

API Reference

The Quatrix API reference will walk you through a set of file sharing features that you can use in your own app. It explains the basics for creating API calls, covers authentication methods, provides design principles for API calls and displays error codes.

- [Overview](#)
- [Before you start](#)
- [Authentication](#)
- [Design](#)
- [Errors](#)
- [Support](#)

Overview

[Quatrix](#) is secure cloud file sharing platform with centralised administration, easy-to-use interface, and SFTP access. You can use it for sharing and acquiring files securely and reliably online. It combines fast file transfer, convenience and flexibility of use and premium-grade security features. Check out [Quatrix homepage](#) for details.

Quatrix API documentation is intended for developers who want to write applications that interact with Quatrix.

Our goal is to simplify adding file sharing features to your application so that you can focus on integrating the core features to your product.

Before you start

You need to have a Quatrix account to access Quatrix APIs. If you haven't registered yet, set up a free trial account [here](#).

If you are not familiar with Quatrix features, read the introductory page about Quatrix on our [website](#), learn about features [here](#) and explore the user interface before getting down to code.

Authentication

Quatrix supports two methods of API authentication: HTTP Basic Authentication (by default in Swagger) and API keys.

Basic authentication and session cookie

Quatrix maintains state throughout the session, for that reason you need to log in to your Quatrix account in order to get session ID.

You can use [HTTP basic authentication](#) to authorize the requests which means you can simply send the user name and password associated with the account. If you create a basic authentication header manually, stick to the following rules:

- The user email and password should be combined with a single colon (:) e.g.

```
test@example.com:qwerty
```

- The resulting string is encoded using a variant of Base64.
- The authorization method (e.g. Basic) should be prepended to the encoded string and separated with space. This is further used as a value for [Authorization field](#). e.g.

```
Authorization: Basic dGVzdEBleGFtcGxlLmNvbTpxd2VydHk=
```

Follow the steps below to authorize using this method:

1. Using your user name and password execute the API call GET/session/login with the following header in order to get session ID.

```
Authorization: Basic {b64encode('login:password')}
```

2. All your further calls should be executed with the header:

```
X-Auth-Token: {session_id}
```

or cookie:

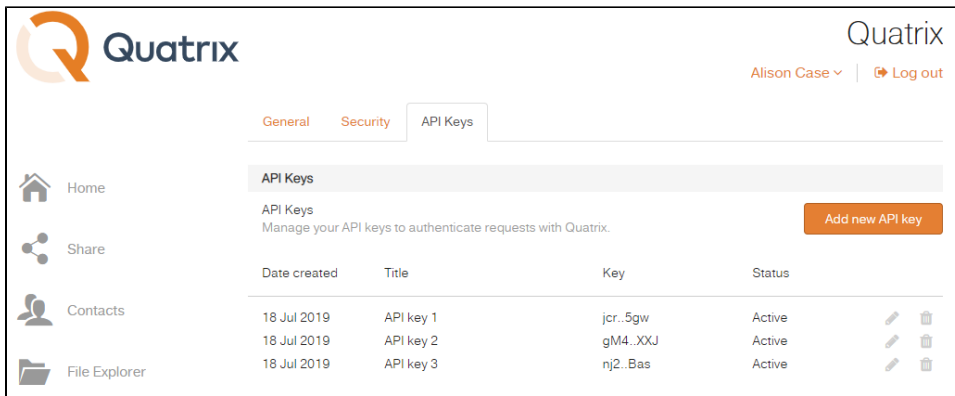
```
token={session_id}
```

API keys for authentication

An API key is a token that the client provides when making API calls. Follow the steps below to get an API key from Quatrix:

1. Log in to Quatrix and open your profile by clicking on the link with your name at the top right.

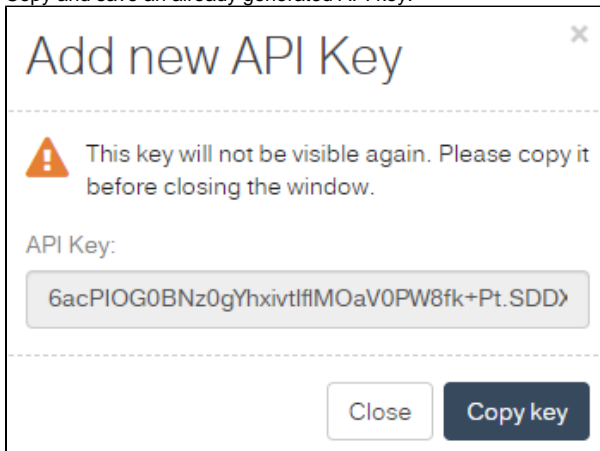
2. Go to API keys sub-tab and click on Add new API key button.



3. Type in the title and click on the [Generate new API key](#) button.

Give your key an informative title so that you can easily tell which key is used for which integration.

