Machina User Documentation

GOOGLE CLOUD'S EXTERNAL KEY MANAGER INTEGRATION GUIDE

If you have comments about this documentation, submit your feedback to:

Documentation@ionicsecurity.com

TABLE OF CONTENTS

POLICY

1: OVERVIEW

Machina gives you the ability to enforce powerful data access policy with just a few lines of code, providing a consistent, seamless way to assure the ongoing security of data underlying your applications.

This document provides instructions for using Machina data protection keys and policy controls inside Google Cloud Platform via the Cloud External Key Manager (Cloud EKM) integration.

Topics

2: BEST PRACTICES

Machina gives you the ability to enforce powerful data access policy with just a few lines of code, providing a consistent, seamless way to assure the ongoing security of data underlying your applications. This guide will provide best practice advice for using Machina data protection keys and policy controls inside GCP (Google Cloud Platform) via the EKM (external key management) integration.

Topics

Performance & Reliability

Currently, the Machina integration offers a 99.99% resiliency. Successful requests rely on the resilience of the connector, the web service, and the key service. The services are fully managed by Ionic and do not require any configuration by customers beyond the steps mentioned in the "Machina Documentation- Ionic Machina / Google External Key Manager: Deployment Guide."

- The connector portion of the integration is a stateless web service with multiple service replicas running behind a highly available load balancer in the GCP us-west1 region.
- Machina runs on managed cloud infrastructure with independent application layer replicas and clustered databases, all deployed across multiple regions and cloud providers. Machina is designed to handle the loss of an entire region or cloud provider and remain operational. Region and datacenter level problems are mitigated by steering traffic via the DNS record for api.ionic.com, which has a TTL of 120 ms.
- The key service tier runs separately from the web tier. Like the web tier, this tier is configured with independent application layer replicas and clustered databases. Although Ionic-managed key servers are currently limited to a single region; they utilize multiple availability zones.

While other components of Machina are multi-region and redundant, our Google External Key Manager integration is deployed to the us-west1 region only. Please use this integration with GCP Projects and KMS keys in the us-west1 region and

avoid using Machina-backed External Key Manager keys with multi-region services until additional regions are supported. If this poses problems for your use case, please reach out to Ionic support to let us know.

For additional data regarding the most up to date GCP KMS locations, please refer to https://cloud.google.com/kms/docs/locations. Note that this information is subject to change so please review periodically.

Policy construction

The instructions in the "Machina User Documentation - Ionic Security / Google External Key Manager Deployment Guide" provides you with a starting set of policies for controlling access to the keys used to protect resources in GCP projects.

Additional policies can be authored to restrict access to certain keys, e.g. based on time of day, or to prevent access to a key after a given date. You can author these policies and scope them to individual keys, but we recommend instead authoring policies that control data use based on key attributes, adding these attributes to Machina keys as they are created.

Attributes on Machina keys

In this integration, each Machina key is mapped to a single GCP KMS CKV. To ease management, we recommend adding a common set of attributes to all Machina keys which will be used for CKVs of the same CK.

Additional attributes can be added to the individual Machina keys mapped to each CKV to allow different versions of a key to be managed differently via Ionic Machina policy.

Key Granularity

Machina data key attributes make it simple to enforce different access controls for data protected with different keys. If your needs require more granular and varied

controls, we recommend creating more GCP KMS CKVs mapped to different Machina keys.

3: INTEGRATING GOOGLE CLOUD'S EKM WITH MACHINA

Prerequisites

To configure External Key Manager keys, you need the following:

- An existing or new Google Cloud project. To create a new Google Cloud account and project, visit: https://cloud.google.com/Google Cloud/
- A Machina account. To create a new account, visit https://ionic.com/start-for-free/, or reach out to Ionic Customer Success team via our contact form at: https://support.ionic.com/hc/en-us/requests/new. Additionally, if you need specific Enterprise sales agreements you can reach out to Sales@IonicSecurity.com for assistance.

Steps

Step 1: Create or identify Google Cloud account and Ionic tenant

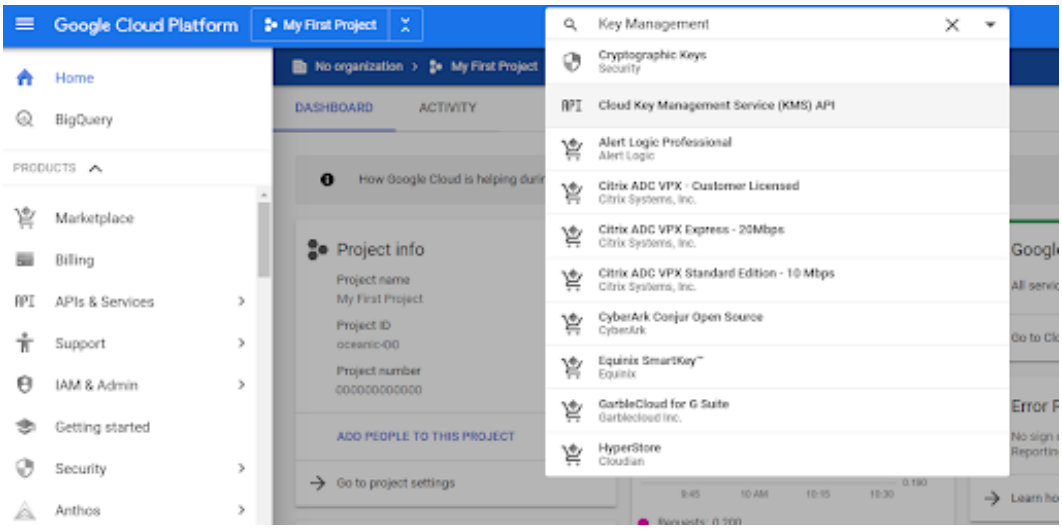
You can configure Cloud EKM keys for multiple Google Cloud projects using a single Machina instance. You are not required to create a new Machina instance for each Google Cloud project.

Google Cloud services integrating with External Key Manager include BigQuery (BQ) and Google Compute Engine (GCE), so you will need to use those services to exercise External Key Manager capabilities. We require launching any services integrating with External Key Manager in the us-west1 region. If you need other regions enabled please contact an Ionic Sales Representative (Sales@IonicSecurity.com).

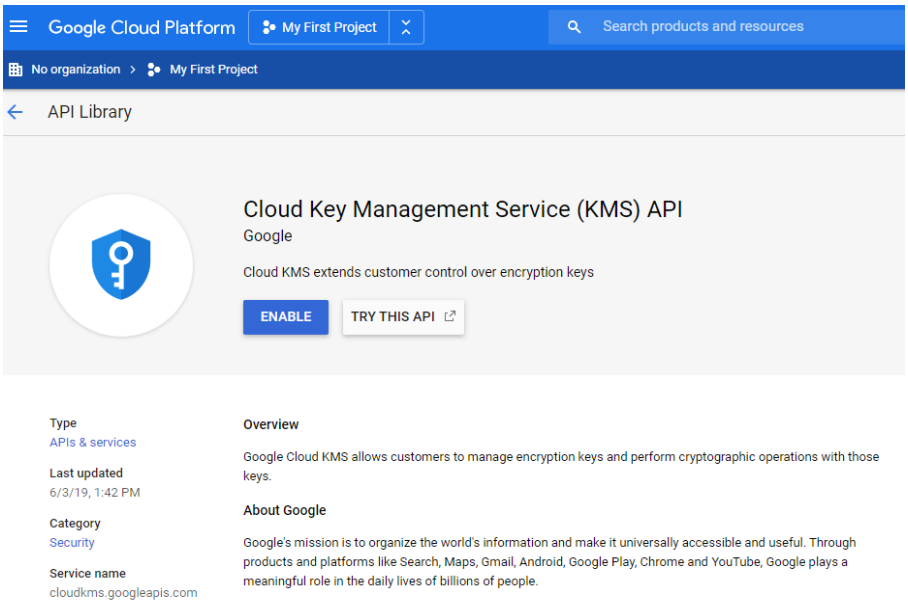
Step 2: Enable Google Key Management Service (KMS) API

To take advantage of the Machina External Key Management integration with the Google Cloud Platform, first enable the Google EKM API.

