Creating a BigQuery Authorized View

1 hour No cost

Overview

When using BigQuery, permissions are configured at the dataset level. Frequently, data engineering teams maintain datasets with many large tables of raw data, but they want to share subsets of these tables with particular analyst audiences.

For example, analysts might have access to a version of a table that excludes columns with user-specific information. Or, perhaps a specific user should be able to see only specific rows from a given BigQuery table or view.

In this lab, you will learn how to create and use Authorized Views in BigQuery. You will also learn how to do row-level filtering using information about the logged-in user.

This lab will provide two Google Cloud users. This is so the BigQuery authorized view permissions can be verified by logging in as a different user.

Objectives

In this lab, you will learn how to perform the following tasks:

- Set permissions on BigQuery datasets.
- Use Authorized Views to provide audiences read-only access to subsets of tables.
- Use the `SESSION_USER()` function to limit access to specific rows within a table/view.

Setup and requirements

For each lab, you get a new Google Cloud project and set of resources for a fixed time at no cost.

1. Sign in to Qwiklabs using an **incognito window**.

2. Note the lab's access time (for example, `1:15:00`), and make sure you can finish within that time.
There is no pause feature. You can restart if needed, but you have to start at the beginning.

3. When ready, click **Start lab**.

4. Note your lab credentials (**Username** and **Password**). You will use them to sign in to the Google Cloud Console.

5. Click **Open Google Console**.

6. Click **Use another account** and copy/paste credentials for **this** lab into the prompts.
If you use other credentials, you'll receive errors or **incur charges**.

7. Accept the terms and skip the recovery resource page.

Note: Do not click **End Lab** unless you have finished the lab or want to restart it. This clears your work and removes the project.

Task 1. Create the source dataset

In this task, you create the source dataset in BigQuery that will be used in this lab.

Create a new BigQuery dataset

1. In the Cloud Console, open the BigQuery user interface by selecting **Navigation menu > BigQuery**, then click **Done**.

2. Create a new dataset within your project by clicking the three dots next to your project ID in the **Explorer** section, then click on **Create dataset**.

3. Enter `source_data` for **Dataset ID**, and click on **Create Dataset** (accepting the other default values).

Create a new BigQuery table with source data

1. Click **Activate Cloud Shell** to open Cloud Shell. If prompted, click **Continue**.

2. Load the source data into a new table in BigQuery by entering the following in Cloud Shell:

```
bq load --autodetect $DEVSHELL_PROJ:source_data.events gs://cloud-
training/gcpsec/labs/bq-authviews-source.csv
```

content_c

3. In the BigQuery console, review the loaded data by drilling down to `[your_project_id].source_data.events` in the **Explorer** section and clicking **Preview**.

Note: You may need to refresh the browser to see the **events** table.

Note: This table has simulated data related to events generated by users of a videoconferencing application. Note that each row has information about the user who generated the event.

4. To ensure a future step will work as intended, enter the following query in the BigQuery **Editor**. Replace `<2nd qwiklabs user>` with `Username 2` in the qwiklabs console.

```
update source_data.events
set email='<2nd qwiklabs user>'
where email='rhonda.burns@example-dev.com'
```

content_c

5. Click **RUN** and wait for the 68 rows to be updated with a new email address.

Click *Check my progress* to verify the objective.

Create the source dataset

Check my progress

Assessment Completed!

Task 2. Create the analyst dataset

In this task, you create the analyst dataset, create a redacted view for the analysts, and create a second view for logged-in users.

Create a dataset

1. Create a new dataset within your project by clicking the three dots next to your project ID in the **Explorer** section, then click on **Create dataset**.

2. Enter `analyst_views` for **Dataset ID**, and click on **Create Dataset** (accepting the other default values).

Create a redacted view for analysts

3. In the BigQuery **Editor** area, enter the following SQL to get event data excluding the user-specific information:

```
SELECT
  date,
  type,
  company,
  call_duration,
  call_type,
  call_num_users,
  call_os,
  rating,
  comment,
  session_id,
  dialin_duration,
  ticket_number,
  ticket_driver
FROM
  `[your_project_id].source_data.events`
```

content_c

Note: Make sure to replace the `[your_project_id]` with the Qwiklabs-created project id.

4. **Run** the query and review the results. Note that the user information is not included.

5. Save the entered query as a view by clicking **Save > Save view**.

6. Select your project, and the **analyst_views** dataset.

7. Enter a destination table name of `no_user_info` and click **Save**. Note, though the UI says destination table, you are only creating a view, not a table.

8. Check that the view works by navigating in the **Explorer** section to **[your_project_id].analyst_views.no_user_info**. You should see the schema information for the view, which excludes user information columns.

