Using CyVerse API Clients

This page is obsolete. Stay tuned for updates.

You can register and use new API clients as well as search for CyVerse users by using the REST API. For an introduction to the CyVerse API clients, see Science APIs.

Registering API clients

You can request clients for available APIs so that you can implement CyVerse APIs programmatically. Once approved to use the client (usually within two business days), you receive an email with an auto-generated key and secret code, which can be regenerated if necessary.

You also can search for CyVerse users with Full REST API.

Registering an API client for your account

1. Go to CyVerse User Management, click **CyVerse Login**, and enter your CyVerse username and password.
   If you have not already done so, you must register for an account. To retrieve or change your password, username, or email address, see Resetting Your Password.
2. Click the **API Clients** tab.
   - **To view documentation about the API,** click 📄 (**User Manual**).
   - **To email the API maintainer,** click ✉ **(Email the maintainer**).
3. To register an API client:
   a. In the **Available APIs** section, click **Register a Client** for the API you want to register.
   b. In the New API Client form, complete all fields:
      - Enter the name of the API client as it will be displayed in your My Clients list.
      - Enter a description of the client.
      - Enter or paste the URL of the website that explains your API client.
      - Enter or paste the IP address for the API client. If more than one IP address exists for the client, separate each IP address with a comma.
      - Enter a description of how you will use the data generated by use of the API client.
   c. Client **Save**.
      Within approximately 2 business days, you will receive an email from CyVerse Support about the registration of the API client. It approved, the client, key, and secret code will be available in your My API Clients list.

Regenerating an API client key

If you think your API client has been compromised, you can regenerate your key and secret code.

1. On the API Clients tab, click **Regenerate Keys**.
   The new key and secret codes are displayed in your My API Clients list.

Helpful Links

On This Page:

Helpful Links:

- Intro to Science APIs
- CyVerse User Management

About using the REST APIs

The REST API provides read-only access for finding users in the user management portal. API Clients must be registered with CyVerse before being granted an API Key and API Secret. The Key and Secret are used to authenticate the client making a request from the User API.

The following HTTP methods are accepted: GET, POST and PUT.

Authentication

All API client authors must have a current CyVerse account and all API clients must be registered on the user's API Clients tab in the user management portal.

After submitting the form, CyVerse staff will review the submitted application and either approve or deny it. This usually occurs within one business day. After approval, API authors will be issued an automatically generated API key and secret. API clients will pass the key and secret using the HTTP Basic Auth scheme. A Basic Auth header must be passed on each and every request.

> ⚠ After this release, CyVerse will deprecate Basic Auth and use 2-legged OAuth2 instead. Be prepared to rewrite your client to use OAuth2. Sorry.

Basic information

- You may search by full or partial usernames, by any combination of first and last names, or by email address.
- Search terms MUST be at least 3 characters long.
- Search terms are not case sensitive.
- Base URL: https://user.iplantcollaborative.org/api/v1/ .

Successful requests that return one or more users have an HTTP status of **200**. Response headers typically appear as such:

Typical Response Header

```
HTTP/1.1 200 OK
Cache-Control: no-cache, max-age=0
Connection: close
Content-Encoding: gzip
Content-Length: 175
Content-Type: application/json
Date: Tue, 01 May 2012 18:57:46 GMT
Expires: Tue, 01 May 2012 18:57:46 GMT
Server: Apache/2.2.3 (CentOS)
Vary: Accept-Encoding,User-Agent
X-Powered-By: PHP/5.3.15
```

Submitting requests

Requests **must** be made over SSL (i.e., to an https://URL) as in the following example:

```
https://user.cyverse.org
```

Request URLs are RESTful: a resource label is requested along with an identifier, i.e. a search term.

Viewing results

- Results are returned in JSON format.
- Responses are GZIP encoded (compressed).
- Results are not cached.
  - **HTTP Expires** headers are set to a past date.
  - **HTTP Cache-control** headers are sent to prevent caching.

