Qlik Cloud Monitoring Applications Installation Guide

Step-by-step instructions to install and configure the Qlik Cloud Monitoring applications.

TABLE OF CONTENTS

ⓘ *The Qlik Cloud Monitoring Apps Install Guide and applications are provided as-is. They are not covered under the* Qlik Support Policy*. No information from the apps is provided to Qlik. It is recommended to always use the latest version of the apps.*

INTRODUCTION

This guide breaks down the configuration installing a Qlik Cloud monitoring application step-by-step. All Qlik Cloud monitoring applications follow the same setup process and leverage the same data connection. Once this process has been completed for a single monitoring application, it can be reused across all others. The last section of the guide which covers any modifications that might need to be done at the script level might vary by application and is documented as such.

The current Qlik Cloud monitoring applications can be found here.

Connect with us!

Subscribe to the Qlik Support Updates Blog

- Get up to date information on new releases.
- Stay informed about current product insights and new features.

>> click here <<

Join Qlik Community

- Connect with an ever-growing audience of active Qlik enthusiasts.
- Share Tips & Tricks and best practices.

>> click here <<

Expand your Knowledge

- Access an evolving database of Qlik Product knowledge
- Get in touch with our world-class Support.

>> click here <<

ℹ *Note that there is a separate installation guide for the **OEM Dashboard** monitoring application. The installation and configuration of that application is not covered within this guide.*

Prerequisites

- Familiarity with the Data Load Editor is recommended
- User with **TenantAdmin** access

Pre-configuration Check

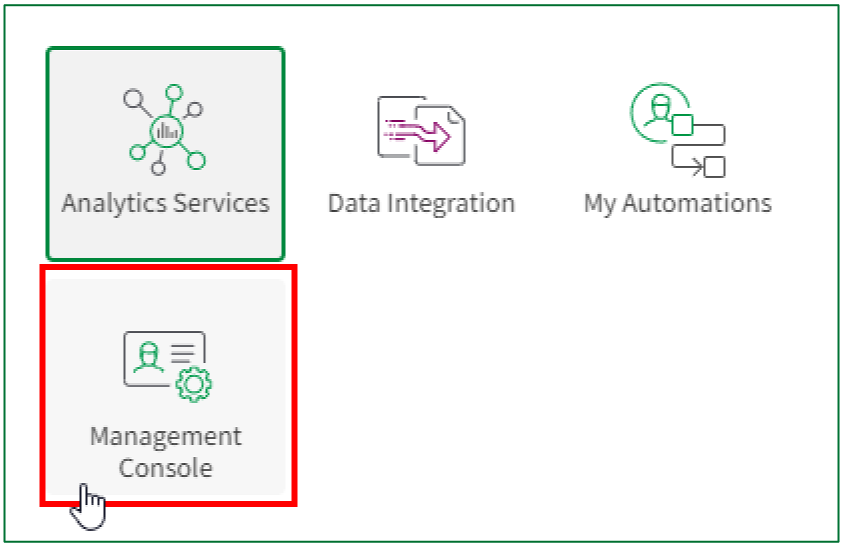
Another Monitoring Application Has Already Been Configured

If another monitoring application has already been configured, this means that the prerequisites have already been met and the data connection required (**monitoring_apps_REST**) has already been created. Proceed directly to Import and Configure the Monitoring Application.

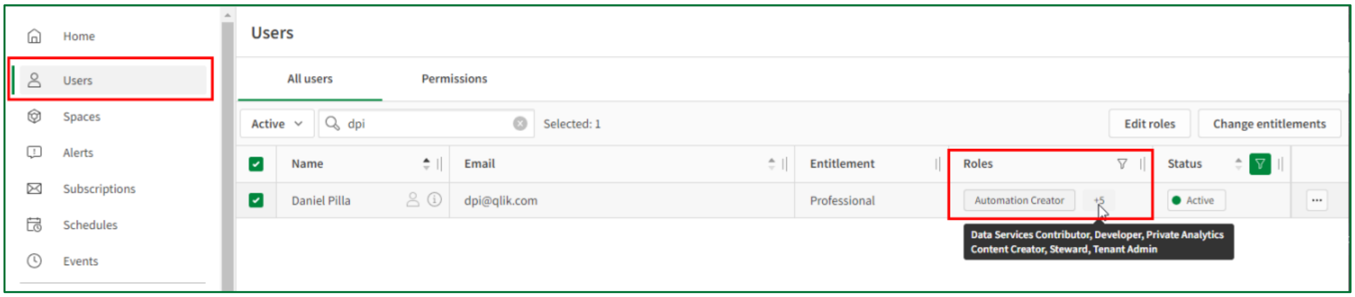
Configure the Tenant for Monitoring Applications