9. Click on **no_user_info** view. Click **QUERY > In new tab** and enter `*` into the SELECT statement so that your SQL query looks like this:

```
SELECT
  *
FROM
  `[your_project_id].analyst_views.no_user_info`
LIMIT
  1000
```

content_c

10. **Run** the query and you should see results similar to those in above step, but without user data.

Create a second view, showing only rows for logged-in user

Next, create a 2nd view using the following information.

1. Enter the Query in the BigQuery **Editor**.

Note: Make sure to replace the `[your_project_id]` with the Qwiklabs-created project id.

```
SELECT
  *
FROM
  `[your_project_id].source_data.events`
WHERE
  email = SESSION_USER()
```

content_c

2. Click **Run**.

3. Save the entered query as a view by clicking **Save > Save view**.

4. Select your project, and the **analyst_views** dataset.

5. Enter a destination table name of `row_filter_session_user` and click **Save**.

Note: This 2nd view allows users to see their own events, but no one else's.

Click *Check my progress* to verify the objective.

Create the analyst dataset

Check my progress

Assessment Completed!

Task 3. Secure the analyst dataset

In this task you share the analyst dataset with Username 2, and secure it by providing the Viewer role.

Share the dataset

1. In the dataset listing to the left of the screen, click on the three dots next to the **analyst_views** dataset and click **Open in new tab**.

2. Then select **Sharing** from the right pane and click on **Permissions**.

3. Click **Add Principal**. In the **New principals** field, enter the email address of the 2nd lab account (Username 2) shown in the **Connection Details** section of the Qwiklabs lab page.

4. Select **BigQuery Data Viewer** as the role and click **Save**.

5. Click **Close**.

Task 4. Secure the source dataset

In this task, you secure the source dataset. You don't want analysts and others outside the data engineering team to have access to the raw data available in the source dataset, so you are going to restrict access.

You do want those using the views you've created to be able to see the data the views produce. This will require authorizing not the users, but the views.

Share the dataset

1. In the dataset listing to the left of the screen, click on the three dots next to the **source_data** dataset and click **Open in new tab**.

2. Then select **Sharing** from the right pane and click on **Permissions**.

3. Expand the **BigQuery Data Viewer** principal from the permission list and click on the trash icon next to it and click **Remove** to confirm. Click **Close**.

4. In **Sharing**, click on **Authorize Views** .

5. In the **Authorized views**, choose the following settings.

Authorized view	`no_user_info`

6. Click **Add Authorization**.

7. Add another entry with these settings replacing the existing view.

Authorized view	`row_filter_session_user`

8. Click **Add Authorization**.

9. Click **Close**.

Click *Check my progress* to verify the objective.

Secure the datasets

Check my progress

Please perform the tasks to secure analyst and source dataset.

Task 5. Test your security settings

In this task, you test the security settings that you applied in previous tasks.

Sign in to the Cloud Console as the second user

1. Open another tab in your incognito window.

2. Browse to the Cloud Console.

3. Click on the user icon in the top-right corner of the screen, and then click **Add account**.

4. Sign in to the Cloud Console with the **Username 2** provided in Qwiklabs.

Check access to the analyst views

1. In the Cloud Console, open the BigQuery user interface by selecting **Navigation menu > BigQuery**, then click **Done**.

2. Verify that you can run queries against the **no_user_info** view by executing the following query:

```
SELECT
  *
FROM
  `analyst_views.no_user_info`
WHERE
  type='register'
```

content_c

You should see a result set with all the user registration events within the table.

3. Verify that you can query the `row_filter_session_user` view, only seeing the rows associated with your account, by executing the following query:

```
SELECT
  *
FROM
  `analyst_views.row_filter_session_user`
```

content_c

You should see a result set with 68 rows specific to the second Qwiklabs user.

Check access to the source dataset

1. Try accessing the raw data in the events table directly using the following query:

```
SELECT
  *
FROM
  `source_data.events`
```

content_c

You will see an **Access denied** error message indicating that you don't have permissions.

2. Try navigating to the source_data dataset via the **Explorer** section of the UI. This should also be disallowed.

Note: The 2nd Qwiklabs user has permissions to view tables and views in the **analyst_views** dataset, but does not have permissions to view anything in the **source_data** dataset.

When the user queries the view, the view itself has permissions necessary to operate against the table in the **source_data** dataset, and it then returns that data to the user.

The row-filtering view gets the user's email address and uses that to filter the rows visible. Every user who queries this view will get different results, specifically the rows with her email in the email column.

Congratulations!

In this lab, you have learned how to do the following:

- Set permissions on BigQuery datasets.
- Authorize Views to provide audiences read-only access to subsets of tables.
- Use the `SESSION_USER()` function to limit access to specific rows within a table/view.

End your lab