQL PostgreSQL database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="log_temp_files")|.value'
```

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance where the database flag needs to be enabled.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_temp_files` from the drop-down menu and set the value as `0`.
6. Click `Save`.
7. Confirm the changes under `Flags` on the Overview page.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Configure the `log_temp_files` database flag for every Cloud SQL PosgreSQL database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags log_temp_files=`0`

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags to be set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_temp_files` is `-1`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/9.6/runtime-config-logging.html#GUC-LOG-TEMP-FILES

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.
```

```
Note: some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.2.16 Ensure that the 'log_min_duration_statement' database flag for Cloud SQL PostgreSQL instance is set to '-1' (disabled) (Automated)

Profile Applicability:

- Level 1

Description:

The `log_min_duration_statement` flag defines the minimum amount of execution time of a statement in milliseconds where the total duration of the statement is logged. Ensure that `log_min_duration_statement` is disabled, i.e., a value of `-1` is set.

Rationale:

Logging SQL statements may include sensitive information that should not be recorded in logs. This recommendation is applicable to PostgreSQL database instances.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page.
3. Go to the `Configuration` card.
4. Under `Database flags`, check that the value of `log_min_duration_statement` flag is set to `-1`.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Use the below command for every Cloud SQL PostgreSQL database instance to verify the value of `log_min_duration_statement` is set to `-1`.

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] |
select(.name=="log_min_duration_statement")|.value'
```

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the PostgreSQL instance where the database flag needs to be enabled.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `log_min_duration_statement` from the drop-down menu and set a value of `-1`.
6. Click `Save`.
7. Confirm the changes under `Flags` on the Overview page.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Configure the `log_min_duration_statement` flag for every Cloud SQL PosgreSQL database instance using the below command:

```
gcloud sql instances patch INSTANCE_NAME --database-flags
log_min_duration_statement=-1

Note: This command will overwrite all database flags previously set. To keep
those and add new ones, include the values for all flags to be set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `log_min_duration_statement` is `-1`.

References:

1. https://cloud.google.com/sql/docs/postgres/flags
2. https://www.postgresql.org/docs/current/runtime-config-logging.html#RUNTIME-CONFIG-LOGGING-WHAT

Additional Information:

```
WARNING: This patch modifies database flag values, which may require your
instance to be restarted. Check the list of supported flags -
https://cloud.google.com/sql/docs/postgres/flags - to see if your instance
will be restarted when this patch is submitted.

Note: Some database flag settings can affect instance availability or
stability and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.
```

```
Note: Configuring the above flag does not require restarting the Cloud SQL
instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.3 SQL Server

This section covers recommendations addressing Cloud SQL for SQL Server on Google Cloud Platform.

6.3.1 Ensure 'external scripts enabled' database flag for Cloud SQL SQL Server instance is set to 'off' (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to set `external scripts enabled` database flag for Cloud SQL SQL Server instance to `off`

Rationale:

`external scripts enabled` enable the execution of scripts with certain remote language extensions. This property is OFF by default. When Advanced Analytics Services is installed, setup can optionally set this property to true. As the External Scripts Enabled feature allows scripts external to SQL such as files located in an R library to be executed, which could adversely affect the security of the system, hence this should be disabled.This recommendation is applicable to SQL Server database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Ensure the database flag `external scripts enabled` that has been set is listed under the `Database flags` section.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Ensure the below command returns `off` for every Cloud SQL SQL Server database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="external scripts
enabled")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the SQL Server instance for which you want to enable to database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `external scripts enabled` from the drop-down menu, and set its value to `off`.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `external scripts enabled` database flag for every Cloud SQL SQL Server database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags "external scripts
enabled=off"

Note :

This command will overwrite all database flags previously set. To keep those
and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `external scripts enabled` is `off`

References:

1. https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/external-scripts-enabled-server-configuration-option?view=sql-server-ver15
2. https://cloud.google.com/sql/docs/sqlserver/flags
3. https://docs.microsoft.com/en-us/sql/advanced-analytics/concepts/security?view=sql-server-ver15

4. https://www.stigviewer.com/stig/ms_sql_server_2016_instance/2018-03-09/finding/V-79347

Additional Information:

```
"WARNING: This patch modifies database flag values, which may require

your instance to be restarted. Check the list of supported flags -

https://cloud.google.com/sql/docs/sqlserver/flags - to see if your

instance will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability, and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines."



Note: Configuring the above flag restarts the Cloud SQL instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	2.9 Implement Application Whitelisting of Scripts The organization's application whitelisting software must ensure that only authorized, digitally signed scripts (such as *.ps1, *.py, macros, etc) are allowed to run on a system.			●

6.3.2 Ensure that the 'cross db ownership chaining' database flag for Cloud SQL SQL Server instance is set to 'off' (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to set `cross db ownership chaining` database flag for Cloud SQL SQL Server instance to `off`.

Rationale:

Use the `cross db ownership` for chaining option to configure cross-database ownership chaining for an instance of Microsoft SQL Server. This server option allows you to control cross-database ownership chaining at the database level or to allow cross-database ownership chaining for all databases. Enabling `cross db ownership` is not recommended unless all of the databases hosted by the instance of SQL Server must participate in cross-database ownership chaining and you are aware of the security implications of this setting.This recommendation is applicable to SQL Server database instances.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Ensure the database flag `cross db ownership chaining` that has been set is listed under the `Database flags` section.

From Command Line:

1. List all Cloud SQL database Instances using the following command:

```
gcloud sql instances list
```

2. Ensure the below command returns `off` for every Cloud SQL SQL Server database instance:

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="cross db ownership
chaining")|.value'
```

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the SQL Server instance for which you want to enable to database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `cross db ownership chaining` from the drop-down menu, and set its value to `off`.
6. Click `Save`.
7. Confirm the changes under `Flags` on the Overview page.

From Command Line:

1. List all Cloud SQL database instances using the following command:

````
gcloud sql instances list
``
2. Configure the `cross db ownership chaining` database flag for every Cloud
SQL SQL Server database instance using the below command:
````

gcloud sql instances patch INSTANCE_NAME --database-flags "cross db ownership chaining=off"

```

```

Note:
This command will overwrite all database flags previously set. To keep those and add new ones, include the values for all flags to be set on the instance; any flag not specifically included is set to its default value. For flags that do not take a value, specify the flag name followed by an equals sign ("=").

