



| INSTALLATION GUIDE

Qlik Cloud Monitoring Apps

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

Table of Contents

Introduction	3
Automate the process	3
Prerequisites	3
Configure the tenant for the Monitoring Apps	4
Grant access to generate API keys	4
Assign the Audit Admin role	9
Configure API key settings	11
Create an API key	12
Create a data connection for the Monitoring Apps	14
Import and configure the Monitoring App	17
Troubleshooting	19
Reload Analyzer	19
App Analyzer	20

Introduction

This guide takes a Qlik Cloud user through installing a Qlik Cloud monitoring application step-by-step.

All Qlik Cloud monitoring applications **other than the OEM Dashboard** 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 Qlik Cloud monitoring applications can be found [here](#).

Automate the process

If you prefer an automated approach to deploying and maintaining these apps, please use the [Qlik Automate installer and API key rotator automations](#) instead of installing these applications manually. These take the complexity out of installing and maintaining these apps, and handle updates and upgrades automatically.

Prerequisites

To begin, you must have:

- A Qlik Cloud tenant.
- A user account assigned the **TenantAdmin** role.

Configure the tenant for the Monitoring Apps

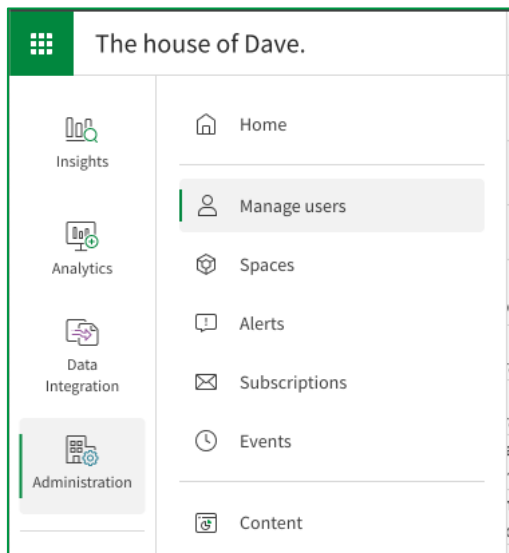
To refresh the apps with data from your tenant, you need an API key for your user account, and the relevant permissions to access that data.

Grant access to generate API keys

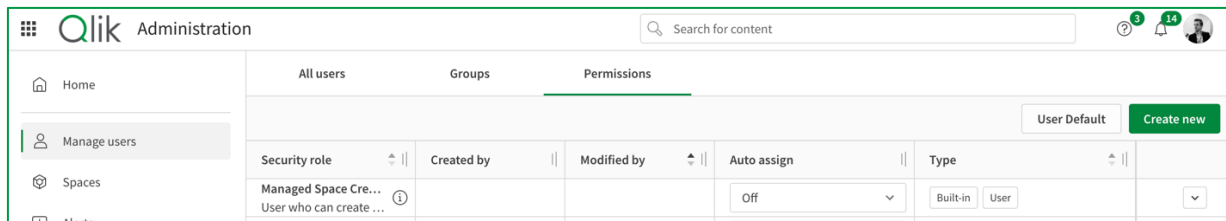
To give users the ability to generate API keys, you must either grant this access to all users or create a custom role which you then assign specifically to your user.