Setup User for Development Access

1. Within the Hub, select the **Grid** icon and then select **Management Console**.



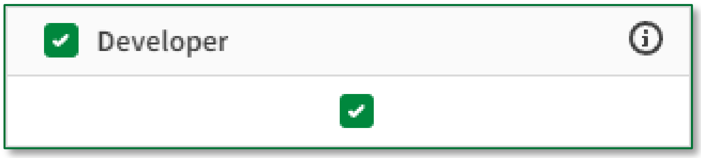
2. Find the user that will be the authenticated user to request the data. This user must have the **TenantAdmin** role.



☑	Name	Email	Entitlement	Roles	Status
☑	Daniel Pilla	dpi@qlik.com	Professional	Automation Creator +5	● Active

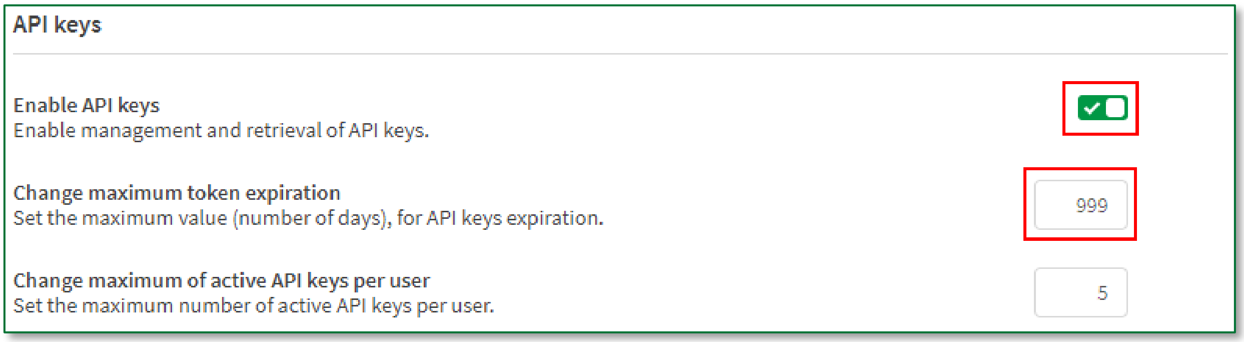
Data Services Contributor, Developer, Private Analytics Content Creator, Steward, Tenant Admin

3. The user must have the Developer role in order to create an API key. If the user does not already have the Developer role, select the user, and click **Edit roles**. Find the **User** role of **Developer** and enable it Select **Save**.



Configure API Settings and Generate API Key

1. Within the Management Console, navigate to **Configuration > Settings > API keys** and toggle on **Enable API keys** and set the **Change maximum token expiration** value to a that is acceptable to your organization. The image below shows 999 which is about 3 years.



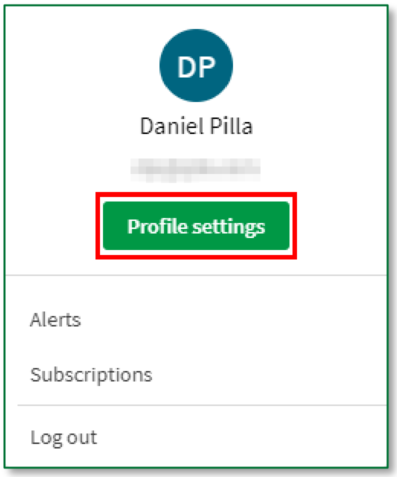
API keys

Enable API keys
Enable management and retrieval of API keys.
✓

Change maximum token expiration
Set the maximum value (number of days), for API keys expiration.
999

