GCS Fine-Grained Access Control (Preview)

Date published: 2023-06-29
Date modified: 2023-09-18

Legal Notice

This document has been released as part of a technical preview for features described herein. Technical preview components are provided as a convenience to our customers for their evaluation and trial usage. These components are provided ‘as is’ without warranty or support. Further, Cloudera assumes no liability for the usage of technical preview components, which should be used by customers at their own risk.

Contents

Introduction to RAZ in GCP environments

CDP Public Cloud defaults to using cloud storage which might be challenging while managing data access across teams and individual users. The Ranger Authorization Service (RAZ) resolves this challenge by enabling Google Cloud Storage (GCS) users to use fine-grained access policies and audit capabilities available in Apache Ranger similar to those used with HDFS files in an on-premises or IaaS deployment.

Many of the use cases that RAZ for GCS enables are cases where access control on files or directories is needed. Some examples include:

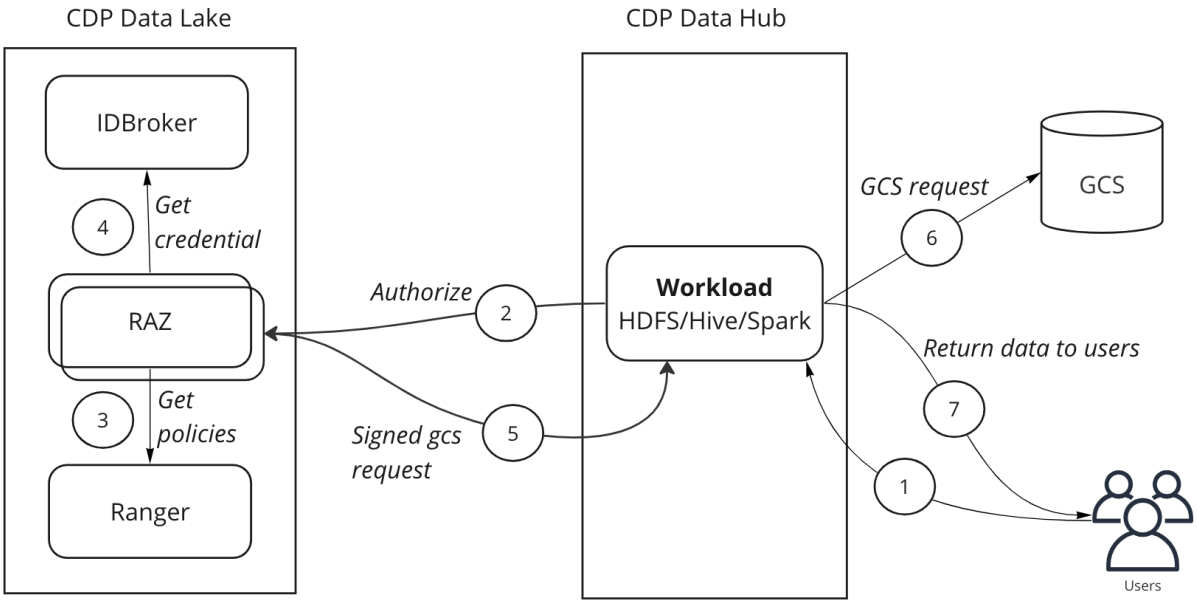
- Per-user home directories.
- Data engineering (Spark) efforts that require access to cloud storage objects and directories.
- Data warehouse queries (Hive/Impala) that use external tables.
- Access to Ranger's rich access control policies such as date-based access revocation, user/group/role-based controls, along with corresponding audit.
- Tag-based access control using the classification propagation feature that originates from directories.

Prior to the introduction of RAZ, controlling access to GCS could be enforced at coarse-grained group level using IDBroker mappings. This required rearchitecting the implementation of important file-centric activities as well as admin-level access to both the Google Cloud account and the CDP account.

In HDP and CDH deployments, files and directories are protected with a combination of HDFS Access Control Lists (ACLs) (in CDH, HDP) and Ranger HDFS policies (in HDP). Similarly, in an GCP CDP Public Cloud environment with RAZ for GCS enabled, Ranger's rich access control policies can be applied to CDP's access to GCS buckets, directories, and files and can be controlled with admin-level access to CDP alone.

