

[APIv2](#)

[APIv2](#)

- APIv2
 - API v2 Overview
 - APIv2 Swagger Documentation
 - Accessing Swagger
 - Viewing APIv2 Information
 - Testing Endpoints

[API v2 Overview](#)

APIv2 is ProVision's currently supported RESTful API version. APIv2 improvements include:

- HTTP Basic Authentication
- Use of HTTP Methods (GET, PUT, POST, etc.)
- Supports JSON payloads
- Additional endpoints and ProVision functionality

To test APIv2 queries, you may:

1. Use a browser extension REST client, such as [Postman](#)
2. Access ProVision's APIv2 Swagger documentation from your ProVision instance (*instance*/dev/swagger), which provides the ability to test inputs and responses using your ProVision instance data.

[APIv2 Swagger Documentation](#)

[Accessing Swagger](#)

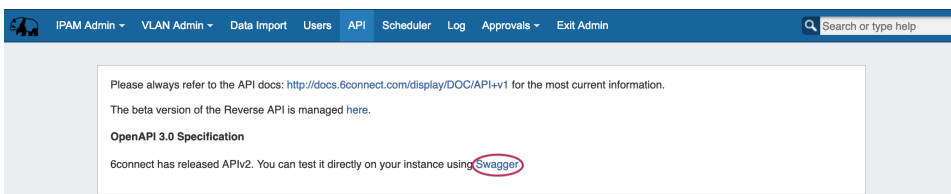
Public APIv2 documentation is located at <https://cloud.6connect.com/APIv2/>.

APIv2 documentation includes:

- [IPAM API](#)
Includes actions for LIRs, IP aggregate and block management, VLAN, IP Rules, and SWIP.
- [Resource API](#)
Includes actions for managing the [ProVision Resource System](#).
The resource API provides CRUD endpoints for resources, resource attributes, resource attachments and resource backups.
- [DNS API](#)
ProVision DNS API allows you to manage DNS Zones, Records, Servers, Groups and ACLS.
- [Users API](#)
Includes actions for ProVision Users, permissions and actions.
- [Usergroups API](#)
Includes actions for ProVision Groups, permissions and actions
- [Scheduler API](#)
The API Allows you to easily schedule tasks.
- [API Composer Platform](#)
API Composer Platform (ACP) is an additional module in ProVision to help automate frequently used combinations of calls.

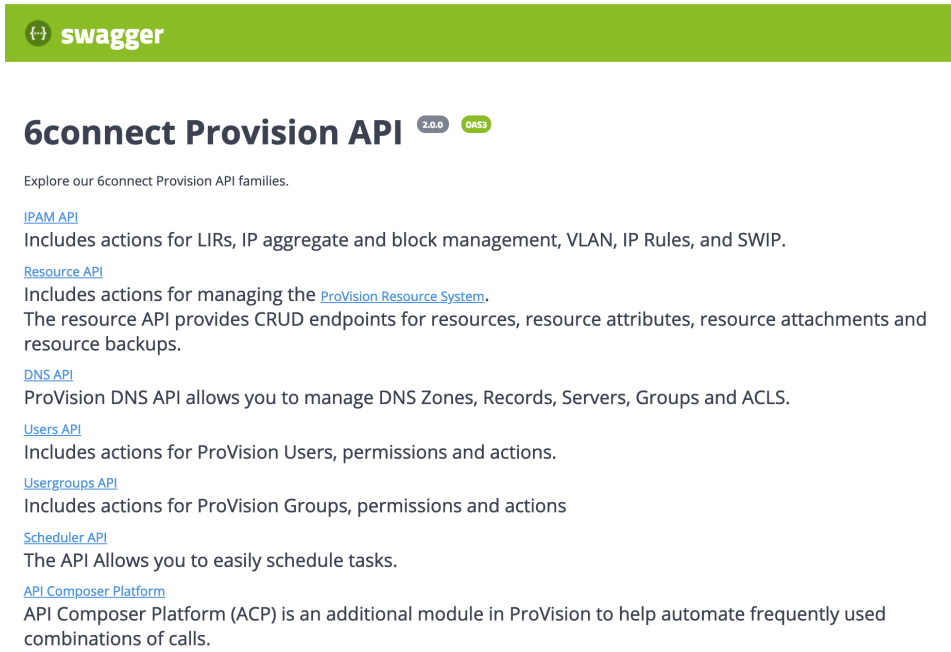
Existing customers may access APIv2 documentation from your ProVision instance (user must have Admin permissions):