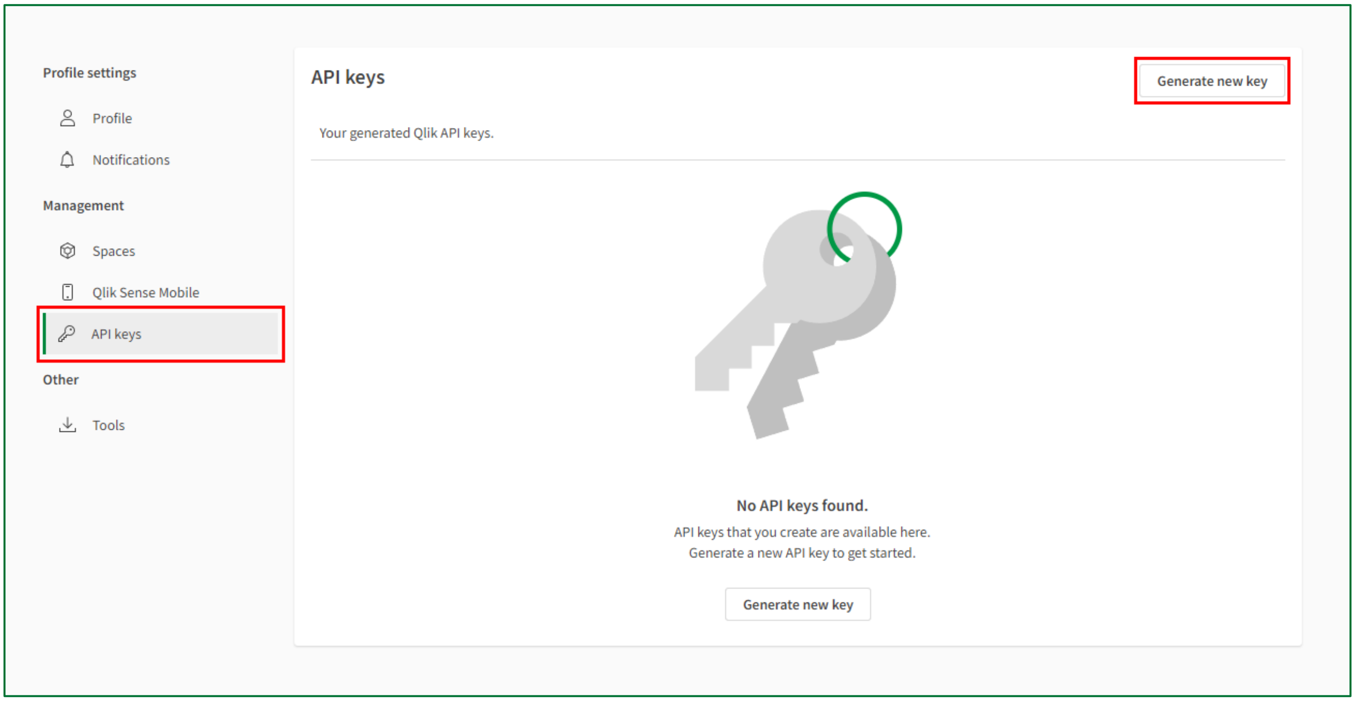
Change maximum of active API keys per user
Set the maximum number of active API keys per user.
5

2. Navigate back to the **Hub** then to **User profile > Profile settings**.



DP
Daniel Pilla
[redacted]

3. Go to the **API keys** section. Select **Generate new key**.



API keys

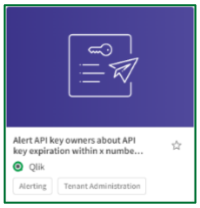
Your generated Qlik API keys.

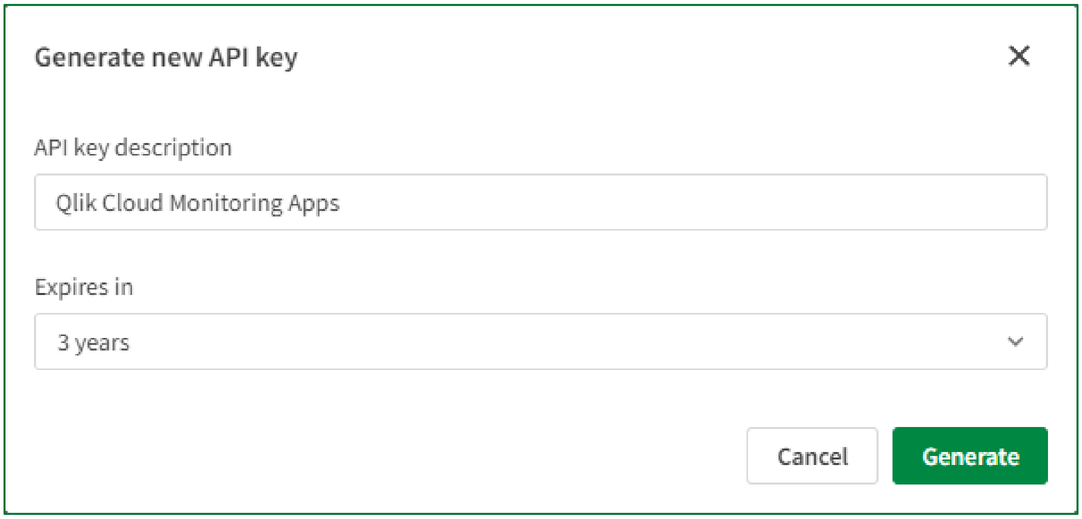
No API keys found.