1. Log into Google Cloud Project.
2. Navigate to the Google Cloud Key Management Service API page. You can do this via a simple search of KMS in the GCP search.



3. Select Enable Cloud Key management Service (KMS) API



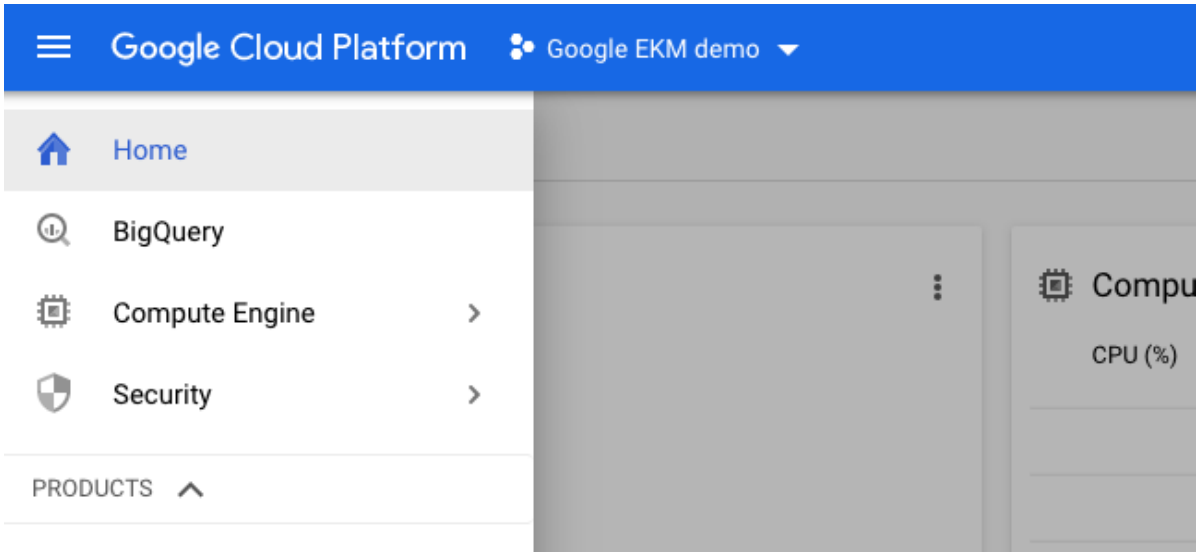
For more information on Google KMS, please visit: https://cloud.google.com/kms/

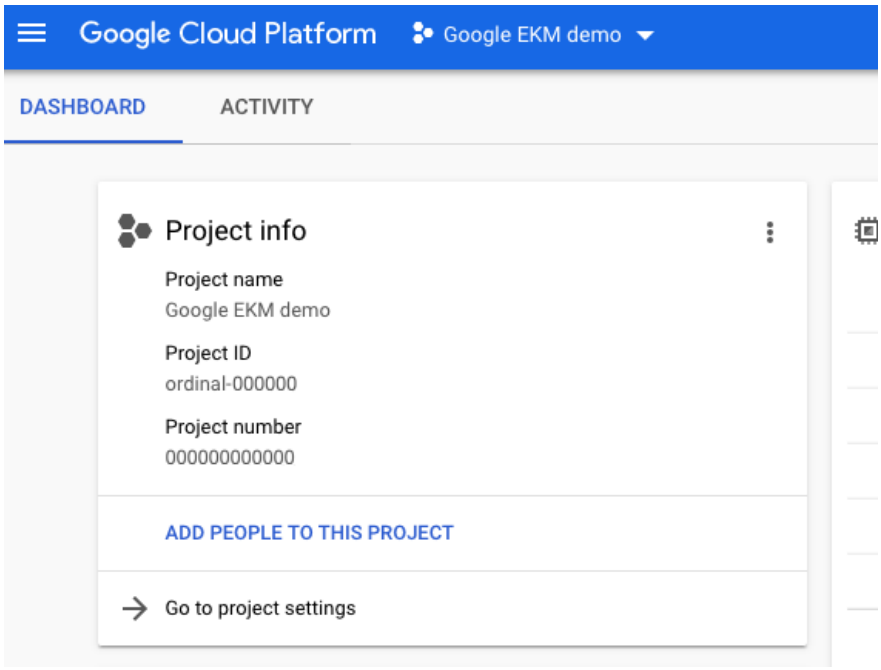
Step 3: Locate the Google Cloud Project Number

Once the Google KMS API is enabled in the Google Cloud project, capture the Google Project Number. This will be required to integrate Ionic Machina and Google KMS. Behind the scenes, a Google service account is automatically generated. This service account will follow this format:

"service-[PROJECT-NUMBER]@gcp-sa-ekms.iam.gserviceaccount.com"

Note that [PROJECT-NUMBER] is the project number for your Google Cloud project, which can be found in the Google Cloud Web Console on the "Home" tab under the "Project Info" section.
You will need the full Google Service Account information shortly in order to complete the integration with Ionic Machina.





Note: You will need the Google Service Account information for Section 5-1

Warning: The Google Service Account being used in this integration must follow the format as outlined above and references a hidden Service Account. There are additional Service Accounts that can be found in the IAM section of GCP. Do not use any service accounts located there in this exercise.

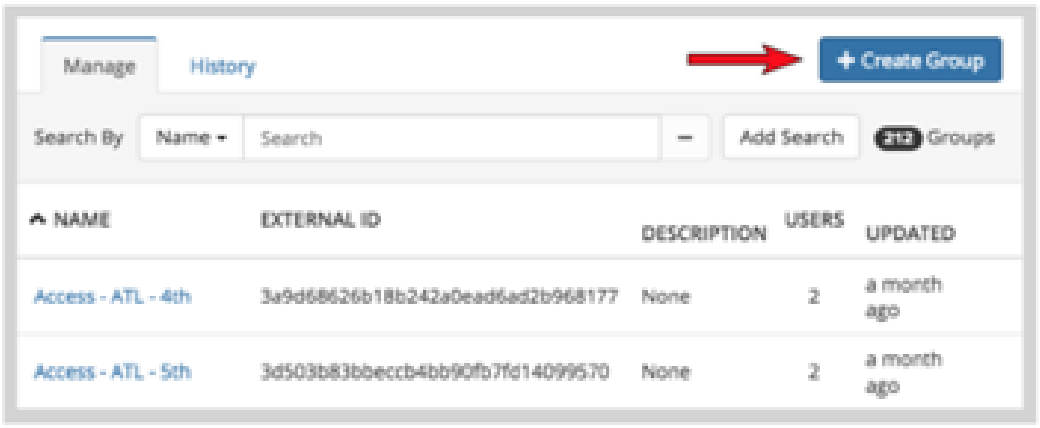
Step 4: Machina Instance Configuration

As mentioned earlier, a Machina instance is required to manage external policy controls over your Google Keys. Below are example Machina configurations regarding Users, Groups and Policies that will enable you to get started with the Machina Google EKM integration. Refer to the Machina Console Admin User Guide for more detailed information.