Grant all users access to API keys

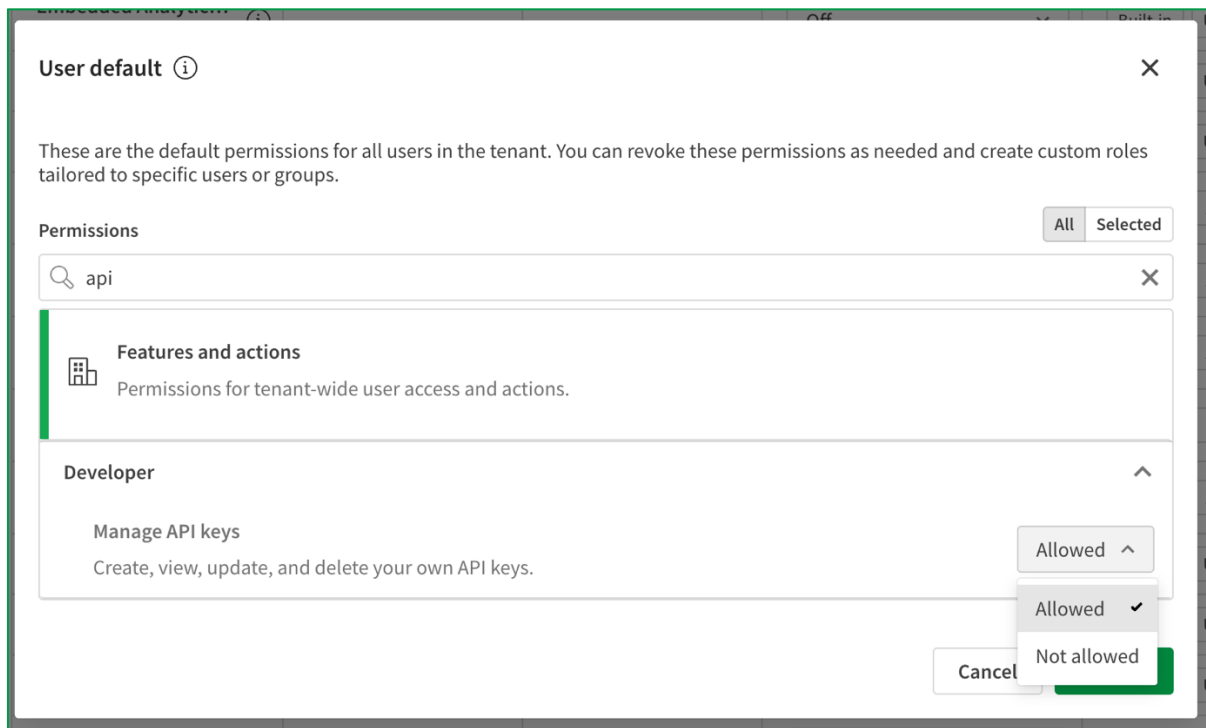
1. While in your tenant, access the **Grid** icon and select **Administration** followed by **Manage users**.



2. Select the **Permissions** tab and click the **User Default** button to open the default permissions available to all users in the tenant.



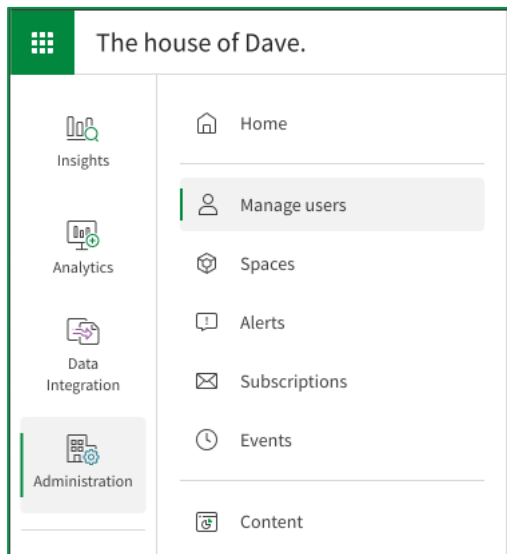
3. In the **User default** dialog, search for **api** and ensure **Manage API keys** is set to **Allowed**. Click **Confirm** to apply the change.