Important
It is recommended that you do not set up IDBroker mappings for workload users in a RAZ-enabled GCP environment.

The following diagram illustrates RAZ authorization process for controlling user access to GCS:



Supported use cases for RAZ in GCP environments

The core RAZ for GCP for Data Lakes and several Data Hub templates are availables. The following Data Hub cluster types are supported with Runtime 7.2.17 or newer:

- Data Engineering
- Data Engineering HA
- Data Engineering Spark3
- Data Mart
- Flow Management Light Duty
- Operational Database with SQL
- Streams Messaging HA

Limitations for using RAZ in GCP environments

The following limitations and known issues have been identified:

- RAZ on GCP uses HMAC keys for authentication. When multiple CDP environments are created on GCP with RAZ enabled, the account limit of 10 HMAC keys per service account may be reached. To avoid reaching the limit, you should create a new Ranger Cloud Access Authorizer service account for each CDP environment on GCP.
- Due to the fact that Google HMAC key generation cannot be limited in scope, the Ranger Cloud Access Authorizer service account has elevated credentials to create HMAC keys for any service account in the same GCP project. Therefore, for security reasons, it is recommended that you create all CDP service accounts and storage buckets in a separate dedicated project.

- There is currently no automated way to enable RAZ in an existing CDP environment which does not have RAZ enabled. If you have a non-RAZ environment, you should create a new environment with RAZ, and then migrate your data using the steps described in Data Lake backup and restore documentation.
- Hue file browser is not supported.
- Flink is not supported.
- S3A access and GCS access may not work from the same cluster without configuration changes, because there are overlapping configuration parameter names for GCS and S3.
- Enabling RAZ for GCP is supported only with Runtime 7.2.17 or newer.

GCP requirements for RAZ-enabled GCP environments

The minimal GCS cloud storage setup recommended for production is described in the Minimum setup for cloud storage for GCP. In addition to this, you should create the following service account for RAZ:

Service account name	**Description**	**Required IAM roles or permissions**	**Scope**
Ranger Cloud Access Authorizer	This service account will be used by CDP services to access workload data via RAZ. It provides full access to the data storage location.	Choose one of the following two approaches: - Option A: Use both built-in and custom roles - Option B: Use only a custom role **Option A: Use both built-in and custom roles:** ● Assign the **storage.objectAdmin** built-in GCP role ● Assign the **storage.hmacKeyAdmin** built-in GCP role ● Create and assign a custom role that grants the **storage.buckets.get** permission. **Option B: Use only a custom role:** Create a custom role with the below permissions and assign it: ● storage.buckets.get	*Data storage* bucket For Data Lake backup and restore: *Backups* bucket, if different from the main data storage bucket

		• storage.hmacKeys.create • storage.hmacKeys.delete • storage.hmacKeys.get • storage.hmacKeys.list • storage.hmacKeys.update • storage.multipartUploads.abort • storage.multipartUploads.create • storage.multipartUploads.list • storage.multipartUploads.listParts • storage.objects.create • storage.objects.delete • storage.objects.get • storage.objects.getIamPolicy • storage.objects.list • storage.objects.setIamPolicy • storage.objects.update	

Use the usual instructions to create GCS buckets, create the service accounts, create custom roles, and add service accounts as members and users to other service accounts. Refer to the instructions in the Minimum setup for cloud storage for GCP.

Note that this additional **Ranger Cloud Access Authorizer** service account mentioned above also needs to be added as a member to the *Data storage* bucket by using the steps described in Add service accounts as members to buckets.

Registering a RAZ-enabled GCP environment

You can use the CDP web interface or Beta CDP CLI to register a RAZ-enabled GCP environment.