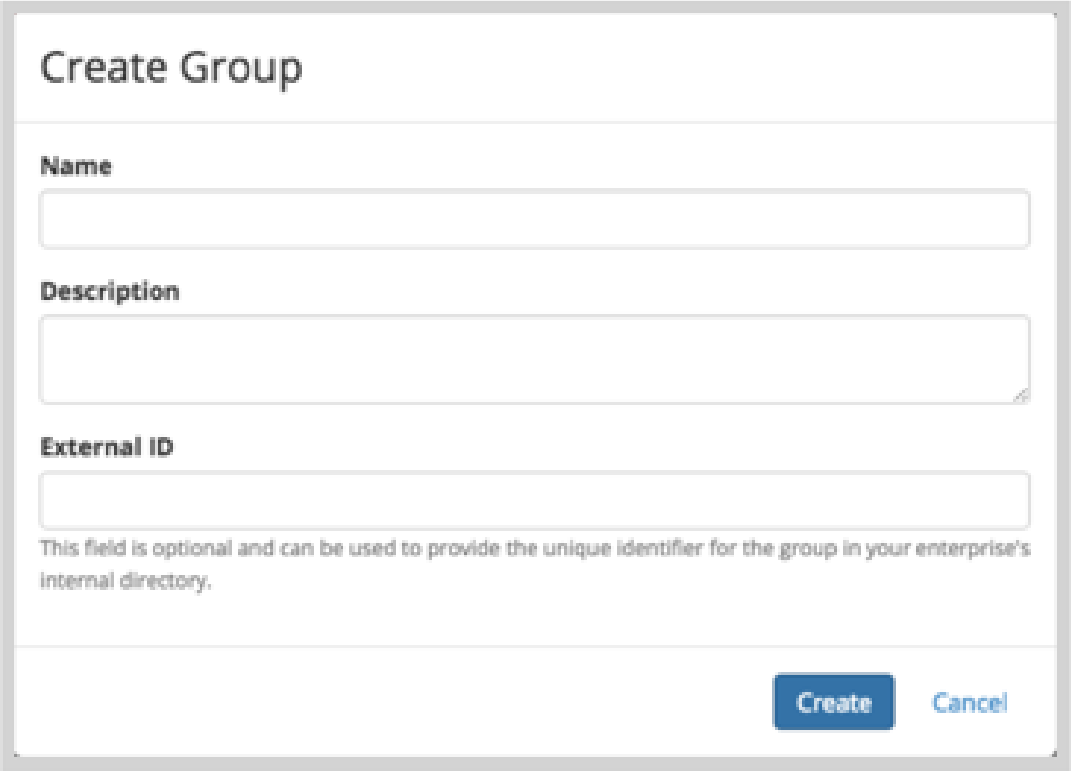
Create the EKM Service Users Group

For more detailed information regarding Group creation in the Machina Console, check our Knowledge Base article on Creating new User Groups.

1. Log in to your Machina Console.
2. Navigate to Groups > Manage tab, click +Create Group.



3. Enter the Name of the group: "EKM Service Users".



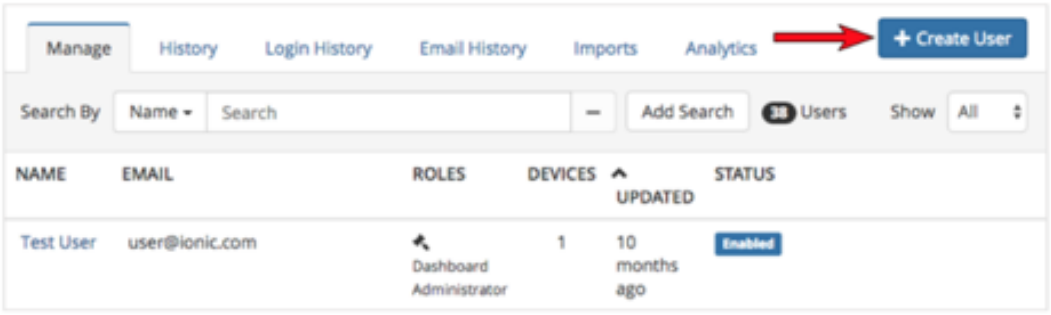
4. Click Create.

Create the EKM Service User

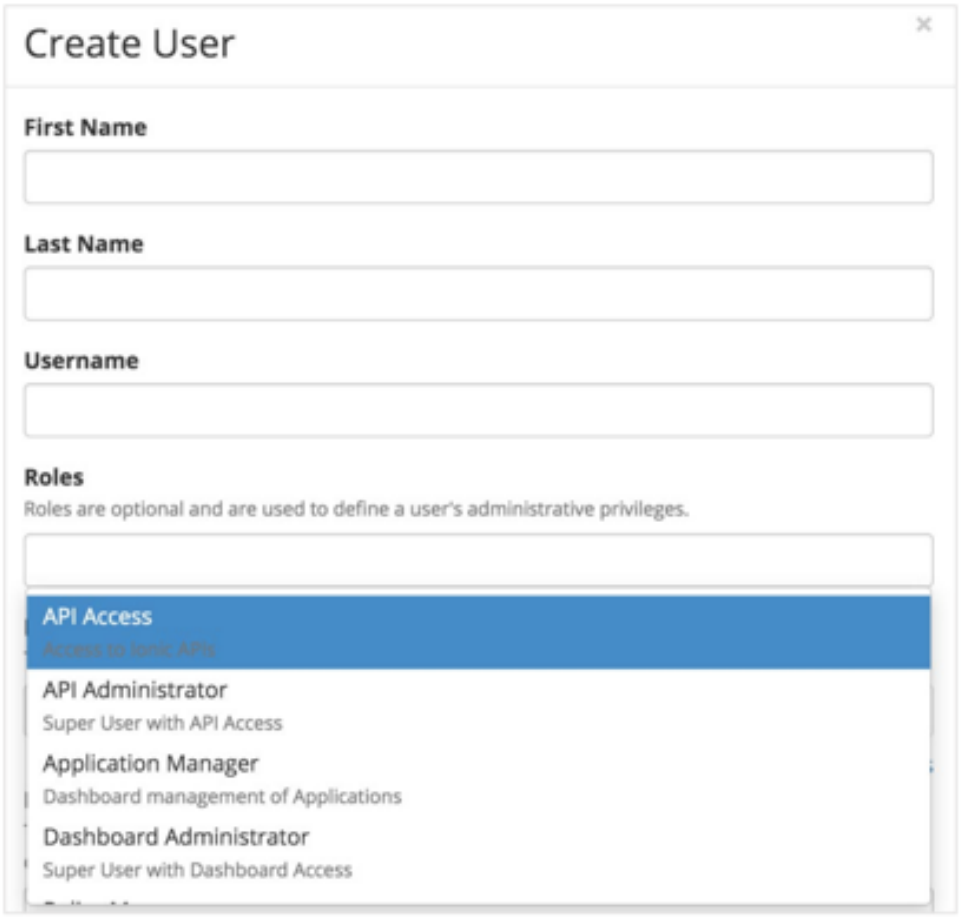
This is a Machina Service User account that is used by the Machina EKM integration to request keys. This must be a separate account from any existing user account(s)

1. Log into the Machina Console.
2. Navigate to the Users Section.

3. On the Users > Manage tab, click +Create User.



4. Enter "EKM Service" into the Username field.



5. Enter an email address for this service user account. The email address must be unique in this Machina instance. You will use this email address in the enrollment process below.