API keys that you create are available here.
Generate a new API key to get started.

4. Enter in an **API key description** and set the desired expiration. Click **Generate**.

   Note: There is a default **Qlik Application Automation** template in Qlik Cloud titled *Alert API key owners about API key expiration within x number of days* that can automatically notify the API key owner as well as TenantAdmin users in advance of the API key expiration date.


Alert API key owners about API key expiration within x numbe...
Qlik
Alerting
Tenant Administration



Generate new API key

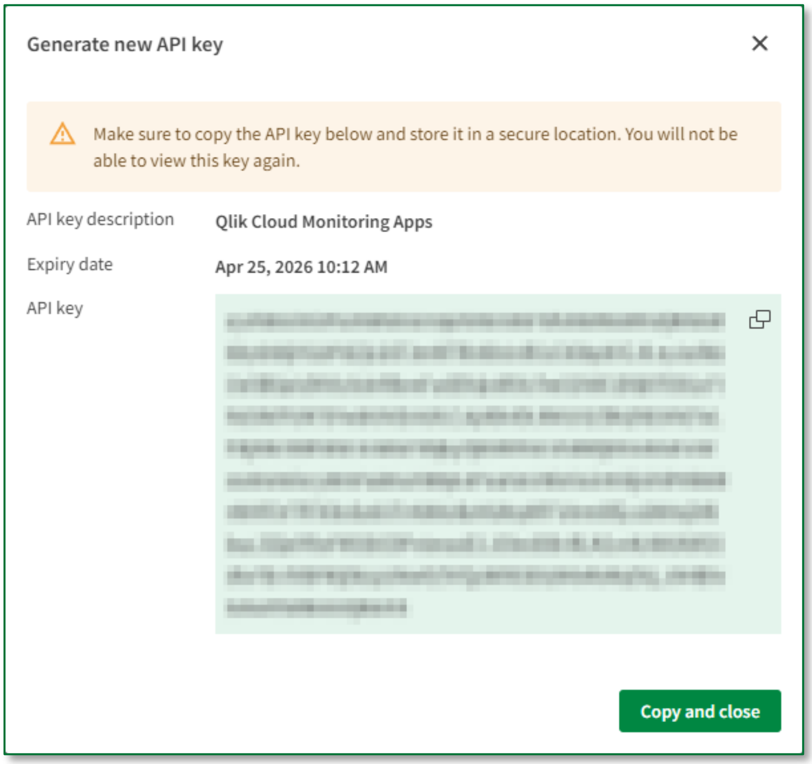
API key description

Qlik Cloud Monitoring Apps

Expires in

3 years

5. Copy the **API key** and store it in a secure location for safe keeping. This key is required to create the REST connection required to reload the app.



⚠ Make sure to copy the API key below and store it in a secure location. You will not be able to view this key again.

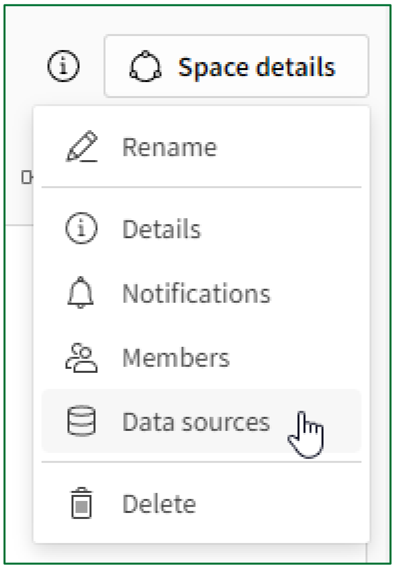
API key description: Qlik Cloud Monitoring Apps

Expiry date: Apr 25, 2026 10:12 AM

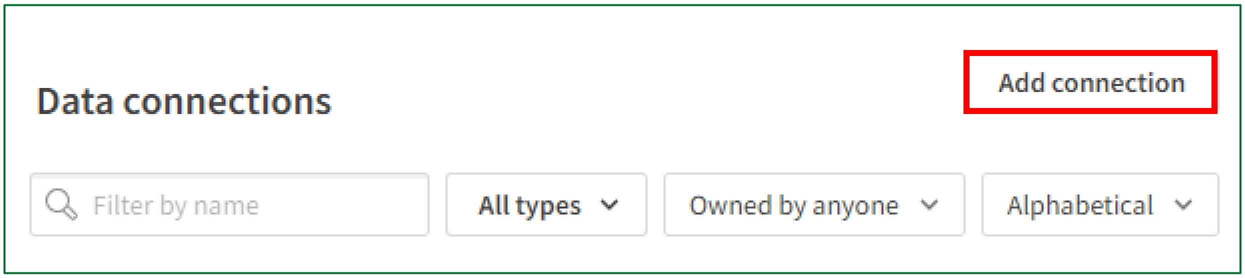
API key: [redacted]