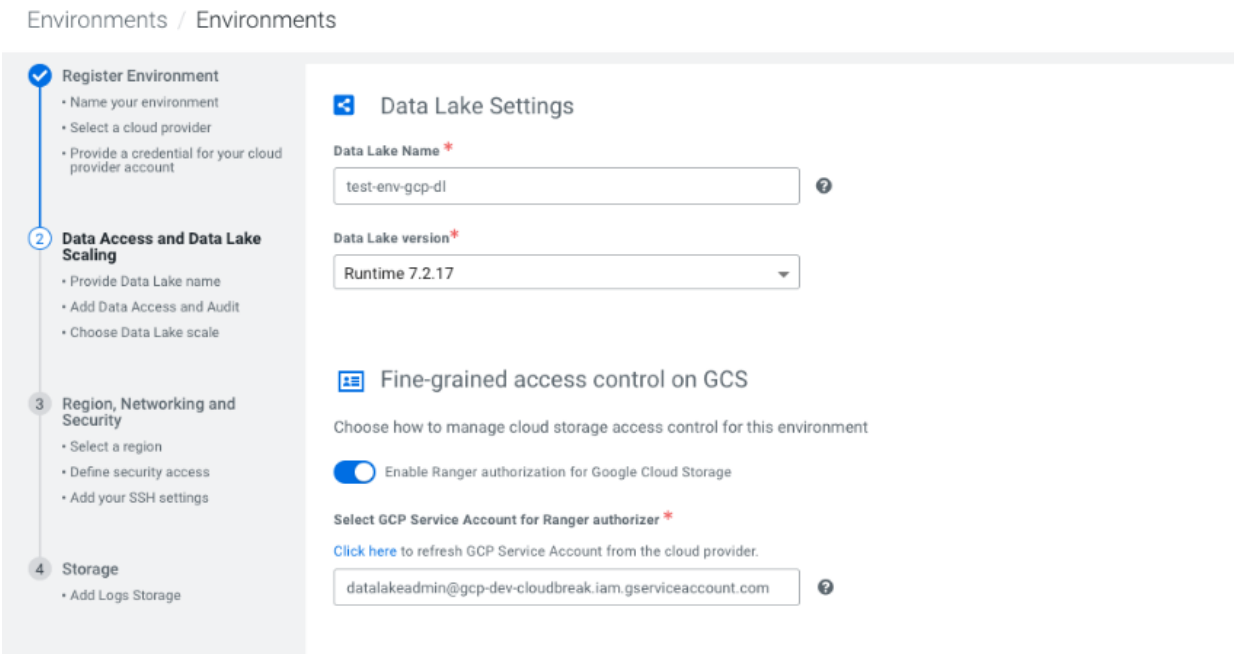
You can enable RAZ on the latest available version of Cloudera Runtime. The minimum version supporting RAZ for GCP environments is Cloudera Runtime 7.2.17.

Using CDP UI to register RAZ-enabled GCP environment

Follow the usual instructions to register a GCP environment in CDP. Make sure to do the following:

1. Select GCP as the cloud provider and then provide the necessary details.
2. Make sure to select the runtime version 7.2.17 or newer.
3. To enable Ranger RAZ authorization, click on **Enable Authorization for Google Cloud Storage** and then select the **Ranger Cloud Access Authorizer** service account (that you

created earlier) from the dropdown:



4. Continue registering the environment as usual.

Using Beta CDP CLI to register RAZ-enabled GCP environment

RAZ for GCP is supported via Beta CLI. Below are the steps to configure RAZ via Beta CDP CLI:

1. Install Beta CLI. See Install Beta CDP CLI.
2. Create a GCP environment with CDP CLI by using the `cdp environments create-gcp-environment` command. For more information, see Register a GCP environment from CDP CLI.
3. Use the following command to set empty IDBroker mappings for the environment. Since RAZ is to be configured, `ranger-cloud-access-authorizer-role` also needs to be set. Below is a CLI command example:

```
Unset
cdp environments set-id-broker-mappings \
--environment-name myraz-gcp-env \
--data-access-role
datalakeadmin@gcp-dev-cloudbreak.iam.gserviceaccount.com \
--set-empty-mappings \
--ranger-audit-role
rangeraudit@gcp-dev-cloudbreak.iam.gserviceaccount.com \
```

```
--ranger-cloud-access-authorizer-role
datalakeadmin@gcp-dev-cloudbreak.iam.gserviceaccount.com
```