   The EKM Service user must have API Access credentials.

6. Add this user to the "EKM Service Users" group. Type the Group name into the Group field and select it when it appears.

Create the Google Cloud Service User

This Ionic account corresponds to the Google Cloud Service Account (see 1-3) that makes requests to wrap and unwrap keys from Google Cloud KMS. Before beginning this step, have the Google Cloud Service User ID. It is in the form of "service-<GC-PROJECT-NUMBER>@gcp-sa-ekms.iam.gserviceaccount.com".

1. Log into the Machina Console.
2. Navigate to the Users Section.
3. On the Users > Manage tab, click +Create User.
4. Enter "Google Cloud Service User" into the Username field.
5. Enter the Google Cloud Service User ID into the External ID field.
6. Click Create.

Create Application Policies

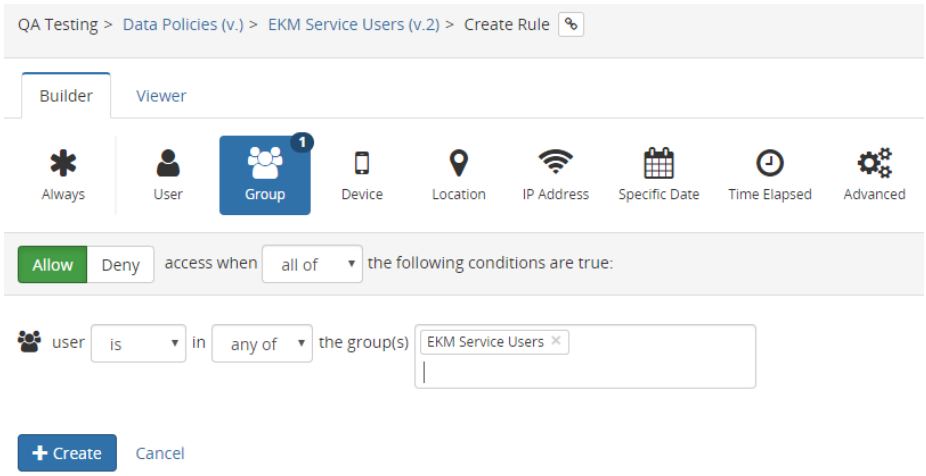
The application policies in this document are trivially simple in order to get up and running. In a production example more complex policies involving User Groups and additional key attributes will be set up to provide highly granular access to critical data. See Creating a new Data Policy in the Machina Console and Creating new Data Markings in the Machina Console for more information on setting up policies.

The first policy allows users in the EKM Service Users Group to access specific data in the Google Cloud project. This policy is required because the EKM Integration acts as an Ionic Service User and supports delegated requests from Google Cloud Service Accounts.

Create EKM Service User Group Policy

1. Log into the Machina Console.
2. Navigate to the Policies section.
3. Click +Create Data Policy.
4. Enter "EKM Service Users" as the name. Click +Create.

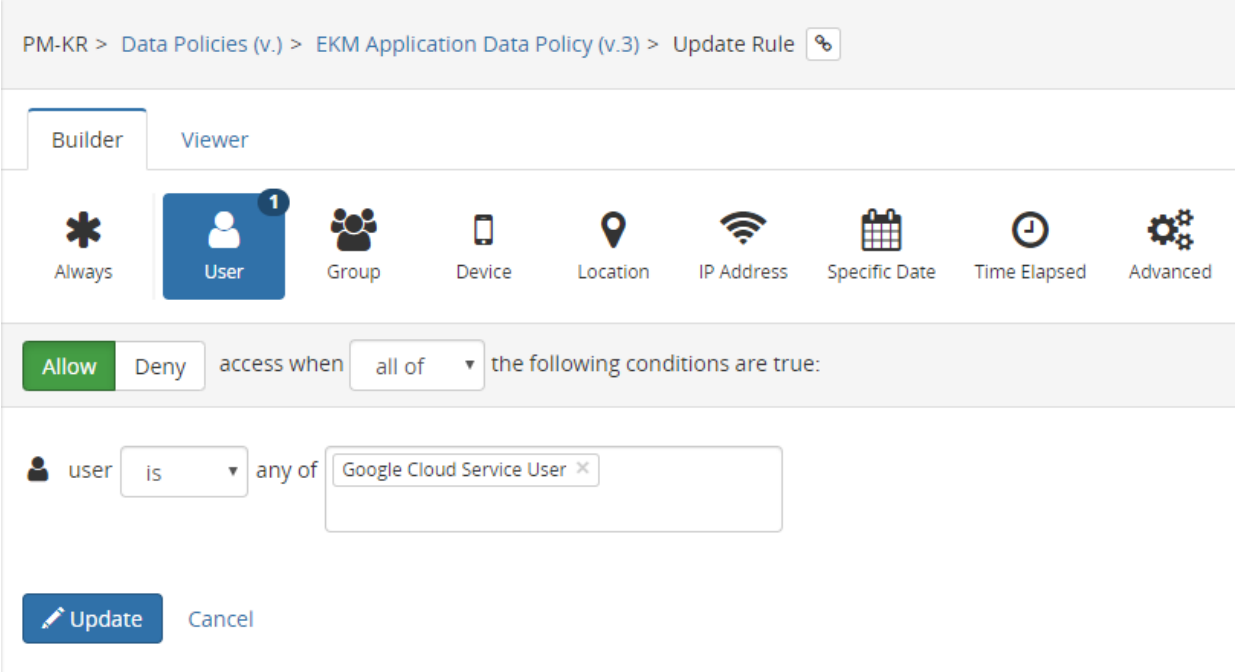
5. Click "Enable Data Policy" and click Enable on the popup.
6. Click +Create Rule.
7. Click on the Group button and click "Specific groups" in the dropdown list.
8. Click in the Groups field and select "EKM Service Users".
9. Click +Create.



Create Google Cloud Service User Policy

The next policy allows specific users to perform EKM operations. This policy is simplistic for demonstration purposes. Actual policies should include security attributes to best protect your applications and data. For an example on using Attribute Based Access Controls (ABAC), see "Using Attribute-Based Access Controls" on page 25.

1. Log into the Machina Console.
2. Navigate to the Policies Section.
3. Click +Create Data Policy.
4. Enter "EKM Application Data Policy" as the name. Click +Create.
5. Click "Enable Data Policy" and click Enable on the popup.
6. Click +Create Rule.
7. Click on the User button and click "Specific user" in the dropdown list.
8. Click in the User field and select "Google Cloud Service User".
9. Click +Create.



Step 5: Complete device registration and key creation

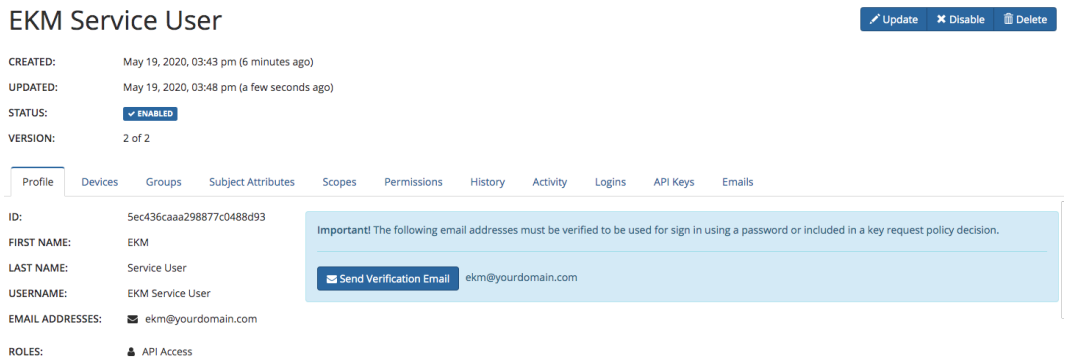
In order for GCP and Machina to appropriately communicate with each other, we need to establish the explicit connection between your Machina profile and the Google Service Account. This is accomplished by relating the Machina secure enrollment profile with the Google Project.

