Introduction to the Code42 API

Who is this article for?
Find your product plan in the Code42 console on the Account menu (https://support.code42.com/Administrator/Cloud/Code42_console_reference/01_Code42_console_overview#Account_menu).
√ Incydr Professional and Enterprise
Ø Incydr Basic and Advanced
√ CrashPlan Cloud
√ CrashPlan for Small Business (https://support.code42.com/Small_Business) (https://support.code42.com/Small_Business)
Ø Other product plans

This article applies to Code42 cloud environments.

Other available versions:
On-premises (https://support.code42.com/Administrator/6/Monitoring_and_managing/Code42_API_resources/Introduction_to_the_Code42_API) (https://support.code42.com/Administrator/Identify_your_Code42_product_version_role)

Overview

This article provides an introduction to the Code42 API. You can use the APIs to perform a variety of tasks, from performing automated actions to integrating with existing systems.

Considerations

• For assistance with using the Code42 API, contact your Customer Success Manager (CSM) to engage the Code42 Professional Services team.

• Code42 API documentation is publicly available on the Code42 Developer Portal (https://developer.code42.com/). You can see it in a browser without signing in. But the API resources themselves only work for you under these conditions:
  ◦ You have a product plan (https://support.code42.com/Terms_and_conditions/Code42_customer_support_resources/Code42_product_plans) that includes access to the Code42 API.
  ◦ Your credentials rely on local authentication. SSO (https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Introduction_to_single_sign-on) or authentication through any third-party provider will not work.
  ◦ Your role provides permission (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Roles_resources/Manage_user_roles) to access the data necessary to a given API resource. For example, if you do not have permission to
change device settings in the Code42 console, then you don't have permission to change device settings with the API.

If your API calls fail because you do not have permission to use them, you will see reply messages like these:

- HTTP 401 Unauthorized
- HTTP 401 Could not authenticate user
- Your Code42 product plan does not permit use of the Code42 API.

Examples of tasks you can perform using the Code42 API

See the following articles for just some of the tasks you can perform using the Code42 API:

- Sample uses of the Code42 API (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources/Sample_uses_of_the_Code42_API)
- Code42 API resources (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources)
- Forensic Search API (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources/Forensic_Search_API)
- Detection list management APIs (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources/Detection_list_management_APIs)
- Manage alerts with the Code42 API (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources/Manage_security_alerts_with_the_Code42_API)
- Search Audit Log events with the Code42 API (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources/Search_Audit_Log_events_with_the_Code42_API)

Tools for interacting with the Code42 API

The Code42 API is accessible through many tools, such as web browsers, scripting tools, and programming languages. Although the examples in our articles show use of the command line tool curl to interact with the Code42 API, you can use many different tools.

For more information see Tools for interacting with the Code42 API (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources/Tools_for_interacting_with_the_Code42_API).

API documentation

API documentation is available on the Code42 Developer Portal (https://developer.code42.com/).
Syntax and usage

The Code42 API uses a standardized API syntax and command structure. For more information, see Code42 API syntax and usage (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources/Code42_API_syntax_and_usage).

HTTP methods

- GET: Retrieve a resource.
- POST: Create a new resource.
- PUT: Update an existing resource.
- DELETE: Destroy an existing resource.

Example commands

Use the Code42 API by sending requests to the Code42 cloud. In your requests, specify the path, resource, and parameters that define your request.

```
curl -X GET -u 'username' 'https://console.us.code42.com/api/User?q=eg@ex.com&active=true'
curl -X POST -u 'username' 'https://console.us.code42.com/c42api/v3/ping/error'
```

Authenticating with curl

The curl commands listed here expose your credentials on your device and your network. For more secure alternatives, see Authentication below.

Authentication

Most requests to the Code42 API must be authenticated. While you can use basic authentication (username and password) we strongly recommend using a token authentication method for increased security.

For more information about authentication with the Code42 API, see Code42 API authentication methods (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources/Code42_API_authentication_methods).
Basic authentication

To use basic authentication, include your Code42 username in the API request and supply your password when prompted. For example:

```
curl -u "username" https://console.us.code42.com/api/Computer
```

Token authentication

To use token authentication:

1. Include your Code42 username in a **GET** request to **auth/jwt** and supply your password when prompted. For example:

```
curl -u "username" https://console.us.code42.com/c42api/v3/auth/jwt?useBody=true
```

**Authentication cookies are not supported**

You must include the **?useBody=true** query parameter in the **auth/jwt** request to return the token in the response body. Omitting the **?useBody=true** query parameter or using **?useBody=false** results in the token being returned in an authentication cookie, which is not supported by the Code42 API. Include the returned token in subsequent API requests using the authorization header with the **Bearer** scheme, for example, `-H 'authorization: Bearer <token>'`

2. From the reply, copy the value of the **v3_user_token**. In the example below, it is `eyJjdHki...txd546Eg`

```
{"data":{"v3_user_token":"eyJjdHki...txd546Eg"},"error":null,"warnings":null}
```

3. Place the token in the header of your API requests. For example:

```
-H "authorization: Bearer eyJjdHki...txd546Eg"
```

Related topics

- Code42 API resources ([https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources](https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources))
- Introduction to the Code42 API ([https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/](https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/))
- Code42 API syntax and usage (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources/Code42_API_syntax_and_usage)
- Code42 API authentication methods (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources/Code42_API_authentication_methods)
- Tools for interacting with the Code42 API (https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/Code42_API_resources/Tools_for_interacting_with_the_Code42_API)