Searching for users

Searching by CyVerse username

Issue an HTTP GET request to https://user.iplantcollaborative.org/api/v1/users/username/PARTIAL_OR_FULL_USERNAME.

Request

```
GET [https://user.iplantcollaborative.org/api/v1/users/username/mgatt]
```

Response

```
{
    "users": [
 {
  "id":"1",
  "username":"mgatto10",
  "firstname":"\u0627\u062d\u0645\u062f",
  "lastname":"\u0635\u0641\u064a\u0627\u0646",
  "email":"mgatto10@email.arizona.edu",
  "institution":null,
  "position":"Kaatib"
 },
 {
  "id":"2",
  "username":"mgatto30",
  "firstname":"Michael",
  "lastname":"Gatto",
  "email":"mgatto10@iplantcollaborative.org",
  "institution":"Iplant",
  "position":"Programmer, Senior"
 }
    ]
}
```

⚠ Multi-byte characters appear as **\u0XXX** after being encoded by PHP's implementation of a JSON encoder.

Searching by first or last name

Issue an HTTP GET request to https://fermin.iplantcollaborative.org/api/v1/users/name/PARTIAL_OR_FULL_FIRSTNAME(+LASTNAME).

If you include both first and last names, it must be URL-encoded, as in the following example for John Adams:

https://user.iplantcollaborative/api/v1/users/username/john+adams

Request

```
GET [https://user.iplantcollaborative.org/api/v1/users/name/michael]
```

Response

```
[
    "users":
 {
  "id":"2",
  "username":"jadams",
  "firstname":"John",
  "lastname":"Adams",
  "email":"jadams@example.org"
 }
]
```

Searching by email address

Issue an HTTP GET request to https://user.iplantcollaborative.org/api/v1/users/email/PARTIAL_OR_FULL_EMAIL

Request

```
GET [https://fermin.iplantcollaborative.org/api/v1/users/email/@example.org]
```

Response

```
[
    "users":
 {
  "id":"2",
  "username":"mgatto30",
  "firstname":"Michael",
  "lastname":"Gatto",
  "email":"mgatto30@example.org"
 }
]
```

No users found

If no users are found, the API returns an empty JSON array with an HTTP 404 response code, as in the following response:

Response

Code Block

```
[

   "users": {}

]
```

Adding a service to a user's account

Request

```
POST [https://elliot.iplantcollaborative.org/api/v1/service/coge/add/mgatto]
```

Response

```
[
    "service":
        {"Request to add coge to mgatto's account was successful."}
]
```

Troubleshooting errors and diagnostics

General errors

Errors are returned to API clients in JSON format with HTTP Response codes in the 4xx and 5xx ranges.

- The initial key is "error".
- Sub-keys to classify errors include: "HTTP", "Auth", "Exception" and "Validation".

Invalid HTTP method: *Method not supported*

```
{"error: {"HTTP": "Method not supported; only GET requests are accepted"}}
```

Search term too short: *HTTP Status is 400: Bad Request*

```
{"error": {"Validation: "Search terms must have at least 3 characters"}}
```

Authentication errors

HTTP Status is 401: Not Authorized

```
{"error": {"Auth: "Invalid API Key"}}
```

Invalid Secret code: *Invalid API Secret*

```
{"error": {"Auth: "Invalid API Secret"}}
```

Unregistered API client: *API Client not yet Approved for Access*

```
{"error": {"Auth: "API Client not yet Approved for Access"}}
```

Unrecognized IP address: *Unrecognized API Client*

This error may be returned when the request was sent from an IP address that has not been registered with the client:

```
{"error": {"Auth: "Unrecognized API Client"}}
```

PHP server error: *HTTP Status is 500: Server Error*

```
{"error":
    {"Exception":
        {
     "code":0,
     "message":"[
  Semantical Error
     ]  line 0, col 71 near 'i.name as institution, pf.position\n': Error: 'i' is not
defined.",
        "file":"\/var\/www\/Doctrine\/ORM\/Query\/QueryException.php",
     "line":47
        }
    }
}
```