Reset password

Using the EKM Service User account created in the Machina console, please reset your password.

1. Log into the Machina Console.
2. Select Users.
3. Select EKM Service User.

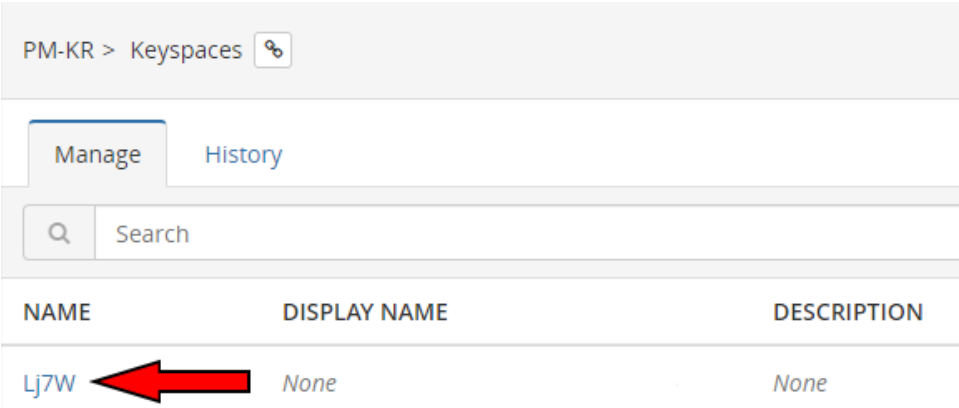
4. Select Send Verification Email.



5. When you receive the verification email, click the link and navigate to the Machina console.
6. Select Forgot Password. (Note: Confirm you are resetting the correct account password)
7. Update your password information.

Create a new device profile

Using the Machina CLI tool (https://dev.ionic.com/tools), create a new device profile for "EKM Service User" with the command below, replacing "ABcd" with the keyspace id associated with your Machina instance. You can find the keyspace in the Machina Console under the Keyspaces tab.



This example shows the use of the general option "--devicefile" to specify the file you'd like to save your Secure Enrollment Profile (SEP) in and the "--devicetype" option to save it as a password protected file. You can then use this SEP file in later commands like creating keys, fetching keys, encrypt and decrypting files, etc. Also, for more information regarding creating device SEPs please refer to our developer site: https://dev.ionic.com. There you can find more information and examples to explore.

```
machina \

    --devicetype password \

    --devicepw <your_sep_password> \ ← Create profile
password here

    --devicefile profiles.pw \

    profile enroll \

    --keyspace <keyspace> \

    --pass <your_machina_password> \ ← The Machina Console
password here

    --email <your_email_address> \

    --type idc
```

*Note: --devicepw has a default minimum length of 6 characters.

This new device profile will be uploaded to the EKM integration to facilitate delegated requests with "Google Cloud Service User" as the delegator and "EKM Integration User" as the delegatee.

Upload the new device profile

1. Once you have created your SEP, submit a ticket to Ionic Customer Success requesting configuration of the Google Cloud External Key Manager integration. In the request, include the following:
   - Google Cloud service account id you will use for the integration (from 1-3 above).
   - Attach the Device Profile (the profile is located in the --devicefile path that you identified during the creation step above.
   - Your device profile's password must be shared with CS if creating a password protected profile (recommended)

2. Customer Success will provide you with instructions for delivering the SEP and inform you when the integration is ready for use.

   Please contact CustomerSupport@IonicSecurity.com if you need any assistance.

Step 6: Create Machina keys

In this step, you will create a Machina key for each Google Cloud External Key Manager key. Each Google Cloud External Key Manager key is backed by a Machina key.

1. Create Machina keys using the Machina CLI and the device profile created in step 5.
2. You will need the keyID of this key. Use the following CLI command:

   machina \

   --devicetype password \

   --devicepw <your_sep_password> \ ← Create profile password here

   --devicefile profiles.pw \

   key create
3. This will generate output like the following:

```
{
  "keyCount" : 1,
  "keys" : [
    {
      "keyRefId" : "ionicsdk_key",
      "deviceId" : "ABcd.I.00000000-1111-2222-3333-4444",
      "originId" : "ionic-keyserver",
      "keyId" : "ABcdIti1ruA",
    }
  ]
}
```

   The "keyId" in the above output will be used in the next step.

Step 7: Map Google External KMS key to Machina URI

The last step in setting up EKM is to create an association between a Google Cloud CryptoKey and the Ionic Key Encryption Key (KEK) created above.

1. In the Google Cloud Console, navigate to Security -> Cryptographic Keys.
2. Click Create Key Ring and enter the following:
   - Key ring name: EKMTestKeyRing
   - Key ring location: us-west1
3. Click Create to finish creating the key ring.
4. The Create Key page is displayed. Enter the following information:
   - Select "Externally managed key"
   - Key name: TestKEK
   - Key URI: https://ekm.ionic.com/v0/<keyID>. Replace "<keyID>" with the keyID of the key created in step 6.
5. Click Create to finish creating the EKM key.

The key TestKEK can now be used to protect BigQuery data sets and Google Compute Engine images.

A cryptographic key is a resource that is used for encrypting and decrypting data or for producing and verifying digital signatures. Any key can have multiple versions.
Learn more

Project name	Key ring	Location
ionic	EKMTestKeyRing	us-west1

What type of key do you want to create?

- ○ Generated key
  A standard customer managed encryption key. The key material will be generated for you. Learn more
- ○ Imported key
  For importing your key material into GCP. Learn more
- ◉ Externally managed key
  The key material will be stored in an external key manager. Learn more

Key name *
TestKEK

Purpose
Symmetric encrypt/decrypt

Key type and algorithm
External symmetric key

Link your external key

Please note, by using an external key manager, Google has no control over the availability of your key. Google can't recover your data if you lose keys you manage outside of GCP.
Learn more

Complete the following steps to link your external key. Learn more

1. Create an external key in your external key manager.
2. Authorize the Google service account for this key permission to use the external key in your external key manager.
3. Copy and paste the Key URI here.

Key URI *
https://ekm.ionic.com/v0/ABcdGffxnes

CREATE CANCEL

Step 8: Associate an External Machina Key to a new Google Compute Engine (GCE) Table

Prerequisites:

- An existing or new Google Cloud project. To create a new Google Cloud account and project, visit:https://cloud.google.com/Google Cloud/
- A Machina account. To create a new account, visithttps://ionic.com/start-for-free/, or reach out to our Customer Success team via our contact form at: https://support.ionic.com/hc/en-us/requests/new

To configure External Key Manager keys for your GCE virtual instance:

1. In the Google Cloud Console, navigate to Compute Engine.
2. Navigate to VM Instances.
3. Select Create Instance.
4. Complete the configuration form.
5. Remember to select US-West-1 as your region.
6. Select "Management, security, disks, networking, sole tenancy" for advanced options.
7. Select "Disks".
8. Select Customer-managed-key.
9. Enter Key Resource ID.
10. Click Create.