References:

1. https://cloud.google.com/sql/docs/sqlserver/flags
2. https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/cross-db-ownership-chaining-server-configuration-option?view=sql-server-ver15

Additional Information:

```
WARNING: This patch modifies database flag values, which may require

your instance to be restarted. Check the list of supported flags -
```

```
https://cloud.google.com/sql/docs/sqlserver/flags - to see if your

instance will be restarted when this patch is submitted.

Note: Some database flag settings can affect instance availability or
stability, and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.



Note: Configuring the above flag does not restart the Cloud SQL instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

6.3.3 Ensure 'user connections' database flag for Cloud SQL SQL Server instance is set as appropriate (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to set `user connections` database flag for Cloud SQL SQL Server instance according organization-defined value.

Rationale:

The `user connections` option specifies the maximum number of simultaneous user connections that are allowed on an instance of SQL Server. The actual number of user connections allowed also depends on the version of SQL Server that you are using, and also the limits of your application or applications and hardware. SQL Server allows a maximum of 32,767 user connections. Because user connections is a dynamic (self-configuring) option, SQL Server adjusts the maximum number of user connections automatically as needed, up to the maximum value allowable. For example, if only 10 users are logged in, 10 user connection objects are allocated. In most cases, you do not have to change the value for this option. The default is 0, which means that the maximum (32,767) user connections are allowed. This recommendation is applicable to SQL Server database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Ensure the database flag `user connections` that has been set is listed under the `Database flags` section.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Ensure the below command returns value, which is according to your organization recommended value, for every Cloud SQL SQL Server database instance.

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="user connections")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the SQL Server instance for which you want to enable to database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `user connections` from the drop-down menu, and set its value to your organization recommended value.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `user connections` database flag for every Cloud SQL SQL Server database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags "user
connections=[0-32,767]"

Note :

This command will overwrite all database flags previously set. To keep those
and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default `user connections` is configured `0 (32, 767)`.

References:

1. https://cloud.google.com/sql/docs/sqlserver/flags

2. https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-the-user-connections-server-configuration-option?view=sql-server-ver15
3. https://www.stigviewer.com/stig/ms_sql_server_2016_instance/2018-03-09/finding/V-79119

Additional Information:

```
WARNING: This patch modifies database flag values, which may require

your instance to be restarted. Check the list of supported flags -

https://cloud.google.com/sql/docs/sqlserver/flags - to see if your

instance will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability, and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.

Note: Configuring the above flag does not restart the Cloud SQL instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	16 Account Monitoring and Control Account Monitoring and Control			

6.3.4 Ensure 'user options' database flag for Cloud SQL SQL Server instance is not configured (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended that, `user options` database flag for Cloud SQL SQL Server instance should not be configured.

Rationale:

The `user options` option specifies global defaults for all users. A list of default query processing options is established for the duration of a user's work session. The user options option allows you to change the default values of the SET options (if the server's default settings are not appropriate).

A user can override these defaults by using the SET statement. You can configure user options dynamically for new logins. After you change the setting of user options, new login sessions use the new setting; current login sessions are not affected. This recommendation is applicable to SQL Server database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Ensure the database flag `user options` that has been set is not listed under the `Database flags` section.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Ensure the below command returns empty result for every Cloud SQL SQL Server database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="user options")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the SQL Server instance for which you want to enable to database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. Click the X next `user options` flag shown
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Clear the `user options` database flag for every Cloud SQL SQL Server database instance using either of the below commands.

```
1.Clearing all flags to their default value

gcloud sql instances patch [INSTANCE_NAME] --clear-database-flags

OR
2. To clear only `user options` database flag, configure the database flag by
overriding the `user options`. Exclude `user options` flag and its value, and
keep all other flags you want to configure.

gcloud sql instances patch [INSTANCE_NAME] --database-flags
[FLAG1=VALUE1,FLAG2=VALUE2]

Note :

This command will overwrite all database flags previously set. To keep those
and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default 'user options' is not configured.

References:

1. https://cloud.google.com/sql/docs/sqlserver/flags
2. https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-the-user-options-server-configuration-option?view=sql-server-ver15
3. https://www.stigviewer.com/stig/ms_sql_server_2016_instance/2018-03-09/finding/V-79335

Additional Information:

```
WARNING: This patch modifies database flag values, which may require

your instance to be restarted. Check the list of supported flags -

https://cloud.google.com/sql/docs/sqlserver/flags - to see if your

instance will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability, and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.



Note: Configuring the above flag does not restart the Cloud SQL instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	5.1 Establish Secure Configurations Maintain documented, standard security configuration standards for all authorized operating systems and software.	●	●	●

6.3.5 Ensure 'remote access' database flag for Cloud SQL SQL Server instance is set to 'off' (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to set `remote access` database flag for Cloud SQL SQL Server instance to `off`.

Rationale:

The `remote access` option controls the execution of stored procedures from local or remote servers on which instances of SQL Server are running. This default value for this option is 1. This grants permission to run local stored procedures from remote servers or remote stored procedures from the local server. To prevent local stored procedures from being run from a remote server or remote stored procedures from being run on the local server, this must be disabled. The Remote Access option controls the execution of local stored procedures on remote servers or remote stored procedures on local server. 'Remote access' functionality can be abused to launch a Denial-of-Service (DoS) attack on remote servers by off-loading query processing to a target, hence this should be disabled. This recommendation is applicable to SQL Server database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Ensure the database flag `remote access` that has been set is listed under the `Database flags` section.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Ensure the below command returns `off` for every Cloud SQL SQL Server database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="remote access")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the SQL Server instance for which you want to enable to database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `remote access` from the drop-down menu, and set its value to `off`.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `remote access` database flag for every Cloud SQL SQL Server database instance using the below command

```
gcloud sql instances patch INSTANCE_NAME --database-flags "remote access=off"

Note :

This command will overwrite all database flags previously set. To keep those
and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

Default Value:

By default 'remote access' is 'on'.

References:

1. https://cloud.google.com/sql/docs/sqlserver/flags

2. https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-the-remote-access-server-configuration-option?view=sql-server-ver15
3. https://www.stigviewer.com/stig/ms_sql_server_2016_instance/2018-03-09/finding/V-79337

Additional Information:

```
WARNING: This patch modifies database flag values, which may require

your instance to be restarted. Check the list of supported flags -

https://cloud.google.com/sql/docs/sqlserver/flags - to see if your

instance will be restarted when this patch is submitted.

Note: some database flag settings can affect instance availability or
stability, and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.



Note: Configuring the above flag does not restart the Cloud SQL instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

6.3.6 Ensure '3625 (trace flag)' database flag for Cloud SQL SQL Server instance is set to 'off' (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to set `3625 (trace flag)` database flag for Cloud SQL SQL Server instance to `off`.

Rationale:

Trace flags are frequently used to diagnose performance issues or to debug stored procedures or complex computer systems, but they may also be recommended by Microsoft Support to address behavior that is negatively impacting a specific workload. All documented trace flags and those recommended by Microsoft Support are fully supported in a production environment when used as directed. `3625(trace log)` Limits the amount of information returned to users who are not members of the sysadmin fixed server role, by masking the parameters of some error messages using '******'. This can help prevent disclosure of sensitive information, hence this is recommended to disable this flag. This recommendation is applicable to SQL Server database instances.

Audit:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Ensure the database flag `3625` that has been set is listed under the `Database flags` section.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Ensure the below command returns `off` for every Cloud SQL SQL Server database instance

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="3625")|.value'
```

Remediation:

Using Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the SQL Server instance for which you want to enable to database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `3625` from the drop-down menu, and set its value to `off`.
6. Click `Save` to save your changes.
7. Confirm your changes under `Flags` on the Overview page.

Using Command Line:

1. List all Cloud SQL database Instances

```
gcloud sql instances list
```

2. Configure the `3625` database flag for every Cloud SQL SQL Server database instance using the below command.