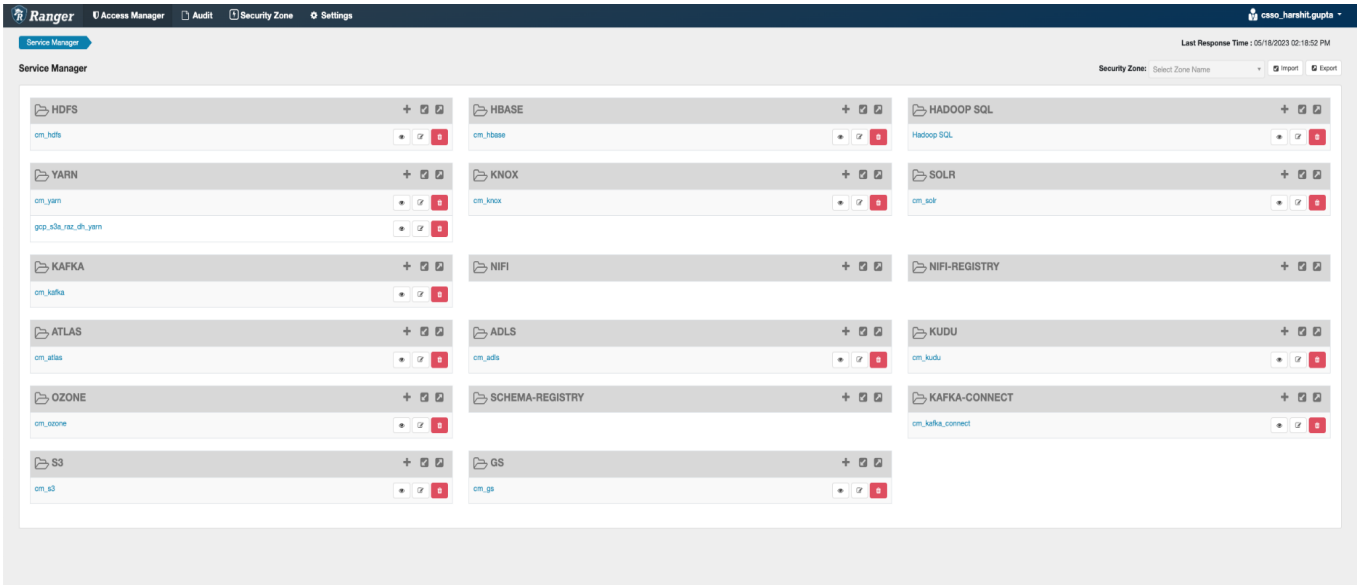
4. Create RAZ enabled Data Lake using Beta CLI. Note that the `--enable-ranger-raz` option is provided to enable RAZ. Runtime version should be greater than or equal to 7.2.17. Below is a CLI command example:

```
Unset
./cdp.sh datalake create-gcp-datalake \
--environment-name myraz-gcp-env \
--datalake-name myraz-gcp-dl \
--runtime 7.2.17  \
--cloud-provider-configuration
"serviceAccountEmail=idbroker@gcp-dev-cloudbreak.iam.gserviceacco
unt.com,storageLocation=gs://perf-team-west2-bucket" \
--enable-ranger-raz --scale LIGHT_DUTY
```

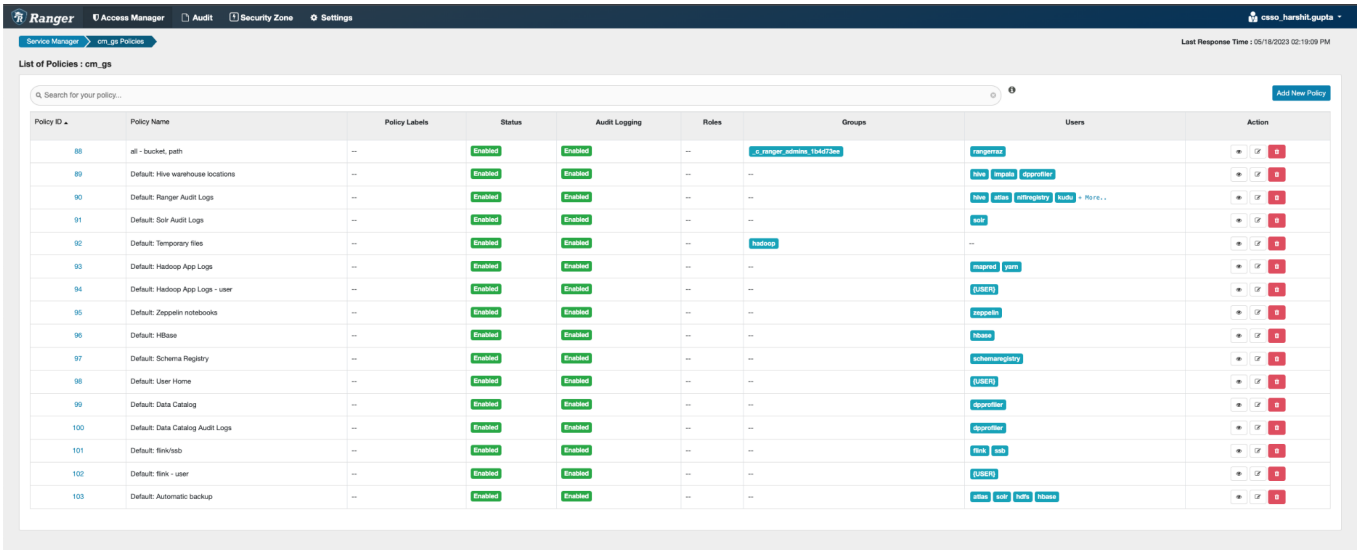
5. Once Data Lake is created successfully, log in to Cloudera Manager web UI and verify that ranger-RANGER_RAZ server has been configured.