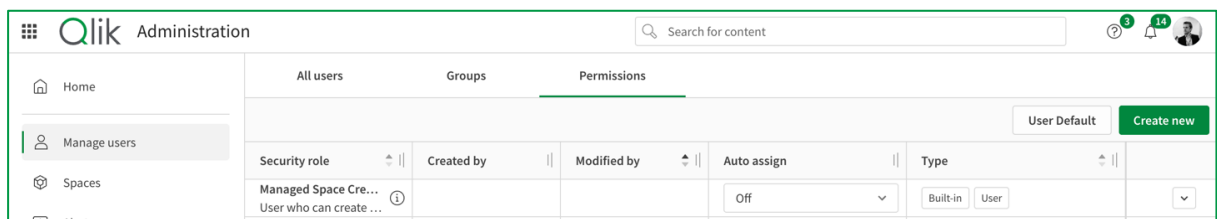
All users in the tenant will now have access to create API keys.

Grant only your user access to API keys

1. While in your tenant, access the **Grid** icon and select **Administration** followed by **Manage users**.



2. Select the **Permissions** tab and click the **Create new** button to open the **Create new role** dialog.



3. In the **Create new role** dialog, set an appropriate name and description for the role, then search for **api** and mark **Manage API keys** to **Allowed**.

Create new role ⓘ

Create a new role with customizable permissions to assign to specific users or groups. Remove these permissions from the **User default** role to restrict access only to users with the new role.

Name

API Key Access

Description

Users who can create API keys.

Permissions All Selected

api

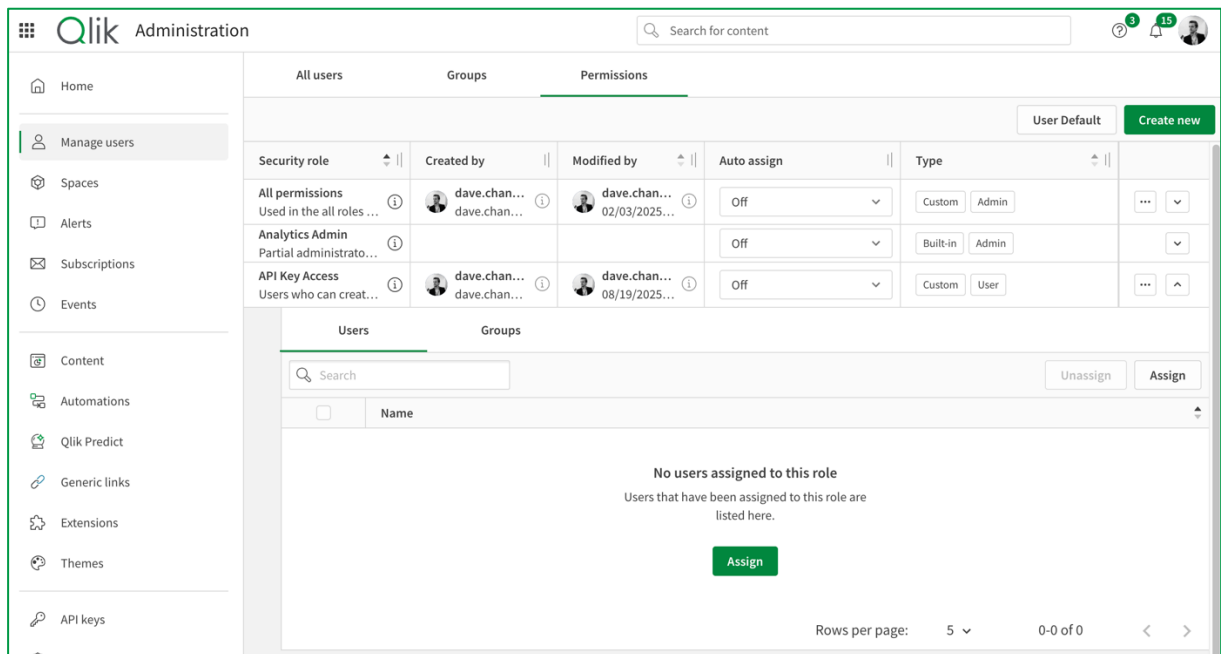
Features and actions
Permissions for tenant-wide user access and actions.