This creates an association with this specific GCE Virtual instance and the individual key with the selected Machina external key. Now you can create an additional instance for your other keys and protect them in the same fashion, individually. Each of these keys can have attributes that were created (gce_ enabled, gce_finance_enabled, gce_hr_enabled, etc) and each of these keys can have related policies and rules to protect them. For more information on creating Machina Policies, please refer to our Policy Training Guide (https://support.ionic.com/hc/en-us/articles/360033383474-Machina-Console-Policy-Training-Guide-08-22-2019). You can also see Policy in Action for details around these specific policies and rules.

4: USING ATTRIBUTE-BASED ACCESS CONTROLS

While the above example includes a bare-bones allow/deny policy to show just how easy it is to set up Machina with Google EKM, one of the major differences between Ionic Machina and other vendors is our ability to control assets at a granular level via attributes.

The policy example below allows specific users to perform operations on distinct tables in BigQuery as well as controlling access to Google Compute Engine based on the associated attributes.

Prerequisites

- Created required users (EKM Service, Google Cloud Service User)
- Created related Group (EKM Service Users Group)
- API Access for EKM Service user
- Device SEP and Google Service Account information submitted and approved by Customer Success

Topics

Step 1: Create Machina keys for each Google Cloud EKM key

In this integration, each Google Cloud External Key Manager key is backed by a Machina key. You will need to create Machina keys for each Google Cloud External Key Manager key with additional Attributes for Big Query tables and Google Compute Engine.

1. Create additional Machina keys using the Machina CLI and the device profile created earlier.
2. You will need the keyID of this key. Use the following CLI command:

```
machina \
--devicetype password \
--devicepw <your_sep_password> \ ← Create profile password
here
--devicefile profiles.pw \
key create --attr GCP:BQ_MyFinance_Data
```

3. This will generate output like the following:

```
{
   "keyCount":1,
   "keys":[
      {
         "keyRefId":"ionicsdk_key",
         "deviceId":"ABcd.I.00000000-1111-2222-3333-4444",
         "originId":"ionic-keyserver",
         "keyId":"ABcdIti1ruA",
         "attrs":{
            "gcp":[
               "BQ_MyFinance_Data"
            ]
         }
      }
   ]
}
```

4. You now have a specific key with attributes that can be assigned to your Big Query Finance table.
5. To show the flexibility of attributes, let's create another key with additional attributes for your Big Query HR table.
6.

```
A new Machina key is required in order to apply new
attributes to the Google CryptoKey. Use the following CLI
command:
machina \
--devicetype password \
```

```
--devicepw <your_sep_password> \ ← Create profile password
here
--devicefile profiles.pw \
key create --attr GCP:BQ_MyHR_Data
```

7. This will generate output like the following Machina Key with GCP:BQ_MyHR_ Data attributes:

```
{
   "keyCount":1,
   "keys":[
      {
         "keyRefId":"ionicsdk_key",
         "deviceId":"ABcd.I.00000000-1111-2222-3333-4444",
         "originId":"ionic-keyserver",
         "keyId":"ABcdLgia1rtK",
         "attrs":{
            "GCP":[
              "BQ_MyHR_Data"
            ]
         }
      }
   ]
}
```

8. You will repeat the above with any number of attributes you require for the individual tables. The flexibility is almost limitless and allows you to correlate your data and applications with elements ranging from tables, key_create timestamps, regional association, etc.
9. Generate a key for your GCE environment with the attribute value "My_GCE_ Instance". These attributes will allow you to create restrictions around your GCE tools.

   machina \

   --devicetype password \

   --devicepw <your_sep_password> \ ← Create profile password here \

   --devicefile profiles.pw \

   key create --attr GCP:My_GCE_Instance

10. This will generate output like the following:

```
{
   "keyCount":1,
   "keys":[
      {
         "keyRefId":"ionicsdk_key",
         "deviceId":"ABcd.I.00000000-1111-2222-3333-4444",
         "originId":"ionic-keyserver",
         "keyId":"ABcdItiwa2A",
         "attrs":{
            "gcp":[
               "GCP:My_GCE_Instance"
            ]
         }
      }
   ]
}
```


Each Machina key created above includes customer-specified attributes for Google Compute Engine and BigQuery. This allows an attribute based access control to be applied to all keys being used in these environments.

Step 2: Map Google External KMS key to Machina URI

Now that you have three separate keys with corresponding attributes (BQ_ MyFinance_Data, BQ_MyHR_Data, and My_GCE_Instance), we can set up controls around your Google applications by creating an association between a Google Cloud CryptoKey and the Ionic Key Encryption Key (KEK) created above. This is very similar to what was done earlier; however, each Google key will have its own Machina key with associated attributes.

To create an externally managed key encryption key

1. In the Google Cloud Console, navigate to Security -> Cryptographic Keys.
2. You can use the same Key Ring created earlier (EKMTestKeyRing).
3. The Create Key page is displayed. Enter the following information:
   - Select "Externally managed key"
   - Key name: BQ_MyFinance_Data_KEK

- Key URI: https://ekm.ionic.com/v0/<keyID>. Replace "<keyID>" with the keyID of the key created in Section 1 step 3.

4. Click Create to finish creating the EKM key. The key BQ_MyFinance_Data_KEK can now be used to protect your BigQuery finance table.

   Note: You may name your Keys whatever you like. We named BQ_MyFinance_Data_KEK for clarity

Repeat the above steps and create BQ_MyHR_Data_KEK and My_GCE_Instance_KEK using the corresponding Machina created key_ids in "Using Attribute-Based Access Controls" on page 25.

Step 3: Associate the Google CryptoKey to a BigQuery Table

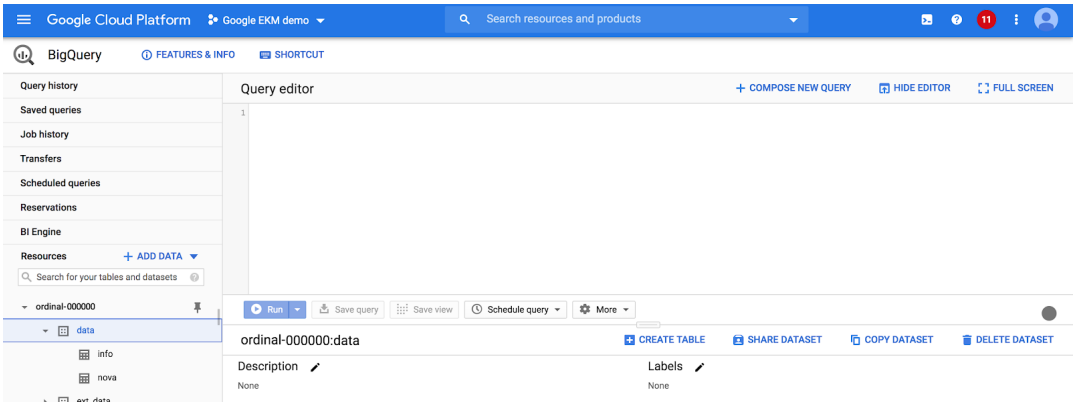
In this step, you will associate an externally managed Google CryptoKey to a new BigQuery table.