Ranger policy options for RAZ-enabled GCP environment

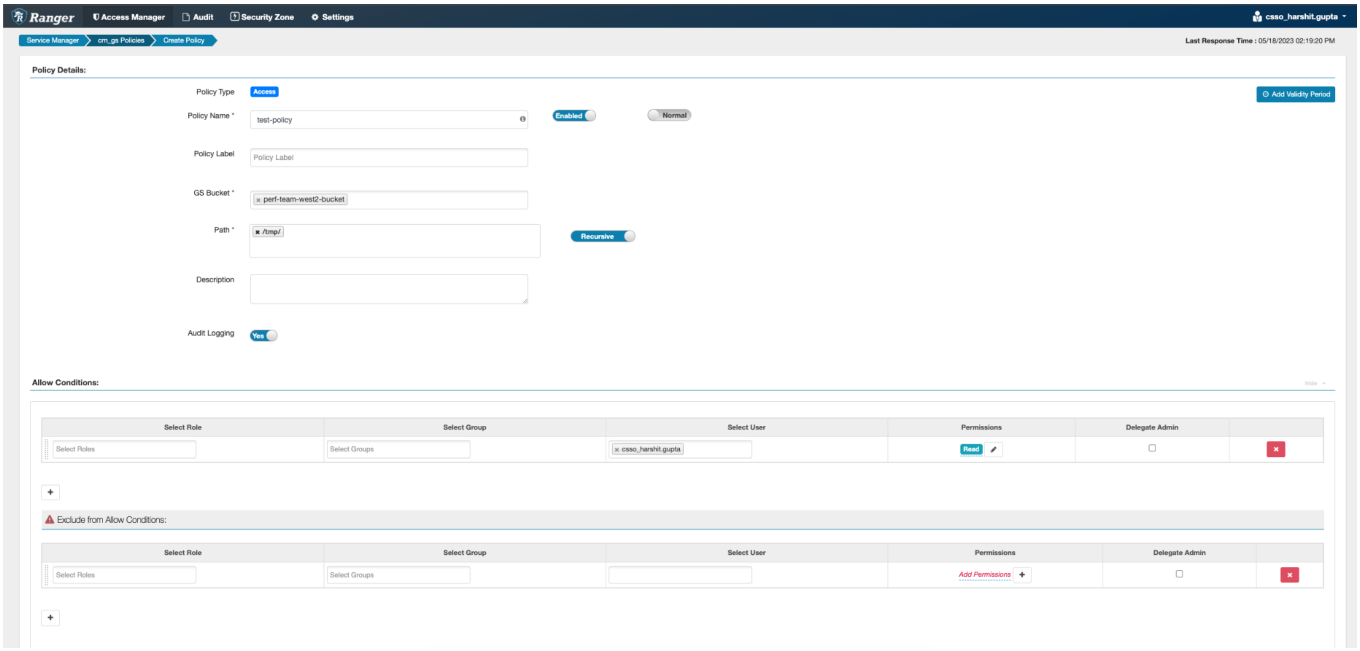
The default Ranger policies for GCS are located under the **GS Tab (ServiceDef)** in the **cm_gs repo:**



This is the default view of the **cm_gs** repo with the default Ranger policies specified:



The Ranger policy creation works similar to that of RAZ for S3. You just specify the GCS bucket name, the path, and the policy parameters:



Troubleshooting

Refer to this section to learn about errors that you may encounter and steps to resolve them.