```
gcloud sql instances patch INSTANCE_NAME --database-flags "3625=off"

Note :

This command will overwrite all database flags previously set. To keep those
and add new ones, include the values for all flags you want set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

References:

1. https://cloud.google.com/sql/docs/sqlserver/flags
2. https://docs.microsoft.com/en-us/sql/t-sql/database-console-commands/dbcc-traceon-trace-flags-transact-sql?view=sql-server-ver15#trace-flags

Additional Information:

```
WARNING: This patch modifies database flag values, which may require

your instance to be restarted. Check the list of supported flags -

https://cloud.google.com/sql/docs/sqlserver/flags - to see if your

instance will be restarted when this patch is submitted.



Note: some database flag settings can affect instance availability or
stability, and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.



Note: Configuring the above flag restarts the Cloud SQL instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			

6.3.7 Ensure that the 'contained database authentication' database flag for Cloud SQL on the SQL Server instance is set to 'off' (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to set `contained database authentication` database flag for Cloud SQL on the SQL Server instance is set to `off`.

Rationale:

A contained database includes all database settings and metadata required to define the database and has no configuration dependencies on the instance of the Database Engine where the database is installed. Users can connect to the database without authenticating a login at the Database Engine level. Isolating the database from the Database Engine makes it possible to easily move the database to another instance of SQL Server. Contained databases have some unique threats that should be understood and mitigated by SQL Server Database Engine administrators. Most of the threats are related to the USER WITH PASSWORD authentication process, which moves the authentication boundary from the Database Engine level to the database level, hence this is recommended to disable this flag. This recommendation is applicable to SQL Server database instances.

Impact:

When `contained database authentication` is off (0) for the instance, contained databases cannot be created, or attached to the Database Engine.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance to open its `Instance Overview` page
3. Ensure the database flag `contained database authentication` that has been set is listed under the `Database flags` section.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Ensure the below command returns `off` for every Cloud SQL SQL Server database instance.

```
gcloud sql instances describe INSTANCE_NAME --format=json | jq
'.settings.databaseFlags[] | select(.name=="contained database
authentication")|.value'
```

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the SQL Server instance for which you want to enable to database flag.
3. Click `Edit`.
4. Scroll down to the `Flags` section.
5. To set a flag that has not been set on the instance before, click `Add item`, choose the flag `contained database authentication` from the drop-down menu, and set its value to `off`.
6. Click `Save`.
7. Confirm the changes under `Flags` on the Overview page.

From Command Line:

1. List all Cloud SQL database Instances using the following command:

```
gcloud sql instances list
```

2. Configure the `contained database authentication` database flag for every Cloud SQL SQL Server database instance using the below command:

```
gcloud sql instances patch INSTANCE_NAME --database-flags "contained database
authentication=off"

Note:

This command will overwrite all database flags previously set. To keep those
and add new ones, , include the values for all flags to be set on the
instance; any flag not specifically included is set to its default value. For
flags that do not take a value, specify the flag name followed by an equals
sign ("=").
```

References:

1. https://cloud.google.com/sql/docs/sqlserver/flags

2. https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/contained-database-authentication-server-configuration-option?view=sql-server-ver15
3. https://docs.microsoft.com/en-us/sql/relational-databases/databases/security-best-practices-with-contained-databases?view=sql-server-ver15

Additional Information:

```
WARNING: This patch modifies database flag values, which may require

your instance to be restarted. Check the list of supported flags -

https://cloud.google.com/sql/docs/sqlserver/flags - to see if your

instance will be restarted when this patch is submitted.

Note: Some database flag settings can affect instance availability or
stability, and remove the instance from the Cloud SQL SLA. For information
about these flags, see Operational Guidelines.



Note: Configuring the above flag does not restart the Cloud SQL instance.
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

6.4 Ensure that the Cloud SQL database instance requires all incoming connections to use SSL (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to enforce all incoming connections to SQL database instance to use SSL.

Rationale:

SQL database connections if successfully trapped (MITM); can reveal sensitive data like credentials, database queries, query outputs etc. For security, it is recommended to always use SSL encryption when connecting to your instance. This recommendation is applicable for Postgresql, MySql generation 1, MySql generation 2 and SQL Server 2017 instances.

Impact:

After enforcing SSL connection, existing client will not be able to communicate with SQL server unless configured with appropriate client-certificates to communicate to SQL database instance.

Audit:

From Console:

1. Go to https://console.cloud.google.com/sql/instances.
2. Click on an instance name to see its configuration overview.
3. In the left-side panel, select `Connections`.
4. In the `SSL connections` section, ensure that `Only secured connections are allowed to connect to this instance.`.

From Command Line:

1. List all SQL database instances using the following command:

```
gcloud sql instances list
```

2. Get the detailed configuration for every SQL database instance using the following command:

```
gcloud sql instances describe INSTANCE_NAME
```

Ensure that section `settings: ipConfiguration` has the parameter `requireSsl` set to `true`.

Remediation:

From Console:

1. Go to https://console.cloud.google.com/sql/instances.
2. Click on an instance name to see its configuration overview.
3. In the left-side panel, select `Connections`.
4. In the `SSL connections` section, click `Allow only SSL connections`.
5. Under `Configure SSL server certificates` click `Create new certificate`.
6. Under `Configure SSL client certificates` click `Create a client certificate`.
7. Follow the instructions shown to learn how to connect to your instance.

From Command Line:
To enforce SSL encryption for an instance run the command:

```
gcloud sql instances patch INSTANCE_NAME --require-ssl
```

Note:
`RESTART` is required for type MySQL Generation 1 Instances (`backendType: FIRST_GEN`) to get this configuration in effect.

Default Value:

By default parameter `settings: ipConfiguration: requireSsl` is not set which is equivalent to `requireSsl:false`.

References:

1. https://cloud.google.com/sql/docs/postgres/configure-ssl-instance

Additional Information:

By default `Settings: ipConfiguration` has no `authorizedNetworks` set/configured. In that case even if by default `requireSsl` is not set, which is equivalent to `requireSsl:false` there is no risk as instance cannot be accessed outside of the network unless `authorizedNetworks` are configured. However, If default for `requireSsl` is not updated to `true` any `authorizedNetworks` created later on will not enforce SSL only connection.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●
v7	16.5 Encrypt Transmittal of Username and Authentication Credentials Ensure that all account usernames and authentication credentials are transmitted across networks using encrypted channels.		●	●

6.5 Ensure that Cloud SQL database instances are not open to the world (Automated)

Profile Applicability:

- Level 1

Description:

Database Server should accept connections only from trusted Network(s)/IP(s) and restrict access from the world.

Rationale:

To minimize attack surface on a Database server instance, only trusted/known and required IP(s) should be white-listed to connect to it.

An authorized network should not have IPs/networks configured to `0.0.0.0/0` which will allow access to the instance from anywhere in the world. Note that authorized networks apply only to instances with public IPs.

Impact:

The Cloud SQL database instance would not be available to the world.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Click the instance name to open its `Instance details` page.
3. Under the `Configuration` section click `Edit configurations`
4. Under `Configuration options` expand the `Connectivity` section.
5. Ensure that no authorized network is configured to allow `0.0.0.0/0`.

From Command Line:

1. List all Cloud SQL database Instances using the following command:

```
gcloud sql instances list
```

2. Get detailed configuration for every Cloud SQL database instance.

```
gcloud sql instances describe INSTANCE_NAME
```

Ensure that the section `settings: ipConfiguration : authorizedNetworks` does not have any parameter `value` containing `0.0.0.0/0`.