Developer ^

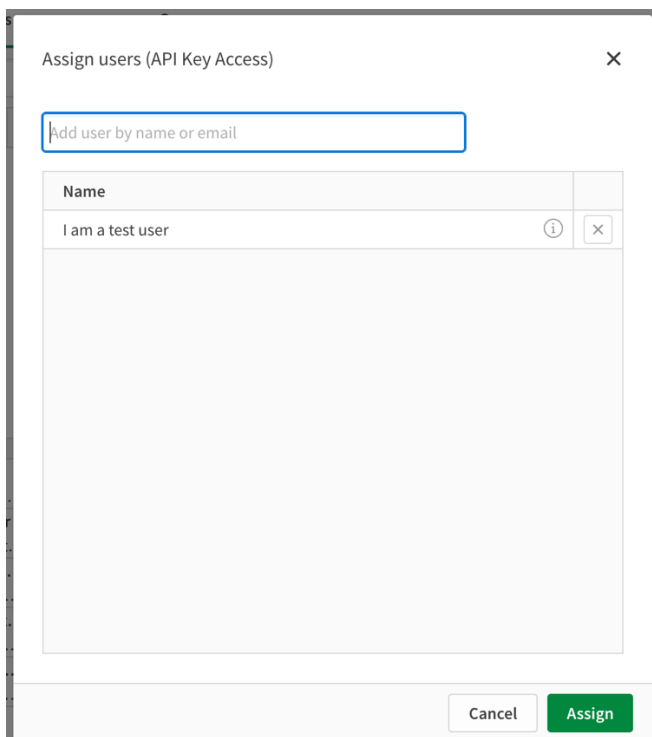
Manage API keys
Create, view, update, and delete your own API keys. Allowed ▾

Cancel Confirm

4. Once back in the **Permissions** tab, click the  icon on the new role to expand the role assignment tool. Select **Assign** to open the assignment modal.



5. In the **Assign users** modal, search for and add your user to the role, then select **Assign** to apply the changes.

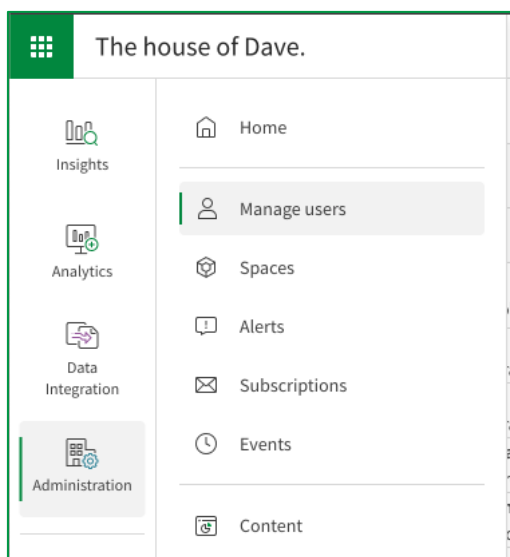


Your user will now have access to create API keys.

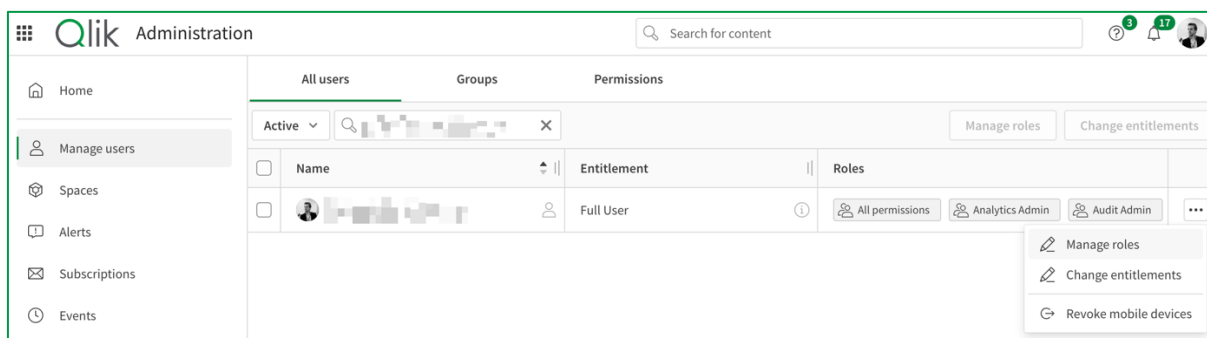
Assign the Audit Admin role

This is only required if you plan to install the **Answers Analyzer** app, skip this step if you are not using this app.