Create Data Connection for the Monitoring Apps

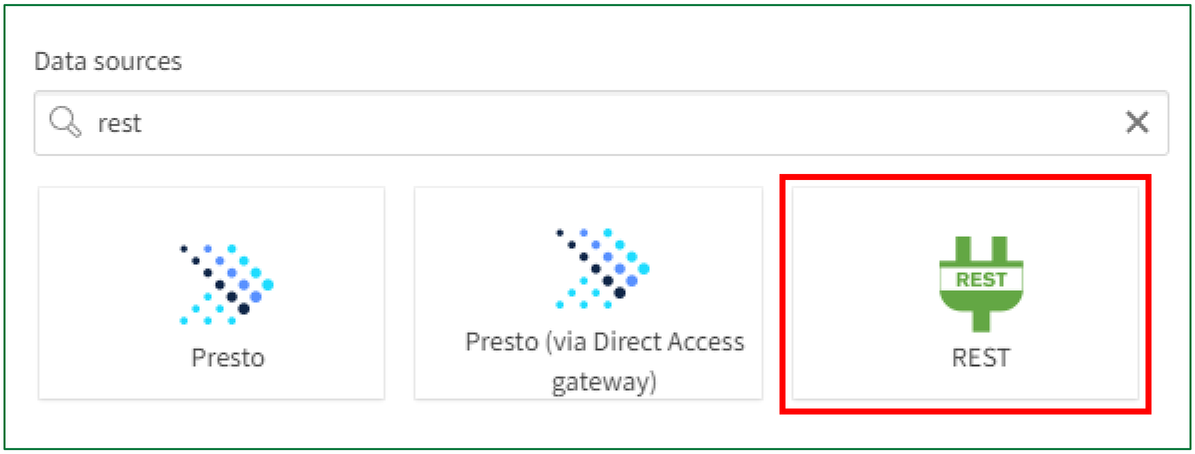
1. Navigate to the space of your choice and select **Space details** and then **Data sources**.



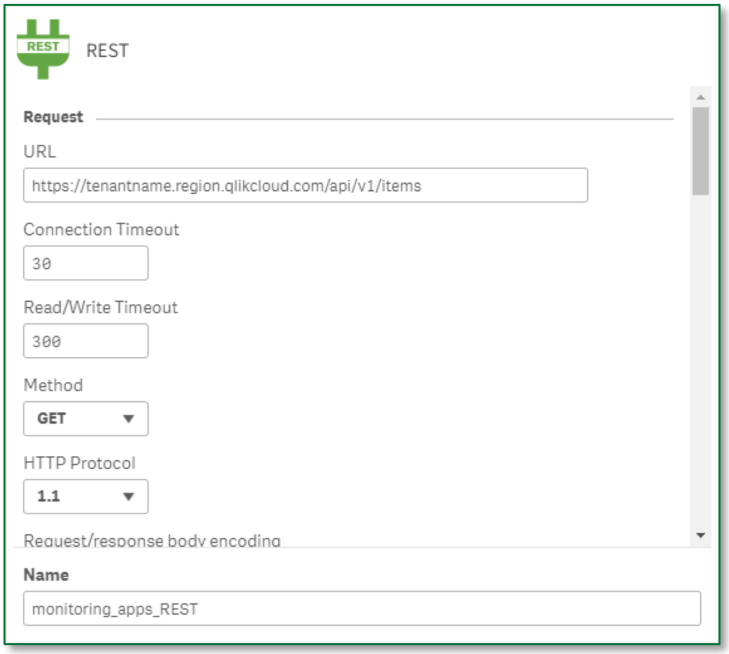
2. Under **Data connections**, select **Add connection**.



3. Select the **REST** connector.



4. Set the **URL** to *https://<tenant>.<region>.qlikcloud.com/api/v1/items*, replacing the *<tenant>.<region>* to match the environment to be monitored.
   Example: *https://company.us.qlikcloud.com/api/v1/items*
   Note: The default hostname is recommended as the alias hostname may change over time.