Prerequisites:

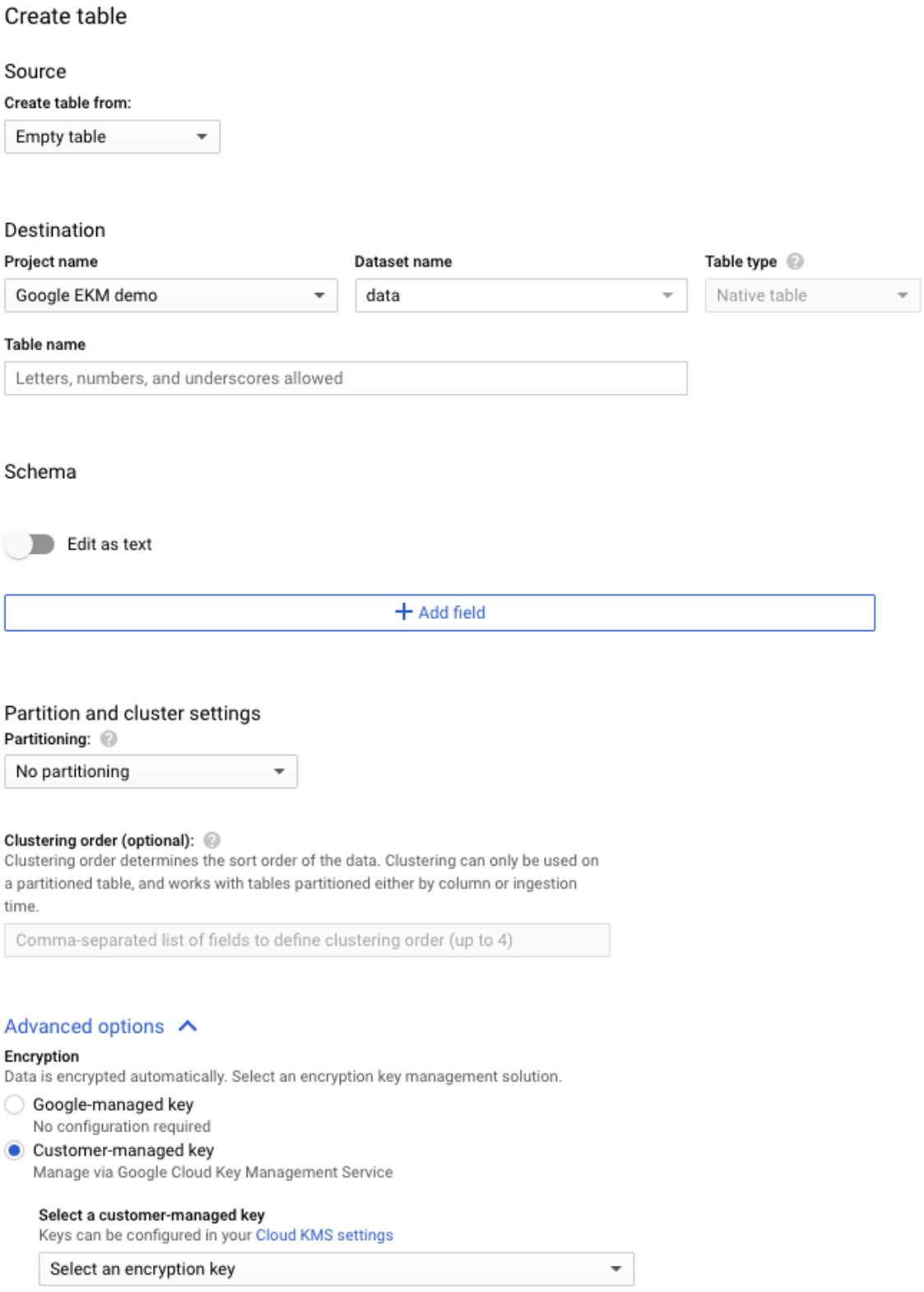
- An existing or new Google Cloud project. To create a new Google Cloud account and project, visit: https://cloud.google.com/Google Cloud/
- A Machina account. To create a new account, visit https://ionic.com/start-for-free/, or reach out to our Customer Success team via our contact form at: https://support.ionic.com/hc/en-us/requests/new
- An existing BigQuery DataSet

To configure External Key Manager keys for your BigQuery Table:

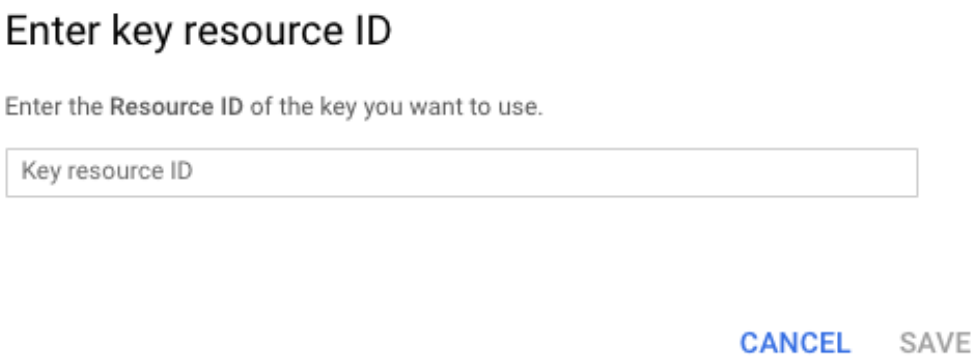
1. In the Google Cloud Console, navigate to BigQuery.
2. Select specific Project.
3. Select specific DataSet.



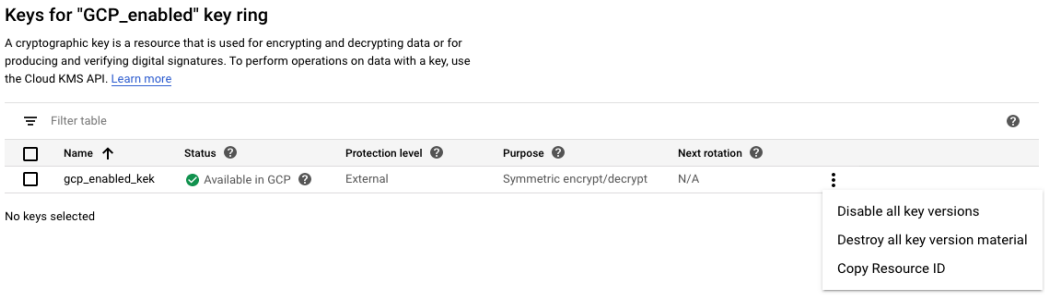
4. Select Create Table.
5. Configure Table with Project, DataSet, Name etc.
6. Select Customer-managed key in order to set your Machina KEK.


Create table
Source
Create table from:
Empty table
Destination
Project name
Google EKM demo
Dataset name
data
Table type
Native table
Table name
Letters, numbers, and underscores allowed
Schema
Edit as text
Add field
Partition and cluster settings
Partitioning:
No partitioning
Clustering order (optional):
Clustering order determines the sort order of the data. Clustering can only be used on a partitioned table, and works with tables partitioned either by column or ingestion time.
Comma-separated list of fields to define clustering order (up to 4)
Advanced options
Encryption
Data is encrypted automatically. Select an encryption key management solution.
Google-managed key
No configuration required
Customer-managed key
Manage via Google Cloud Key Management Service
Select a customer-managed key
Keys can be configured in your Cloud KMS settings
Select an encryption key

7. Set your Resource ID to correlate the Key with this table.



Note: If not present in the drop down, you can find the ResourceID in the Google Cloud Project console. Go to Security -> Cryptographic Keys -> Selecting the KeyRing -> Select the relevant Key -> Capture Resource ID.



Warning: Remember to use the Key Resource ID and NOT the KeyRing Resource ID or CryptoKeyVersion.