1. While in your tenant, access the **Grid** icon and select **Administration** followed by **Manage users**.



2. Select the **...** icon next to your user and select **Manage roles**.



3. In the **Manage roles** modal, navigate to the **Admin** tab and ensure **Audit Admin** is selected (in addition to any roles you may already have). Click **Save** to apply the roles.














Manage roles

×

Here you can assign or remove roles for a single user. To manage roles for groups or all users, use the Permissions tab.

UserAdmin

Add user by name or email

Name	<input type="checkbox"/> All permissions ⓘ	<input type="checkbox"/> Analytics Admin ⓘ	<input checked="" type="checkbox"/> Audit Admin ⓘ	<input type="checkbox"/> Data Admin ⓘ		
    	<input type="checkbox"/>  	<input type="checkbox"/>  	<input checked="" type="checkbox"/>  	<input type="checkbox"/>  	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Configure API key settings

In the **Administration** area, navigate to **Settings > API keys**.

Ensure the value for **Change maximum token expiration** is set to a suitable value. The shorter the maximum token expiry, the more frequently you'll need to manually update the REST connection used to load data into the apps, since once the API key expires, the apps will cease to reload successfully.

API keys

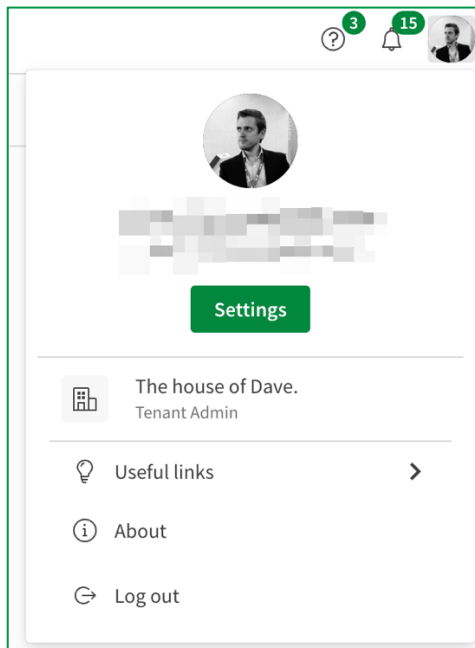
Change maximum token expiration Set the maximum value (number of days), for API keys expiration.	365
Change maximum of active API keys per user Set the maximum number of active API keys per user.	25

To reduce maintenance, consider using the Qlik Cloud Monitoring App installers to automate API key lifecycles.