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Click the instance name to open its `Instance details` page.
3. Under the `Configuration` section click `Edit configurations`
4. Under `Configuration options` expand the `Connectivity` section.
5. Click the `delete` icon for the authorized network `0.0.0.0/0`.
6. Click `Save` to update the instance.

From Command Line:
Update the authorized network list by dropping off any addresses.

```
gcloud sql instances patch INSTANCE_NAME --authorized-
networks=IP_ADDR1,IP_ADDR2...
```

Prevention:
To prevent new SQL instances from being configured to accept incoming connections from any IP addresses, set up a `Restrict Authorized Networks on Cloud SQL instances` Organization Policy at: https://console.cloud.google.com/iam-admin/orgpolicies/sql-restrictAuthorizedNetworks.

Default Value:

By default, authorized networks are not configured. Remote connection to Cloud SQL database instance is not possible unless authorized networks are configured.

References:

1. https://cloud.google.com/sql/docs/mysql/configure-ip
2. https://console.cloud.google.com/iam-admin/orgpolicies/sql-restrictAuthorizedNetworks
3. https://cloud.google.com/resource-manager/docs/organization-policy/org-policy-constraints
4. https://cloud.google.com/sql/docs/mysql/connection-org-policy

Additional Information:

There is no IPv6 configuration found for Google cloud SQL server services.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

6.6 Ensure that Cloud SQL database instances do not have public IPs (Automated)

Profile Applicability:

- Level 2

Description:

It is recommended to configure Second Generation Sql instance to use private IPs instead of public IPs.

Rationale:

To lower the organization's attack surface, Cloud SQL databases should not have public IPs. Private IPs provide improved network security and lower latency for your application.

Impact:

Removing the public IP address on SQL instances may break some applications that relied on it for database connectivity.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console: https://console.cloud.google.com/sql/instances
2. Ensure that every instance has a private IP address and no public IP address configured.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. For every instance of type `instanceType: CLOUD_SQL_INSTANCE` with `backendType: SECOND_GEN`, get detailed configuration. Ignore instances of type `READ_REPLICA_INSTANCE` because these instances inherit their settings from the primary instance. Also, note that first generation instances cannot be configured to have a private IP address.

```
gcloud sql instances describe INSTANCE_NAME
```

3. Ensure that the setting `ipAddresses` has an IP address configured of `type: PRIVATE` and has no IP address of `type: PRIMARY`. `PRIMARY` email addresses are public addresses. An instance can have both a private and public address at the same time. Note also that you cannot use private IP with First Generation instances.

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console: https://console.cloud.google.com/sql/instances
2. Click the instance name to open its Instance details page.
3. Select the `Connections` tab.
4. Deselect the `Public IP` checkbox.
5. Click `Save` to update the instance.

From Command Line:

1. For every instance remove its public IP and assign a private IP instead:

```
gcloud beta sql instances patch INSTANCE_NAME --network=VPC_NETWOR_NAME --no-
assign-ip
```

2. Confirm the changes using the following command::

```
gcloud sql instances describe INSTANCE_NAME
```

Prevention:

To prevent new SQL instances from getting configured with public IP addresses, set up a `Restrict Public IP access on Cloud SQL instances` Organization policy at: https://console.cloud.google.com/iam-admin/orgpolicies/sql-restrictPublicIp.

Default Value:

By default, Cloud Sql instances have a public IP.

References:

1. https://cloud.google.com/sql/docs/mysql/configure-private-ip
2. https://cloud.google.com/sql/docs/mysql/private-ip
3. https://cloud.google.com/resource-manager/docs/organization-policy/org-policy-constraints
4. https://console.cloud.google.com/iam-admin/orgpolicies/sql-restrictPublicIp

Additional Information:

Replicas inherit their private IP status from their primary instance. You cannot configure a private IP directly on a replica.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			

6.7 Ensure that Cloud SQL database instances are configured with automated backups (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended to have all SQL database instances set to enable automated backups.

Rationale:

Backups provide a way to restore a Cloud SQL instance to recover lost data or recover from a problem with that instance. Automated backups need to be set for any instance that contains data that should be protected from loss or damage. This recommendation is applicable for SQL Server, PostgreSql, MySql generation 1 and MySql generation 2 instances.

Audit:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Click the instance name to open its instance details page.
3. Go to the `Backups` menu.
4. Ensure that `Automated backups` is set to `Enabled` and `Backup time` is mentioned.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Ensure that the below command returns `True` for every Cloud SQL database instance.

```
gcloud sql instances describe INSTANCE_NAME --
format="value('Enabled':settings.backupConfiguration.enabled)"
```

Remediation:

From Console:

1. Go to the Cloud SQL Instances page in the Google Cloud Console by visiting https://console.cloud.google.com/sql/instances.
2. Select the instance where the backups need to be configured.
3. Click `Edit`.
4. In the `Backups` section, check `Enable automated backups', and choose a backup window.
5. Click `Save`.

From Command Line:

1. List all Cloud SQL database instances using the following command:

```
gcloud sql instances list
```

2. Enable `Automated backups` for every Cloud SQL database instance using the below command:

```
gcloud sql instances patch INSTANCE_NAME --backup-start-time [HH:MM]
```

The `backup-start-time` parameter is specified in 24-hour time, in the UTC±00 time zone, and specifies the start of a 4-hour backup window. Backups can start any time during the backup window.

Default Value:

By default, automated backups are not configured for Cloud SQL instances. Data backup is not possible on any Cloud SQL instance unless Automated Backup is configured.

References:

1. https://cloud.google.com/sql/docs/mysql/backup-recovery/backups
2. https://cloud.google.com/sql/docs/postgres/backup-recovery/backing-up

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	10.1 Ensure Regular Automated Back Ups Ensure that all system data is automatically backed up on regular basis.	●	●	●

7 BigQuery

This section addresses Google CloudPlatform BigQuery. BigQuery is a serverless, highly-scalable, and cost-effective cloud data warehouse with an in-memory BI Engine and machine learning built in.

7.1 Ensure that BigQuery datasets are not anonymously or publicly accessible (Automated)

Profile Applicability:

- Level 1

Description:

It is recommended that the IAM policy on BigQuery datasets does not allow anonymous and/or public access.

Rationale:

Granting permissions to `allUsers` or `allAuthenticatedUsers` allows anyone to access the dataset. Such access might not be desirable if sensitive data is being stored in the dataset. Therefore, ensure that anonymous and/or public access to a dataset is not allowed.

Impact:

The dataset is not publicly accessible. Explicit modification of IAM privileges would be necessary to make them publicly accessible.

Audit:

From Console:

1. Go to `BigQuery` by visiting: https://console.cloud.google.com/bigquery.
2. Select a dataset from `Resources`.
3. Click `SHARE DATASET` near the right side of the window.
4. Validate that none of the attached roles contain `allUsers` or `allAuthenticatedUsers`.

From Command Line:

1. Retrieve the data set information using the following command:

```
bq show PROJECT_ID:DATASET_NAME
```

2. Ensure that `allUsers` and `allAuthenticatedUsers` have not been granted access to the dataset.

Remediation:

From Console:

1. Go to `BigQuery` by visiting: https://console.cloud.google.com/bigquery.
2. Select the dataset from 'Resources'.
3. Click `SHARE DATASET` near the right side of the window.
4. Review each attached role.
5. Click the delete icon for each member `allUsers` or `allAuthenticatedUsers`. On the popup click `Remove`.