REST

Request

URL
https://tenantname.region.qlikcloud.com/api/v1/items

Connection Timeout
30

Read/Write Timeout
300

Method
GET

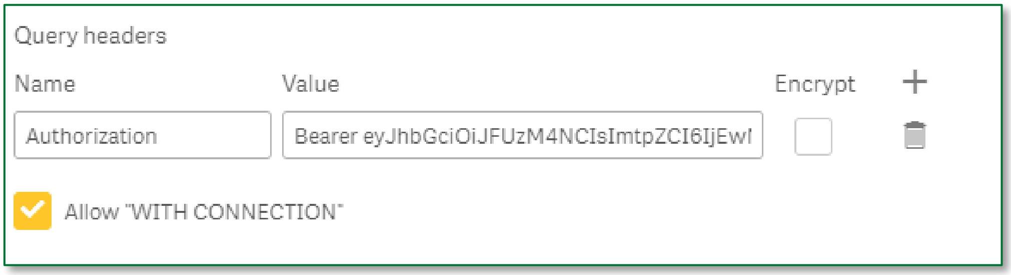
HTTP Protocol
1.1

Request/response body encoding

Name
monitoring_apps_REST

5. Use the default values for the settings and scroll down to the **Additional request parameters** section.

6. Under **Query Headers**, add the name **Authorization** and set the value to **Bearer <paste API key here>.** In addition, ensure that the **Allow "WITH CONNECTION"** option is enabled.



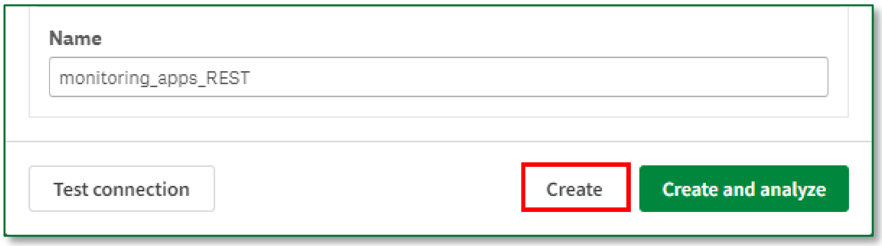
Query headers

Name	Value	Encrypt
Authorization	Bearer eyJhbGciOiJFUzM4NCIsImtpZCI6IjEwI	☐

☑ Allow "WITH CONNECTION"

7. Rename the connection to **monitoring_apps_REST**. This is the default connection name that is set in the load script of the app. Click **Create**.

   Note: The connection will not save if the API key is not valid.



Name

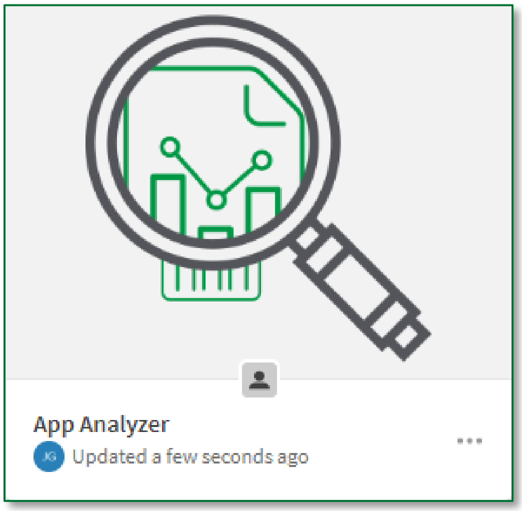
monitoring_apps_REST

Import and Configure the Monitoring Application

Import the Monitoring Application

1. Go to the **Hub > Add new > Upload app**.

2. **Drag and drop** or **browse to the location** to upload the application into the environment. Choose the **Space**, add **tags** if necessary, and click **Upload**.