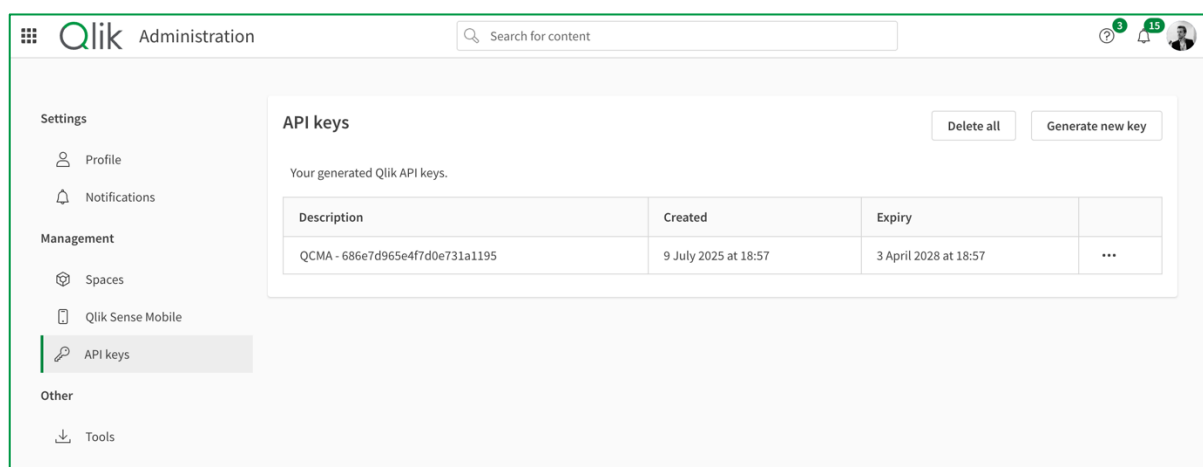
Create an API key

The REST connector requires an API key for the tenant, allowing it to act as your user when accessing APIs. As they have a finite expiration date, they do need to be replaced once they expire.

1. Click on your profile image in the top right of the screen, then select **Settings**.

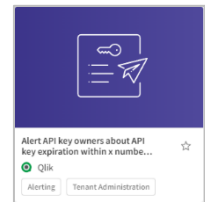


2. Navigate to the **API keys** section and click **Generate new key**. Note this screen also shows you existing API keys for your account.



3. 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.

A screenshot of the "Generate new API key" dialog box. It has a title bar with a close button (X). The form contains two input fields: "API key description" with the value "Qik Cloud Monitoring Apps" and "Expires in" with a dropdown menu set to "6 hours". At the bottom right, there are two buttons: "Cancel" and "Generate".

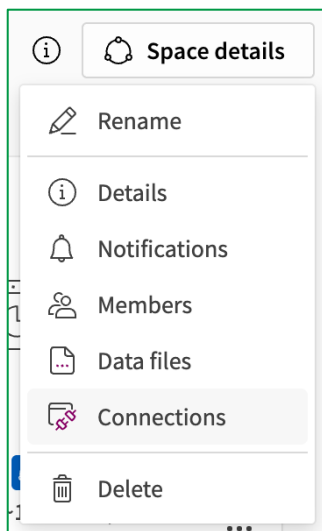
4. Copy the **API key** value and store it in a secure location for safe keeping. This key is required to create the REST connection required to reload the app.

A screenshot of the "Generate new API key" dialog box, showing the generated API key. It includes a warning message: "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." The form displays the "API key description" as "Qlik Cloud Monitoring Apps", the "Expiry date" as "Apr 25, 2026 10:12 AM", and the "API key" as a long alphanumeric string. A copy icon is next to the API key. At the bottom right, there is a "Copy and close" button.

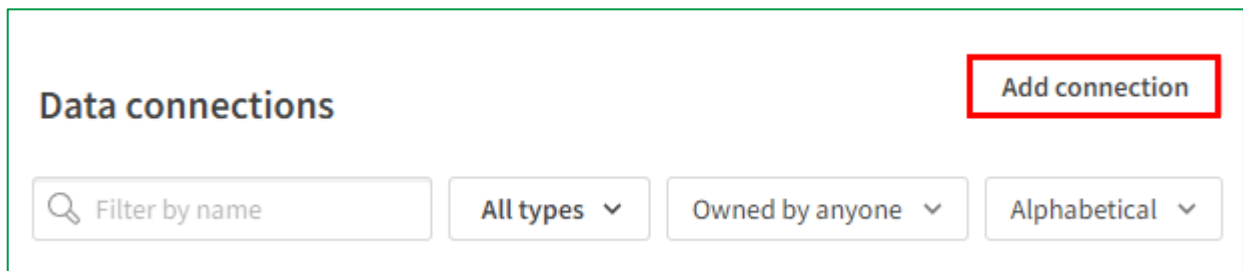
Create a data connection for the Monitoring Apps

The apps will use a shared data connection for loading data from your tenant. If you haven't already, ensure you have a space created for your apps, and put the data connection into that space.