From Command Line:

1. Retrieve the data set information:

```
bq show --format=prettyjson PROJECT_ID:DATASET_NAME > PATH_TO_FILE
```

2. In the access section of the JSON file, update the dataset information to remove all roles containing `allUsers` or `allAuthenticatedUsers`.
3. Update the dataset:

```
bq update --source PATH_TO_FILE PROJECT_ID:DATASET_NAME
```

Prevention:
You can prevent Bigquery dataset from becoming publicly accessible by setting up the `Domain restricted sharing` organization policy at: https://console.cloud.google.com/iam-admin/orgpolicies/iam-allowedPolicyMemberDomains .

Default Value:

By default, BigQuery datasets are not publicly accessible.

References:

1. https://cloud.google.com/bigquery/docs/dataset-access-controls

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the	●	●	●

Controls Version	Control	IG 1	IG 2	IG 3
	principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.			

7.2 Ensure that all BigQuery Tables are encrypted with Customer-managed encryption key (CMEK) (Automated)

Profile Applicability:

- Level 2

Description:

BigQuery by default encrypts the data as rest by employing `Envelope Encryption` using Google managed cryptographic keys. The data is encrypted using the `data encryption keys` and data encryption keys themselves are further encrypted using `key encryption keys`. This is seamless and do not require any additional input from the user. However, if you want to have greater control, Customer-managed encryption keys (CMEK) can be used as encryption key management solution for BigQuery Data Sets. If CMEK is used, the CMEK is used to encrypt the data encryption keys instead of using google-managed encryption keys.

Rationale:

BigQuery by default encrypts the data as rest by employing `Envelope Encryption` using Google managed cryptographic keys. This is seamless and does not require any additional input from the user.

For greater control over the encryption, customer-managed encryption keys (CMEK) can be used as encryption key management solution for BigQuery tables. The CMEK is used to encrypt the data encryption keys instead of using google-managed encryption keys. BigQuery stores the table and CMEK association and the encryption/decryption is done automatically.

Applying the Default Customer-managed keys on BigQuery data sets ensures that all the new tables created in the future will be encrypted using CMEK but existing tables need to be updated to use CMEK individually.

```
Note: Google does not store your keys on its servers and cannot access your
protected data unless you provide the key. This also means that if you forget
or lose your key, there is no way for Google to recover the key or to recover
any data encrypted with the lost key.
```

Impact:

Using Customer-managed encryption keys (CMEK) will incur additional labor-hour investment to create, protect, and manage the keys.

Audit:

From Console:

1. Go to `Big Data`
2. Go to `BigQuery`
3. Under `Resources`, select the project
4. Select Data Set, select the table
5. Go to `Details` tab
6. Under `Table info`, verify `Customer-managed key` is present.
7. Repeat for each table in all data sets for all projects.

From Command Line:
Use the following command to view the table details. Verify the `kmsKeyName` is present.

```
bq show <table_object>
```

Remediation:

Currently, there is no way to update the encryption of existing data in the table. The data needs to be copied to either an original table or another table while specifying the customer managed encryption key (CMEK).
From Command Line:
Use the following command to copy the data. The source and the destination needs to be same in case copying to the original table.

```
bq cp --destination_kms_key <customer_managed_key>
source_dataset.source_table destination_dataset.destination_table
```

Default Value:

Google Managed keys are used as `key encryption keys`.

References:

1. https://cloud.google.com/bigquery/docs/customer-managed-encryption

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			

7.3 Ensure that a Default Customer-managed encryption key (CMEK) is specified for all BigQuery Data Sets (Automated)

Profile Applicability:

- Level 2

Description:

BigQuery by default encrypts the data as rest by employing `Envelope Encryption` using Google managed cryptographic keys. The data is encrypted using the `data encryption keys` and data encryption keys themselves are further encrypted using `key encryption keys`. This is seamless and do not require any additional input from the user. However, if you want to have greater control, Customer-managed encryption keys (CMEK) can be used as encryption key management solution for BigQuery Data Sets.

Rationale:

BigQuery by default encrypts the data as rest by employing `Envelope Encryption` using Google managed cryptographic keys. This is seamless and does not require any additional input from the user.

For greater control over the encryption, customer-managed encryption keys (CMEK) can be used as encryption key management solution for BigQuery Data Sets. Setting a Default Customer-managed encryption key (CMEK) for a data set ensure any tables created in future will use the specified CMEK if none other is provided.

```
Note: Google does not store your keys on its servers and cannot access your
protected data unless you provide the key. This also means that if you forget
or lose your key, there is no way for Google to recover the key or to recover
any data encrypted with the lost key.
```

Impact:

Using Customer-managed encryption keys (CMEK) will incur additional labor-hour investment to create, protect, and manage the keys.

Audit:

From Console:

1. Go to `Big Data`
2. Go to `BigQuery`
3. Under `Resources`, select the project

4. Select Data Set
5. Ensure `Customer-managed key` is present under `Dataset info` section.
6. Repeat for each data set in all projects.

From Command Line:
Use the following command to view the data set details. Verify the `kmsKeyName` is present.

```
bq show <data_set_object>
```

Remediation:

The default CMEK for existing data sets can be updated by specifying the default key in the `EncryptionConfiguration.kmsKeyName` field when calling the `datasets.insert` or `datasets.patch` methods

Default Value:

Google Managed keys are used as `key encryption keys`.

References:

1. https://cloud.google.com/bigquery/docs/customer-managed-encryption

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	13 Data Protection Data Protection			

Appendix: Recommendation Summary Table

Control		Set Correctly	
		Yes	No
1	**Identity and Access Management**		
1.1	Ensure that corporate login credentials are used (Automated)	□	□
1.2	Ensure that multi-factor authentication is enabled for all non-service accounts (Manual)	□	□
1.3	Ensure that Security Key Enforcement is enabled for all admin accounts (Manual)	□	□
1.4	Ensure that there are only GCP-managed service account keys for each service account (Automated)	□	□
1.5	Ensure that Service Account has no Admin privileges (Automated)	□	□
1.6	Ensure that IAM users are not assigned the Service Account User or Service Account Token Creator roles at project level (Automated)	□	□
1.7	Ensure user-managed/external keys for service accounts are rotated every 90 days or less (Automated)	□	□
1.8	Ensure that Separation of duties is enforced while assigning service account related roles to users (Manual)	□	□
1.9	Ensure that Cloud KMS cryptokeys are not anonymously or publicly accessible (Automated)	□	□
1.10	Ensure KMS encryption keys are rotated within a period of 90 days (Automated)	□	□
1.11	Ensure that Separation of duties is enforced while assigning KMS related roles to users (Automated)	□	□
1.12	Ensure API keys are not created for a project (Manual)	□	□
1.13	Ensure API keys are restricted to use by only specified Hosts and Apps (Manual)	□	□
1.14	Ensure API keys are restricted to only APIs that application needs access (Manual)	□	□
1.15	Ensure API keys are rotated every 90 days (Manual)	□	□
2	**Logging and Monitoring**		
2.1	Ensure that Cloud Audit Logging is configured properly across all services and all users from a project (Automated)	□	□
2.2	Ensure that sinks are configured for all log entries (Automated)	□	□