8. Select Create Table.

This creates an association with this specific table and the individual key with the bq_enable_kek key. Now you can create an additional table for your other keys and protect them in the same fashion, individually. Each of these keys have attributes that were created (My_GCE_Instance, BQ_MyFinance_Data, BQ_MyHR_Data) and each of these keys can have related policies and rules to protect them. For more information on creating Machina Policies, please refer to our Policy Training Guide (https://support.ionic.com/hc/en-us/articles/360033383474-Machina-Console-Policy-Training-Guide-08-22-2019).

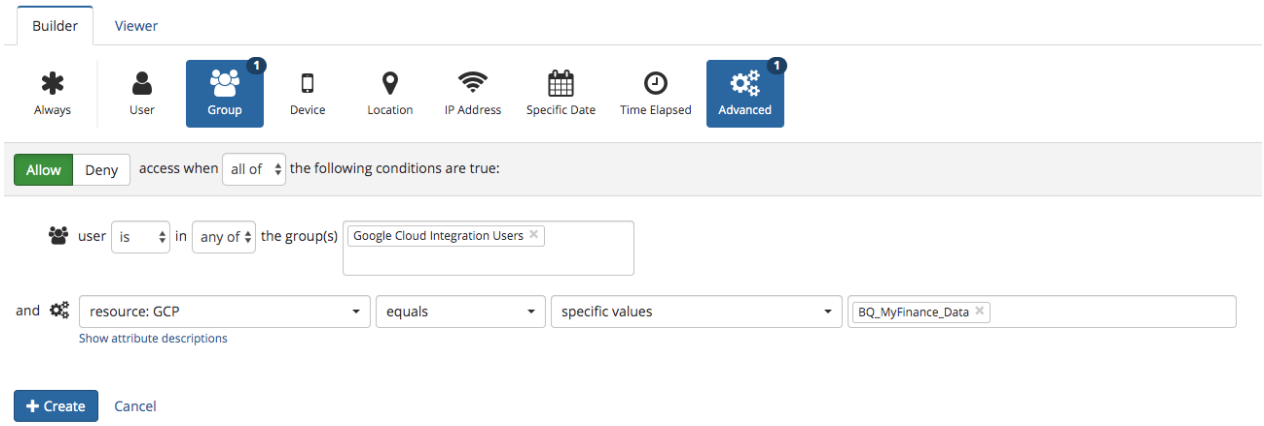
You can also see "Creating an attribute-based EKM service user policy" on page 33

5: CREATING AN ATTRIBUTE-BASED EKM SERVICE USER POLICY

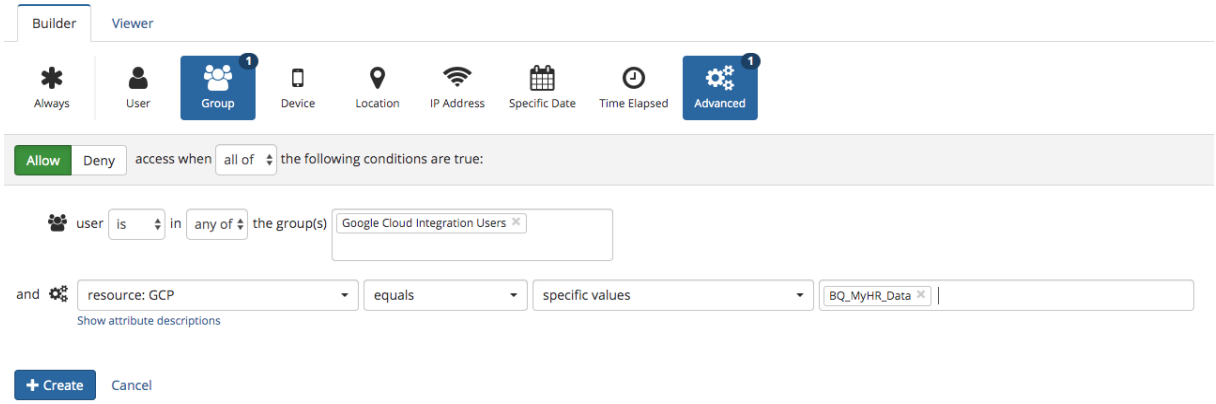
The application policies in this section will focus on specific attributes to allow/deny access to Google Cloud Project elements within BigQuery and Google Compute Engine. For more information and more complex policies involving User Groups and additional key attributes see Creating a new Data Policy in the Machina Console and Creating new Data Markings in the Machina Console .

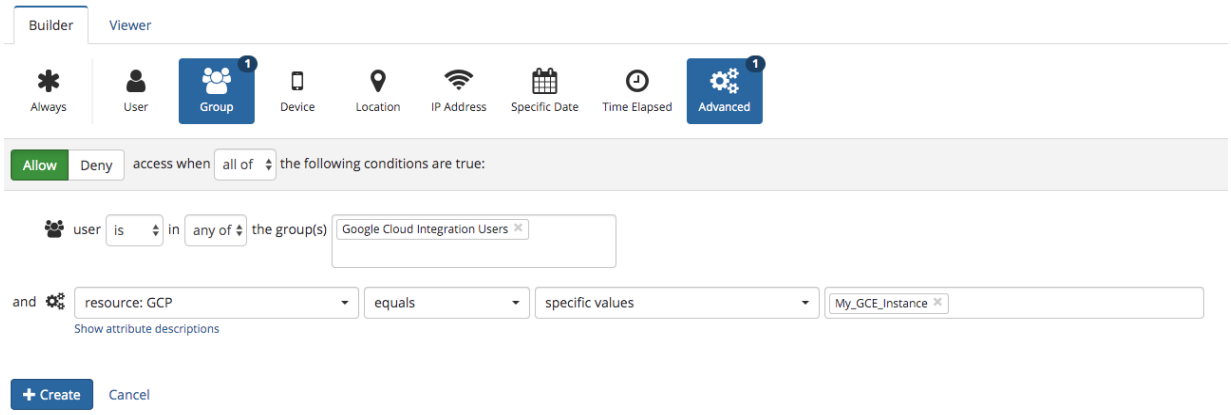
This policy allows users in the EKM Service Users Group that was created in Section 3 to access specific data in a singular BigQuery table, called "Finance."

1. Log into the Machina Console.
2. Navigate to the Data Policies section.
3. Click +Create Data Policy.
4. Enter "allow finance access" as the name. Click +Create.
5. Click "Enable Data Policy" and click Enable on the popup.
6. Click +Create Rule.
7. Click on the Group button and click "Specific groups" in the dropdown list.
8. Click in the User field and select "Google Cloud Service User".
9. Click Advanced.
10. Insert Attribute name 'GCP.
11. Select 'Resource' as the attribute category when prompted.
12. Select 'equals' from the dropdown.
13. Select 'Specific values' from dropdown.
14. Insert value 'BQ_MyFinance_Data'.
15. Click +Create.



You now have an attribute-based policy that, when enabled, allows access only to members of the specified group to the specific BigQuery table that is protected by the specific key with associated attribute of BQ_MyFinance_data.
You can create similar policy rules around the attributes included in the other Machina Keys created earlier for BQ_MyHR_Data, My_GCE_Instance or any other attributes you wish to include in the Machina Key.




Builder
Viewer
Always
User
Group
Device
Location
IP Address
Specific Date
Time Elapsed
Advanced
Allow
Deny
access when
all of
the following conditions are true:
user
is
in
any of
the group(s)
Google Cloud Integration Users
and
resource: GCP
equals
specific values
My_GCE_Instance
Show attribute descriptions
+ Create
Cancel