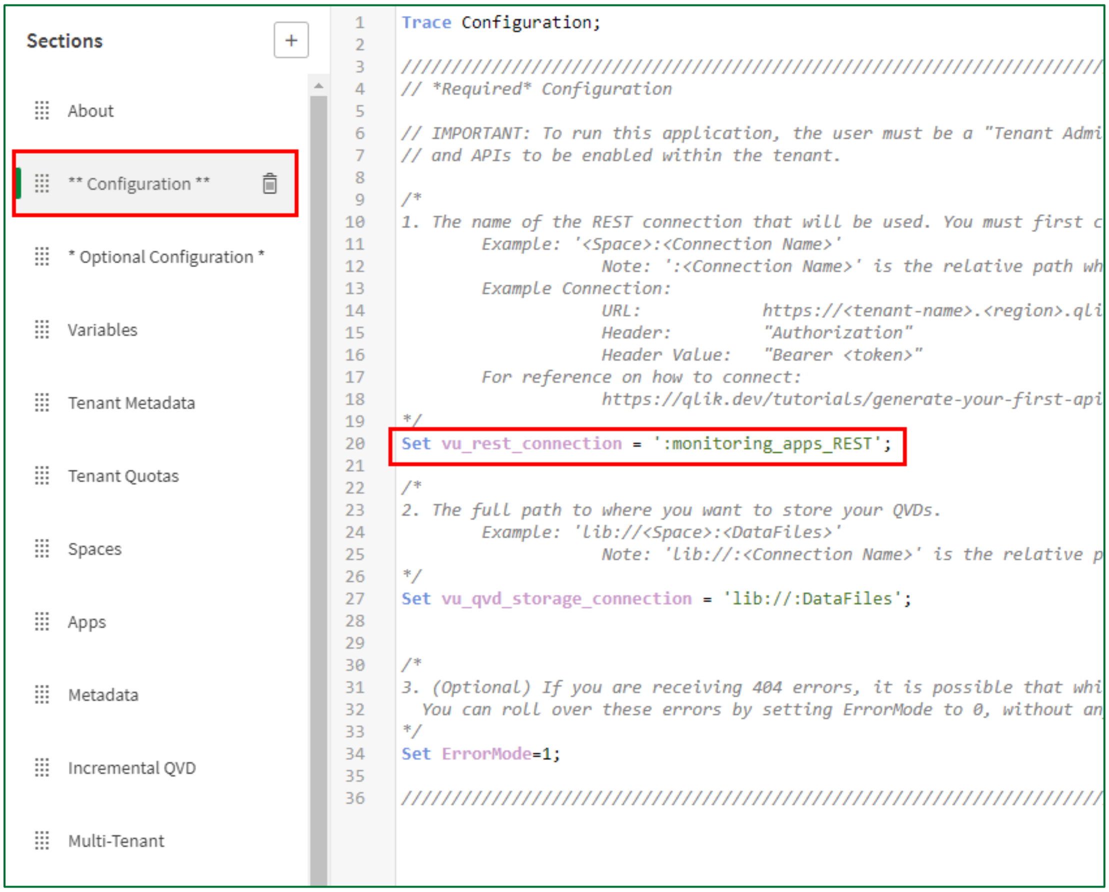


Configure the Monitoring Application

1. Open the application and go to the **Data load editor**.

2. Navigate to the **** Configuration **** tab and verify the `vu_rest_connection` variable value matches the name and space of the REST connection from Create the Data Connection for the Monitoring Apps. If it does not match, update the variable.

   Note: `‘:monitoring_apps_REST’` uses the relative path which will check for a connection in the current space. The full path may also be used, as such: <Space Name>:<Connection Name>.

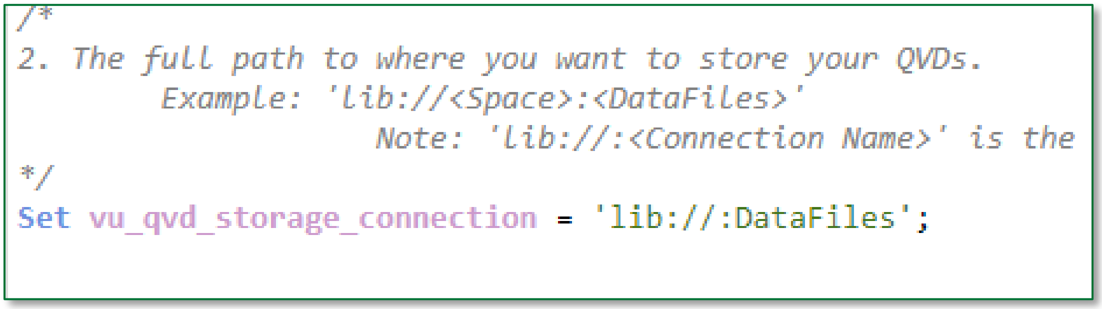


```
Trace Configuration;

//////////////////////////////////////////////////////////////////////
// *Required* Configuration

// IMPORTANT: To run this application, the user must be a "Tenant Admi
// and APIs to be enabled within the tenant.

/*
1. The name of the REST connection that will be used. You must first c
        Example: '<Space>:<Connection Name>'
                    Note: ':<Connection Name>' is the relative path wh
        Example Connection:
                    URL:                https://<tenant-name>.<region>.qli
                    Header:             "Authorization"
                    Header Value:       "Bearer <token>"
        For reference on how to connect:
                    https://qlik.dev/tutorials/generate-your-first-api
*/
Set vu_rest_connection = ':monitoring_apps_REST';

/*
2. The full path to where you want to store your QVDs.
        Example: 'lib://<Space>:<DataFiles>'
                    Note: 'lib://:<Connection Name>' is the relative p
*/
Set vu_qvd_storage_connection = 'lib://:DataFiles';


/*
3. (Optional) If you are receiving 404 errors, it is possible that whi
  You can roll over these errors by setting ErrorMode to 0, without an
*/
Set ErrorMode=1;

//////////////////////////////////////////////////////////////////////
```

The following step does not apply for the ***Access Evaluator*** *application, as that application does not leverage QVDs.*

3. Verify the `vu_qvd_storage_connection` variable value and adjust if necessary. This is the location where incremental QVDs will be stored.