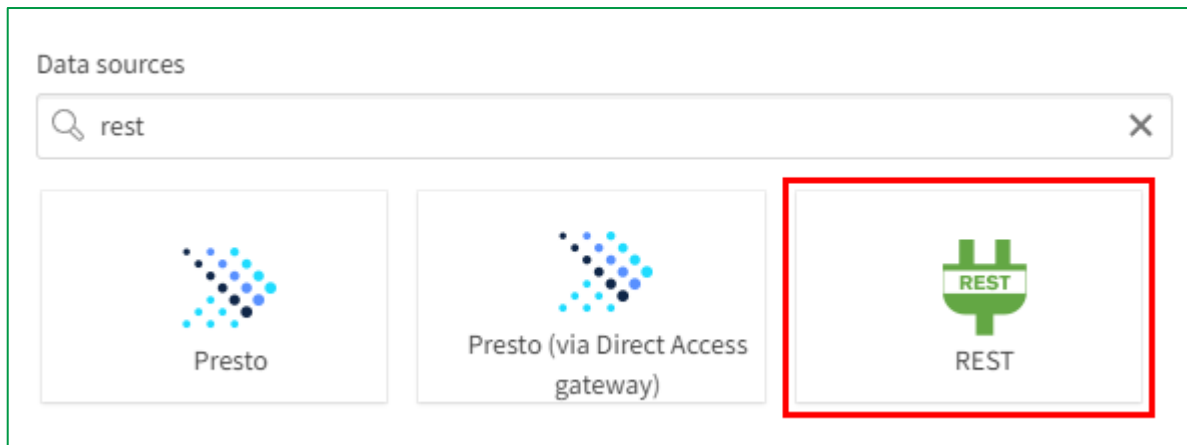
1. Navigate to the space of your choice and select **Space details** and then **Connections**.



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



3. Select the **REST** connector.



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 can be changed.

4. Use the default values for the settings and scroll down to the **Additional request parameters** section.
5. 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	<input type="checkbox"/>	

☒ Allow "WITH CONNECTION"

6. 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, as it'll send a test request to the specified endpoint

Name

monitoring_apps_REST

Test connection

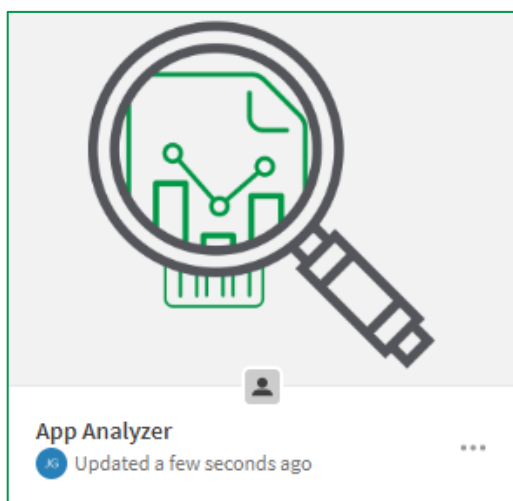
Create

Create and analyze

Import and configure the Monitoring App

With an authenticated REST connection established, you can import the apps and configure them for data loading.