2.3	Ensure that retention policies on log buckets are configured using Bucket Lock (Automated)	□	□
2.4	Ensure log metric filter and alerts exist for project ownership assignments/changes (Automated)	□	□
2.5	Ensure that the log metric filter and alerts exist for Audit Configuration changes (Automated)	□	□
2.6	Ensure that the log metric filter and alerts exist for Custom Role changes (Automated)	□	□
2.7	Ensure that the log metric filter and alerts exist for VPC Network Firewall rule changes (Automated)	□	□
2.8	Ensure that the log metric filter and alerts exist for VPC network route changes (Automated)	□	□
2.9	Ensure that the log metric filter and alerts exist for VPC network changes (Automated)	□	□
2.10	Ensure that the log metric filter and alerts exist for Cloud Storage IAM permission changes (Automated)	□	□
2.11	Ensure that the log metric filter and alerts exist for SQL instance configuration changes (Automated)	□	□
2.12	Ensure that Cloud DNS logging is enabled for all VPC networks (Automated)	□	□
3	**Networking**		
3.1	Ensure that the default network does not exist in a project (Automated)	□	□
3.2	Ensure legacy networks do not exist for a project (Automated)	□	□
3.3	Ensure that DNSSEC is enabled for Cloud DNS (Automated)	□	□
3.4	Ensure that RSASHA1 is not used for the key-signing key in Cloud DNS DNSSEC (Manual)	□	□
3.5	Ensure that RSASHA1 is not used for the zone-signing key in Cloud DNS DNSSEC (Manual)	□	□
3.6	Ensure that SSH access is restricted from the internet (Automated)	□	□
3.7	Ensure that RDP access is restricted from the Internet (Automated)	□	□
3.8	Ensure that VPC Flow Logs is enabled for every subnet in a VPC Network (Automated)	□	□
3.9	Ensure no HTTPS or SSL proxy load balancers permit SSL policies with weak cipher suites (Manual)	□	□
3.10	Ensure Firewall Rules for instances behind Identity Aware Proxy (IAP) only allow the traffic from Google Cloud Loadbalancer (GCLB) Health Check and Proxy Addresses (Manual)	□	□
4	**Virtual Machines**		

4.1	Ensure that instances are not configured to use the default service account (Automated)	□	□
4.2	Ensure that instances are not configured to use the default service account with full access to all Cloud APIs (Automated)	□	□
4.3	Ensure "Block Project-wide SSH keys" is enabled for VM instances (Automated)	□	□
4.4	Ensure oslogin is enabled for a Project (Automated)	□	□
4.5	Ensure 'Enable connecting to serial ports' is not enabled for VM Instance (Automated)	□	□
4.6	Ensure that IP forwarding is not enabled on Instances (Automated)	□	□
4.7	Ensure VM disks for critical VMs are encrypted with Customer-Supplied Encryption Keys (CSEK) (Automated)	□	□
4.8	Ensure Compute instances are launched with Shielded VM enabled (Automated)	□	□
4.9	Ensure that Compute instances do not have public IP addresses (Automated)	□	□
4.10	Ensure that App Engine applications enforce HTTPS connections (Manual)	□	□
4.11	Ensure that Compute instances have Confidential Computing enabled (Automated)	□	□
5	**Storage**		
5.1	Ensure that Cloud Storage bucket is not anonymously or publicly accessible (Automated)	□	□
5.2	Ensure that Cloud Storage buckets have uniform bucket-level access enabled (Automated)	□	□
6	**Cloud SQL Database Services**		
6.1	**MySQL Database**		
6.1.1	Ensure that a MySQL database instance does not allow anyone to connect with administrative privileges (Automated)	□	□
6.1.2	Ensure 'skip_show_database' database flag for Cloud SQL Mysql instance is set to 'on' (Automated)	□	□
6.1.3	Ensure that the 'local_infile' database flag for a Cloud SQL Mysql instance is set to 'off' (Automated)	□	□
6.2	**PostgreSQL Database**		
6.2.1	Ensure that the 'log_checkpoints' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Automated)	□	□
6.2.2	Ensure 'log_error_verbosity' database flag for Cloud SQL PostgreSQL instance is set to 'DEFAULT' or stricter (Manual)	□	□
6.2.3	Ensure that the 'log_connections' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Automated)	□	□

6.2.4	Ensure that the 'log_disconnections' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Automated)	□	□
6.2.5	Ensure 'log_duration' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Manual)	□	□
6.2.6	Ensure that the 'log_lock_waits' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Automated)	□	□
6.2.7	Ensure 'log_statement' database flag for Cloud SQL PostgreSQL instance is set appropriately (Manual)	□	□
6.2.8	Ensure 'log_hostname' database flag for Cloud SQL PostgreSQL instance is set appropriately (Automated)	□	□
6.2.9	Ensure 'log_parser_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Automated)	□	□
6.2.10	Ensure 'log_planner_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Automated)	□	□
6.2.11	Ensure 'log_executor_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Automated)	□	□
6.2.12	Ensure 'log_statement_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Automated)	□	□
6.2.13	Ensure that the 'log_min_messages' database flag for Cloud SQL PostgreSQL instance is set appropriately (Manual)	□	□
6.2.14	Ensure 'log_min_error_statement' database flag for Cloud SQL PostgreSQL instance is set to 'Error' or stricter (Automated)	□	□
6.2.15	Ensure that the 'log_temp_files' database flag for Cloud SQL PostgreSQL instance is set to '0' (on) (Automated)	□	□
6.2.16	Ensure that the 'log_min_duration_statement' database flag for Cloud SQL PostgreSQL instance is set to '-1' (disabled) (Automated)	□	□
6.3	**SQL Server**		
6.3.1	Ensure 'external scripts enabled' database flag for Cloud SQL SQL Server instance is set to 'off' (Automated)	□	□
6.3.2	Ensure that the 'cross db ownership chaining' database flag for Cloud SQL SQL Server instance is set to 'off' (Automated)	□	□
6.3.3	Ensure 'user connections' database flag for Cloud SQL SQL Server instance is set as appropriate (Automated)	□	□
6.3.4	Ensure 'user options' database flag for Cloud SQL SQL Server instance is not configured (Automated)	□	□
6.3.5	Ensure 'remote access' database flag for Cloud SQL SQL Server instance is set to 'off' (Automated)	□	□
6.3.6	Ensure '3625 (trace flag)' database flag for Cloud SQL SQL Server instance is set to 'off' (Automated)	□	□
6.3.7	Ensure that the 'contained database authentication' database flag for Cloud SQL on the SQL Server instance is set to 'off' (Automated)	□	□

6.4	Ensure that the Cloud SQL database instance requires all incoming connections to use SSL (Automated)	□	□
6.5	Ensure that Cloud SQL database instances are not open to the world (Automated)	□	□
6.6	Ensure that Cloud SQL database instances do not have public IPs (Automated)	□	□
6.7	Ensure that Cloud SQL database instances are configured with automated backups (Automated)	□	□
7	**BigQuery**		
7.1	Ensure that BigQuery datasets are not anonymously or publicly accessible (Automated)	□	□
7.2	Ensure that all BigQuery Tables are encrypted with Customer-managed encryption key (CMEK) (Automated)	□	□
7.3	Ensure that a Default Customer-managed encryption key (CMEK) is specified for all BigQuery Data Sets (Automated)	□	□