   Note: `‘Lib://:DataFiles’` uses the relative path which will check for a connection in the current space. The full path may also be used, as such: <Space Name>:<Connection Name>.



```
/*
2. The full path to where you want to store your QVDs.
        Example: 'lib://<Space>:<DataFiles>'
                        Note: 'lib://:<Connection Name>' is the
*/
Set vu_qvd_storage_connection = 'lib://:DataFiles';
```

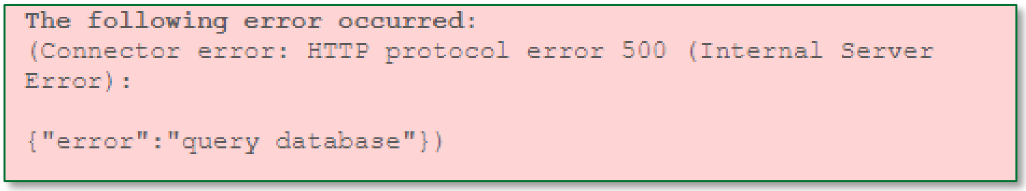
4. The application is now ready to reload. Click **Load data** in the top right-hand corner to reload the app.

Troubleshooting

Reload Analyzer

Large Amount of Data

If while reloading the application a `500 (Internal Server Error)` is presented, this is likely due to the fact that the tenant has a large amount of reload data, and the 90 day range (default) is too large.



```
The following error occurred:
(Connector error: HTTP protocol error 500 (Internal Server
Error):

{"error":"query database"})
```

1. Navigate to the *** Optional Configuration *** tab.

2. Find the `vu_initial_days_back` variable and adjust it to a smaller range. The default is `90`, so it must be an integer less than that number. Repeat this step until the reload no longer shows the 500 error.

    a. If set to `90` set to `60`

    b. If set to `60` set to `30`

    c. If set to `30` set to `7`

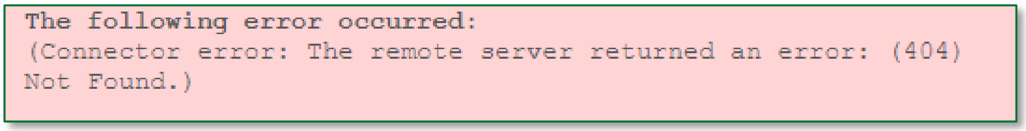
    d. etc

Note: This variable is ignored in the load script following a successful reload. After a successful reload, the app will begin to build incrementally on whatever the foundation (`vu_initial_days_back)` was set to. Meaning, if `vu_initial_days_back` was set to `30`, the app was reloaded successfully, and then `vu_initial_days_back` was set to `60`, and the app was reloaded successfully again, the result will not hold 60 days of data, but rather it will hold 30 days of data with the incremental addition of the time since the last reload. If it is desired to increase this number, the `reload_analyzer_*` QVDs would need to first be deleted. Hence, step 2 above shows that it is best to decrement this variable rather than increment. If chosen to increment, QVDs will need to be deleted prior to the subsequent reload for the variable to function properly.

App Analyzer

Receiving 404 Errors

If while reloading the application a (`Connector error: The remote server returned an error: (404) Not Found.)` error is presented. This is caused by an application being deleted while the application is being reloaded, hence its metadata can no longer be fetched.



```
The following error occurred:
(Connector error: The remote server returned an error: (404)
Not Found.)
```

1. Navigate to the ** Configuration ** tab.

2. Find the `ErrorMode` variable and set it to `0`. This will prevent the script from halting on those errors. Only make this modification once you are confident that the application is properly reloading other than these occasional errors occurring.

About Qlik

Qlik’s vision is a data-literate world, where everyone can use data and analytics to improve decision-making and solve their most challenging problems. Qlik provides an end-to-end, real-time data integration and analytics platform to close the gaps between data, insights and action. By transforming data into active intelligence, businesses can drive better decisions, improve revenue and profitability, and optimize customer relationships. Qlik does business in more than 100 countries and serves over 50,000 customers around the world.

qlik.com