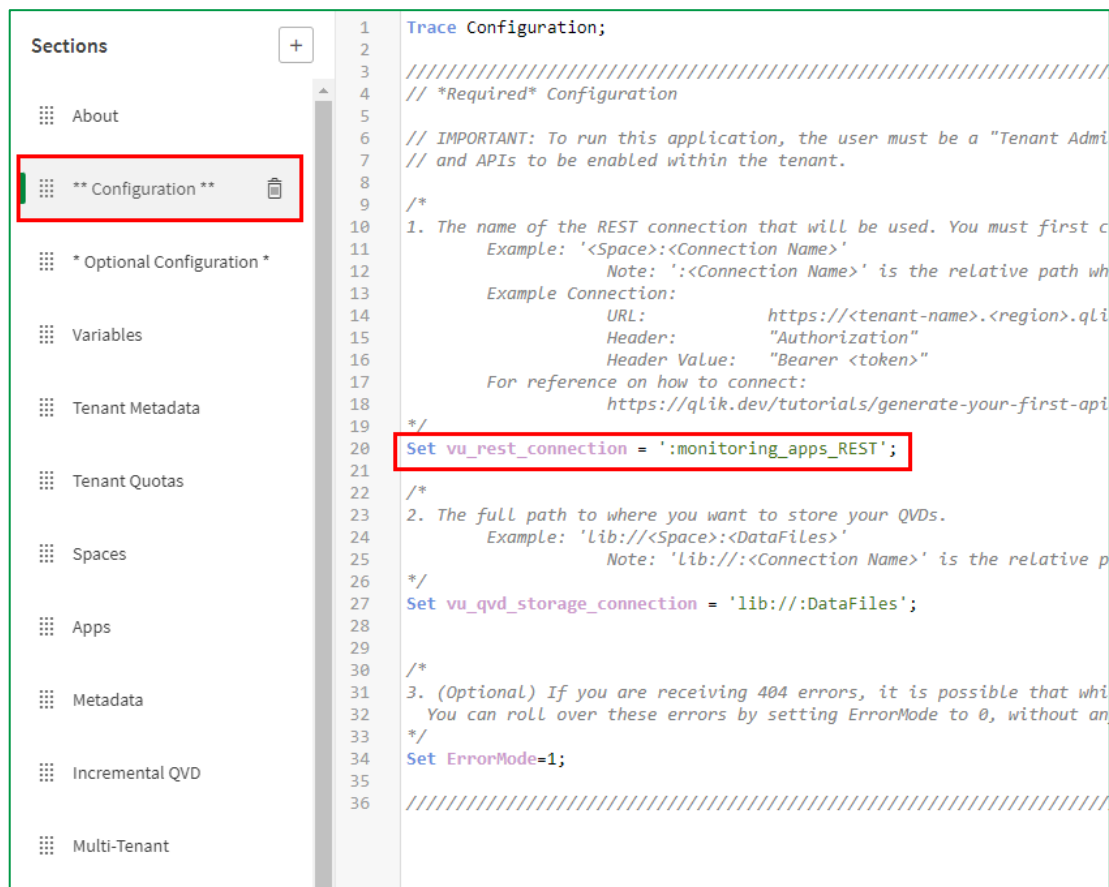
1. Navigate to the **Analytics Activity Center > Create > Upload**.
2. **Drag and drop** or **browse to the location** to upload the application into the environment. Choose the **Space**, add **tags** if necessary, and click **Upload**.



3. Open the application and navigate to the **Data load editor**.
4. Navigate to the **** Configuration **** tab and verify the `vu_rest_connection` variable value matches the name and space of the REST connection you created earlier. If it does not match, update the variable to match the name of the connection.

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>.



5. If configuring the Access Evaluator application, skip this step.

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 relative path within the tenant.
*/
Set vu_qvd_storage_connection = 'lib://:DataFiles';
```

6. 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 because 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.



Qlik transforms complex data landscapes into actionable insights, driving strategic business outcomes. Serving over 40,000 global customers, our portfolio leverages advanced, enterprise-grade AI/ML and pervasive data quality. We excel in data integration and governance, offering comprehensive solutions that work with diverse data sources. Intuitive and real-time analytics from Qlik uncover hidden patterns, empowering teams to address complex challenges and seize new opportunities. Our AI/ML tools, both practical and scalable, lead to better decisions, faster. As strategic partners, our platform-agnostic technology and expertise make our customers more competitive.

[Qlik.com](https://www.qlik.com)

© 2025 QlikTech International AB. All rights reserved. All company and/or product names may be trade names, trademarks and/or registered trademarks of the respective owners with which they are associated. For the full list of Qlik trademarks please visit: <https://www.qlik.com/us/legal/trademarks>.