Appendix: Change History

Date	Version	Changes for this version
Apr 10, 2020	1.2.0	ADD - Ensure 'skip_show_database' database flag for Cloud SQL Mysql instance is set to 'on' (Ticket 10215)
Mar 16, 2021	1.2.0	ADD - Ensure 'log_parser_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Ticket 10117)
Mar 16, 2021	1.2.0	ADD - Ensure 'log_planner_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Ticket 10118)
Mar 16, 2021	1.2.0	ADD - Ensure 'log_executor_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Ticket 10119)
Mar 16, 2021	1.2.0	ADD - Ensure 'log_statement_stats' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Ticket 10120)
Mar 16, 2021	1.2.0	ADD - Ensure 'log_min_error_statement' database flag for Cloud SQL PostgreSQL instance is set to 'Error' or stricter (Ticket 10122)
Apr 13, 2021	1.2.0	ADD - Ensure 'log_error_verbosity' database flag for Cloud SQL PostgreSQL instance is set to 'DEFAULT' or stricter (Ticket 10110)
Apr 13, 2021	1.2.0	ADD - Ensure 'log_statement' database flag for Cloud SQL PostgreSQL instance is set to 'ddl' or stricter (Ticket 10115)
Apr 13, 2021	1.2.0	ADD - Ensure 'log_hostname' database flag for Cloud SQL PostgreSQL instance is set to 'off' (Ticket 10116)
Apr 13, 2021	1.2.0	ADD- Ensure 'external scripts enabled' database flag for Cloud SQL SQL Server instance is set to 'off' (Ticket 10217)
Apr 13, 2021	1.2.0	ADD - Ensure 'user connections' database flag for Cloud SQL SQL Server instance is set as appropriate (Ticket 10219)
Apr 13, 2021	1.2.0	ADD - Ensure 'user options' database flag for Cloud SQL SQL Server instance is not configured (Ticket 10220)
Apr 13, 2021	1.2.0	ADD - Ensure 'remote access' database flag for Cloud SQL SQL Server instance is set to 'off' (Ticket 10222)

Apr 13, 2021	1.2.0	ADD - Ensure '3625 (trace flag)' database flag for Cloud SQL SQL Server instance is set to 'off' (Ticket 10223)
Apr 14, 2021	1.2.0	UPDATE - Ensure that the 'log_min_messages' database flag for Cloud SQL PostgreSQL instance is set appropriately - Title and Guideline content are not matching (Ticket 11145)
Apr 19, 2021	1.2.0	ADD - Ensure Firewall Rules for instances behind IAP only allow the traffic from GCLB Health Check and Proxy Addresses (Ticket 9464)
Apr 19, 2021	1.2.0	ADD - Ensure that all BigQuery Tables are encrypted with Customer-managed encryption key (CMEK) (Ticket 9975)
Apr 19, 2021	1.2.0	UPDATE - Ensure that the Cloud SQL database instance requires all incoming connections to use SSL - include SQL Server 2017 (Ticket 11885)
Apr 19, 2021	1.2.0	ADD - Ensure that a Default Customer-managed encryption key (CMEK) is specified for all BigQuery Data Setss (Ticket 9974)
Apr 19, 2021	1.2.0	ADD - Ensure 'log_duration' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Ticket 10113)
Apr 19, 2021	1.2.0	UPDATE - Ensure that VPC Flow Logs is enabled for every subnet in a VPC Network - VPC Flow Logs not supported for internal HTTPS LB subnets (Ticket 12553)
Apr 29, 2021	1.2.0	Modify - Ensure 'log_hostname' database flag for Cloud SQL PostgreSQL instance is set appropriately - Sync with Postgresql 9.5 benchmark v1.1.0 (Ticket 10413)
Apr 29, 2021	1.2.0	UPDATE - Multiple in Logging section - log metric filters use the protoPayload fields (Ticket 12325)
Apr 30, 2021	1.2.0	UPDATE - Ensure 'log_statement' database flag for Cloud SQL PostgreSQL instance is set to 'ddl' or stricter - Align with PostgreSQL 9.5 Benchmark v1.1.0 (Ticket 10412)
Apr 30, 2021	1.2.0	UPDATE - Multiple Recommendations - Update menu references for Logging\Metrics; remove references to Stackdriver Account (Ticket 12755)

Apr 30, 2021	1.2.0	ADD - Ensure that Cloud DNS logging is enabled for all VPC networks (Ticket 12493)
Mar 3, 2020	1.1.0	ADD - Ensure 'log_checkpoints' database flag for Cloud SQL PostgreSQL instance is set to 'on'
Feb 26, 2020	1.1.0	UPDATE - Ensure "Block Project-wide SSH keys" enabled for VM instances - Small command typo: instacnces -> instances (Ticket 6956)
Feb 26, 2020	1.1.0	UPDATE - Ensure that Service Account has no Admin privileges - The suggested Audit Procedure steps for 1.4 exclude "*admin" roles. (Ticket 6955)
Feb 26, 2020	1.1.0	UPDATE - Ensure that there are only GCP-managed service account keys for each service account - Remediation Procedure Typo (Ticket 6974)
Feb 26, 2020	1.1.0	UPDATE- Ensure that Separation of duties is enforced while assigning KMS related roles to users - Rationale Statement Typo (Ticket 7001)
Feb 26, 2020	1.1.0	UPDATE - Ensure that there are only GCP-managed service account keys for each service account - Add Audit steps (Ticket 9833)
Feb 26, 2020	1.1.0	ADD - Ensure Cloud SQL Instances do not have public IP addresses (Ticket 9233)
Feb 26, 2020	1.1.0	ADD - Ensure that Cloud SQL database instances are configured with automated backups (Ticket 10018)
Feb 26, 2020	1.1.0	Update - Ensure that IP forwarding is not enabled on Instances - limit the scope of virtual machines this recommendation is applicable on (Ticket 10089)
Feb 26, 2020	1.1.0	UPDATE - Ensure that the default network does not exist in a project - add an explain of the default network to the rationale statement (Ticket 10082)
Feb 26, 2020	1.1.0	UPDATE - Ensure API keys are rotated every 90 days - Change to Unscored (Ticket 9972)

Feb 26, 2020	1.1.0	UPDATE - Ensure that SSH access is restricted from the internet - Adding Protocol TCP (Ticket 9807)
Feb 26, 2020	1.1.0	UPDATE - Ensure "Block Project-wide SSH keys" enabled for VM instances - limit the scope of virtual machines this recommendation is applicable on (Ticket 10084)
Feb 26, 2020	1.1.0	Update - Ensure that instances are not configured to use the default service account with full access to all Cloud APIs - limit the scope of virtual machines this recommendation is applicable on (Ticket 10085)
Feb 26, 2020	1.1.0	ADD - Ensure 'log_min_messages' database flag for Cloud SQL PostgreSQL instance is set appropriately (Ticket 10121)
Feb 26, 2020	1.1.0	ADD - Ensure that instances are not configured to use the default service account (Ticket 10108)
Feb 26, 2020	1.1.0	ADD - Ensure 'log_temp_files' database flag for Cloud SQL PostgreSQL instance is set to '0' (on) (Ticket 10123)
Feb 26, 2020	1.1.0	ADD - Ensure 'log_min_duration_statement' database flag for Cloud SQL PostgreSQL instance is set to '-1' (disabled) (Ticket 10124)
Feb 26, 2020	1.1.0	ADD - Ensure 'local_infile' database flag for Cloud SQL Mysql instance is set to 'off' (Ticket 10216)
Feb 26, 2020	1.1.0	ADD - Ensure 'cross db ownership chaining' database flag for Cloud SQL SQL Server instance is set to 'off' (Ticket 10218)
Feb 26, 2020	1.1.0	ADD - Ensure 'contained database authentication' database flag for Cloud SQL SQL Server instance is set to 'off' (Ticket 10224)
Feb 20, 2020	1.1.0	ADD - Ensure Appengine Applications are exposed via TLS (Ticket 9447)
Feb 19, 2020	1.1.0	ADD - Ensure 'log_connections' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Ticket 10111)
Feb 19, 2020	1.1.0	ADD - Ensure 'log_disconnections' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Ticket 10112)