Creation of Kudu table is failing

Problem: The Kudu service user is not authorized to access Hive warehouse locations on cloud object stores, which under certain conditions can prevent Kudu tables from being created. This results in the following error:

```
ImpalaRuntimeException: Error creating Kudu table 'default.truckspeedevents'
CAUSED BY: NonRecoverableException: failed to create HMS catalog entry for table
[id=b764bceb167746b7bb3dc1e8722e66e6]: failed to create Hive MetaStore table:
TException - service has thrown: MetaException(message=Got exception:
java.nio.file.AccessDeniedException
```

Workaround: Add "kudu" service user to the allow list for "Default: Hive warehouse locations" in the `cm_gs` Ranger repository.

RAZ fails to authorize request

Problem: RAZ fails to authorize a request with the following error.

StackTrace on Client side:

hdfs dfs -ls gs://perf-team-west2-bucket/

...

CLOUD_PROVIDER_ERROR:
org.apache.ranger.raz.s3.lib.exceptions.RCSUECloudProviderException: Could not determine region for bucket: [perf-team-west2-bucket]

org.apache.ranger.raz.s3.lib.signer.util.ExceptionUtils.createUnableToFindRegionException(ExceptionUtils.java:29)

org.apache.ranger.raz.processor.s3.signer.S3RemoteSigner.getRegion(S3RemoteSigner.java:217)

org.apache.ranger.raz.processor.s3.signer.S3RemoteSigner.getAuthorizationRequest(S3RemoteSigner.java:379)

org.apache.ranger.raz.processor.s3.S3RazProcessor.getRangerAccessRequests(S3RazProcessor.java:183)

org.apache.ranger.raz.processor.s3.S3RazProcessor.preProcess(S3RazProcessor.java:120)

org.apache.ranger.raz.RangerRemoteAuthorizer.authorize(RangerRemoteAuthorizer.java:151)

org.apache.ranger.raz.rest.AuthzREST.authorizeAccess(AuthzREST.java:98)
ls: gs://perf-team-west2-bucket/:
org.apache.ranger.raz.hook.s3.RazS3ClientCredentialsException:

....

23/05/16 08:10:50 INFO impl.MetricsSystemImpl: Stopping s3a-file-system metrics system...

23/05/16 08:10:50 INFO impl.MetricsSystemImpl: s3a-file-system metrics system stopped.

```
23/05/16 08:10:50 INFO impl.MetricsSystemImpl: s3a-file-system metrics system shutdown complete.
```

StackTrace on RAZ server side:

```
com.amazonaws.SdkClientException: Unable to load AWS credentials from any provider in the chain:
[org.apache.ranger.raz.processor.gs.RazGSCredentialProvider@4eeb9a91: RazGSCredentialsProvider.java: Exception while creating HMac Keys in GCP. If service account HMAC key limit reached, manually remove the key from GCP console and restart the service.

Root cause: { com.google.cloud.storage.StorageException: Exception from GCP: Service account HMAC key limit reached }]

com.amazonaws.auth.AWSCredentialsProviderChain.getCredentials(AWSCredentialsProviderChain.java:136)

org.apache.ranger.raz.s3.lib.utils.S3RegionUtils.lookupRegion(S3RegionUtils.java:118)

org.apache.ranger.raz.processor.s3.signer.S3RemoteSigner.getRegion(S3RemoteSigner.java:212)

org.apache.ranger.raz.processor.s3.signer.S3RemoteSigner.getAuthorizationRequest(S3RemoteSigner.java:379)

org.apache.ranger.raz.processor.s3.S3RazProcessor.getRangerAccessRequests(S3RazProcessor.java:183)

org.apache.ranger.raz.processor.s3.S3RazProcessor.preProcess(S3RazProcessor.java:120)

org.apache.ranger.raz.RangerRemoteAuthorizer.authorize(RangerRemoteAuthorizer.java:151)

org.apache.ranger.raz.rest.AuthzREST.authorizeAccess(AuthzREST.java:98)
```

Workaround: As mentioned in the stack trace of server logs above, manually remove the key from the GCP console and restart the RAZ service.