4. Copy and save an already generated API key.



⚠ This API key will never be displayed again (only the key's title will be shown), make sure you copied and stored it securely.

Your newly generated API key is added to the list where you can edit only its title or delete it.

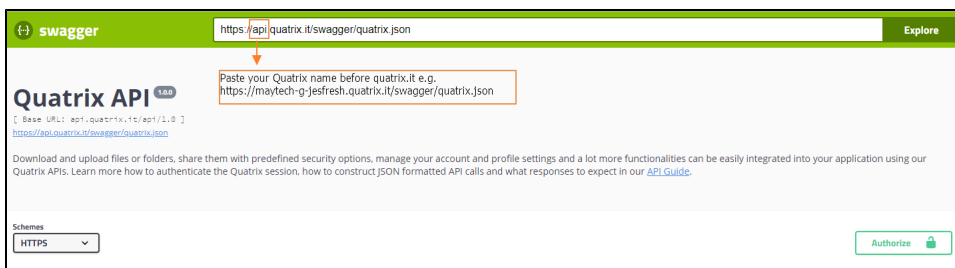
The given API key can be sent as a [request header](#):

```
Authorization: Bearer {api_key_value}
```

Swagger usage example

Follow these steps to authorize with Basic authentication;

1. Open [Swagger](#), paste your URL (<https://yourcompany.quatrix.it/swagger/quatrix.json/>) into the blank field and click on the **Explore** button.



2. Click on the **Authorize** button that will open **Available authorizations** window, fill in your user name and password in **Basic authorization** section and press the **Authorize** button.
3. Go to **Auth** resource and make [GET /session/login](#) API call to retrieve a session ID. [See the screenshot](#).
4. Go back to the **Authorize** button and paste the session ID into the **Value** field in the **API key** section.
5. Click on the **Authorize** button.

Once authorized, you can make any call by clicking first on the **Try it out** button and then on the **Execute** button that will provide results in the response.

Subsequent API calls are authorized using this session ID. The session is valid/active for 15 minutes since the last API action. If there was no activity during this period, the session will expire, and the user must log in again.

Design

We use RESTful design patterns for URLs and our API methods support GET and POST requests. Each method operates on a certain resource (view the list of resources below) and produces a JSON response.

Resource	Description
Auth	Authenticate Quatrix account
Account	Get general information about the current account account
Language	Get details about account language
Contact	Create and manage contacts
PGP	Generate and manage PGP keys
Group	Get information about users and contacts
File	Retrieve and modify files/folders (hereinafter files)
Upload	File uploading
Password reset	Reset password for existing users
Preview	Preview photos, videos or files
Profile	Update profile information of the user
MFA	Enable and disable MFA
Project folder	Create and manage project folders
Site settings	Manage service settings
User	Create and manage users
Widget	Set the widget
Share	Create shares, quick links, file sharing requests
Activity log	Track activity history
Service	Get information about the service
Automation	Create and administer automatic rules
Inbox	Activity tracking of a logged-in contact
SSH key	Create and manage SSH keys
Billing	Account upgrade

You can try out our APIs using [Swagger](#) where you can dig deeper into each resource and explore related endpoints with possible HTTP methods.

Base URL

The base URL for API calls should be formatted in the following way:

```
https:// {YourQuatrixSiteName}.quatrix.it/api/1.0/
```

The current API version is 1.0.

Errors

Conventional [HTTP response codes](#) are used to indicate the success or failure of an API request. When an error occurs, it contains an error code and the description so that you can easily identify the problem and solve it.

In general, we use codes in the range of 2 to indicate the success of the request. E.g. 200 – login success response. Codes in the range of 4 display the API request failure due to e.g. 400 – parameter error, or 401 - CAPTCHA is wrong.

We use the following model to display an error where the message displays general information about an error, and details specifies additional information about it.

```
{
  "msg": "string",
  "code": 0,
  "details": {}
}
```

The table below lists codes with their general descriptions.

Code	Error Detail
-1	Unknown error
10	Bad parameter value
11	Missing parameter
12	Bad host name
20	Not found
21	Not supported
30	Object exists
31	User exists
32	Contact exists
40	Operation not supported
50	Restriction error
51	PGP disabled error
60	Expired error
61	Not allowed without set up 2FA
70	Not allowed
71	Not allowed without set up 2FA
72	Not allowed without password
73	Not allowed without PIN
74	Not allowed without CAPTCHA
75	Not allowed without Basic Auth
76	Not allowed IP restriction
77	Current IP not in IPR masks
78	Widget CAPTCHA required
79	Widget file limit
80	Deactivated
81	Inactive
82	Deactivated role

83	Failed 2FA
84	Failed PIN
90	No data
91	Widget doesn't exist

Support

If you have any problems with integration, code or something urgent, feel free to contact us via phone or [email](#).

Int & UK: +44 (0) 1892 861 222 | USA: 1 800 592 1906