Feb 19, 2020	1.1.0	ADD - Ensure 'log_lock_waits' database flag for Cloud SQL PostgreSQL instance is set to 'on' (Ticket 10114)
Feb 16, 2020	1.1.0	ADD - Ensure that BigQuery datasets are not anonymously or publicly accessible (Ticket 9449)
Feb 16, 2020	1.1.0	ADD - Ensure that Cloud KMS cryptokeys are not anonymously or publicly accessible (Ticket 10081)
Feb 16, 2020	1.1.0	ADD - Ensure Compute instances are launched with Shielded VM enabled (Ticket 9446)
Feb 16, 2020	1.1.0	ADD - Ensure that Compute instances do not have public IP addresses (Ticket 9823)
Feb 16, 2020	1.1.0	ADD - Ensure HTTPS and SSL proxy load balancers do not permit SSL policies with weak cipher suites (Ticket 9450)
Feb 10, 2020	1.1.0	UPDATE - Ensure the default network does not exist in a project - provide specific remediation instructions (Ticket 8358)
Feb 7, 2020	1.1.0	UPDATE - Ensure that corporate login credentials are used - Fix references (Ticket 9971)
Feb 6, 2020	1.1.0	UPDATE - Ensure legacy networks does not exists for a project - Change reference URLs (Ticket 10091)
Feb 6, 2020	1.1.0	UPDATE - Ensure that multi-factor authentication is enabled for all non-service accounts - add reference URL (Ticket 9973)
Feb 3, 2020	1.1.0	UPDATE - Ensure log metric filter and alerts exists for Project Ownership assignments/changes (Ticket 9514)
Feb 3, 2020	1.1.0	UPDATE- Ensure log metric filter and alerts exists for Audit Configuration Changes (Ticket 9515)
Feb 3, 2020	1.1.0	UPDATE - Ensure log metric filter and alerts exists for Custom Role changes (Ticket 9516)
Feb 3, 2020	1.1.0	UPDATE - Ensure log metric filter and alerts exists for VPC Network Firewall rule changes (Ticket 9517)

Feb 3, 2020	1.1.0	UPDATE - Ensure log metric filter and alerts exists for VPC network route changes (Ticket 9518)
Feb 3, 2020	1.1.0	UPDATE - Ensure log metric filter and alerts exists for VPC network changes (Ticket 9519)
Feb 3, 2020	1.1.0	UPDATE - Ensure log metric filter and alerts exists for Project Ownership assignments/changes - UI changes needed (Ticket 6960)
Feb 3, 2020	1.1.0	UPDATE - Multiple in section 2: 2.4-2.11 - Changes in Monitoring API and UI. Need update. (Ticket 8726)
Feb 2, 2020	1.1.0	DELETE - Ensure that MySQL Database Instance does not allows root login from any Host - suggest we remove this recommendation (Ticket 9410)
Feb 2, 2020	1.1.0	ADD - Ensure that Security Key enforcement is enabled for all admin accounts (Ticket 9444)
Feb 2, 2020	1.1.0	UPDATE - Ensure that corporate login credentials are used instead of Gmail accounts - change to include org-level IAM (Ticket 9442)
Feb 2, 2020	1.1.0	UPDATE - Ensure that ServiceAccount has no Admin privileges -- Ignore Apps Script service account (Ticket 9495)
Feb 2, 2020	1.1.0	UPDATE - Ensure the default network does not exist in a project -- Move to Level 2 profile (Ticket 9461)
Feb 2, 2020	1.1.0	ADD - Ensure that retention policies on log buckets are configured using Bucket Lock -- should replace object versioning (Ticket 9822)
Feb 2, 2020	1.1.0	ADD - Ensure that IAM users are not assigned Service Account Token Creator role at project level (Ticket 9445)
Feb 2, 2020	1.1.0	ADD - Ensure that Cloud Storage buckets have Bucket Policy Only enabled (Ticket 9448)
Feb 2, 2020	1.1.0	UPDATE - Ensure that sinks are configured for all Log entries - configure at organization or folder level (Ticket 9456)
Feb 2, 2020	1.1.0	DELETE - Ensure that object versioning is enabled on log-buckets (Ticket 9513)

Feb 2, 2020	1.1.0	UPDATE - Ensure that Cloud Audit Logging is configured properly across all services and all... - configure at folder or organization level (Ticket 9455)
Jan 28, 2020	1.1.0	DELETE - Ensure Private Google Access is enabled for all subnetwork in VPC Network - Following the recommendation doesn't make you more secure (Ticket 9462)
Jan 28, 2020	1.1.0	DELETE - Ensure that logging is enabled for Cloud storage buckets - redundant given CIS Benchmark 2.1 (Ticket 9486)
Jan 15, 2020	1.1.0	REMOVE - Ensure Stackdriver Logging is set to Enabled on Kubernetes Engine Clusters (Ticket 9465)
Jan 15, 2020	1.1.0	DELETE - Kubernetes Engine Section (Ticket 9824)
Jan 15, 2020	1.1.0	UPDATE - Ensure that sinks are configured for all Log entries - Grammatical Findings (Ticket 6984)
Jan 15, 2020	1.1.0	UPDATE - Ensure Encryption keys are rotated within a period of 365 days - Audit and Remediation Procedure Update (Ticket 7036)
Jan 15, 2020	1.1.0	UPDATE - Ensure that object versioning is enabled on log-buckets - update Description, Impact statement and Remediation (Ticket 7040)
Jan 15, 2020	1.1.0	UPDATE - Ensure that RSASHA1 is not used for key-signing key in Cloud DNS DNSSEC - Remediation CLI broken (Ticket 7073)
Jan 15, 2020	1.1.0	UPDATE - Ensure that RSASHA1 is not used for zone-signing key in Cloud DNS DNSSEC - Audit CLI update (Ticket 7075)
Jan 15, 2020	1.1.0	UPDATE - Ensure that RSASHA1 is not used for zone-signing key in Cloud DNS DNSSEC - Remediation CLI broken (Ticket 7074)
Jan 8, 2020	1.1.0	UPDATE - Ensure log metric filter and alerts exists for Audit Configuration Changes - Typo (AWS API calls) (Ticket 9668)
Feb 4, 2019	1.1.0	UPDATE - Ensure that RSASHA1 is not used for key-signing key - update audit CLI (Ticket 7072)
Sep 5, 2018	1.0.0	Document created