1. Log into your ProVision instance.
2. Go to the Admin area of ProVision and click on the [API](#) Tab.
3. Under "OpenAPI 3.0 Specification" click the Swagger link provided.



Viewing APIv2 Information

1. On the 6connect Provision API Swagger home page, click on the name link for the API family that you wish to browse (IPAM, Resource, DNS, etc).



6connect Provision API 2.0.0 OAS3

Explore our 6connect Provision API families.

[IPAM API](#)
Includes actions for LIRs, IP aggregate and block management, VLAN, IP Rules, and SWIP.

[Resource API](#)
Includes actions for managing the [ProVision Resource System](#).
The resource API provides CRUD endpoints for resources, resource attributes, resource attachments and resource backups.

[DNS API](#)
ProVision DNS API allows you to manage DNS Zones, Records, Servers, Groups and ACLS.

[Users API](#)
Includes actions for ProVision Users, permissions and actions.

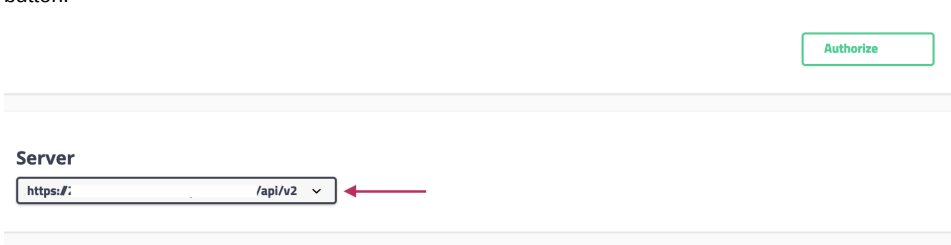
[Usergroups API](#)
Includes actions for ProVision Groups, permissions and actions

[Scheduler API](#)
The API Allows you to easily schedule tasks.

[API Composer Platform](#)
API Composer Platform (ACP) is an additional module in ProVision to help automate frequently used combinations of calls.

2. Once on an API Family page, verify that the displayed server name is correct for your instance/local server.

In most situations, only one ProVision instance/server will be displayed, with authentication already provided from your ProVision login. If your ProVision session has ended, or the server changed, you may need to re-provide ProVision credentials by clicking the "Authorize" button.



Authorize

Server

https://... /api/v2

3. Scroll further down the page and begin reviewing available APIv2 calls and details. Clicking on any call will expand it to view parameter details - you can even test call responses (using your instance data) by clicking "Try it Out"!

The detail information includes a description, parameter list (required parameters are marked with a *), and response information
default

GET

/ipam/lirs GET Lirs

POST

/ipam/lirs Create LIR

GET

/ipam/lirs/{id} Retrieve LIR

Returns information on a single LIR.

Parameters

Try it out

Name	Description
id * required string (path)	ID of the LIR

Responses

Code	Description	Links
200	successful operation	No links
400	Bad Request	No links
401	Invalid credentials	No links

4. Some calls that involve a JSON request body payload (PUT, PATCH, etc) will display "Example Value" and "Model" information under a "Request Body" section - additional parameter descriptions may be displayed under "Model" Information.

Clicking on "Example Value" will show an example of a JSON request body for that call.

Example Value

Model

```
{
  "name": "TestLIR",
  "rir": "ARIN",
  "asn": 20202,
  "parent_id": 0,
  "entities": [
    {
      "string":
    }
  ],
  "type": "entry",
  "section": "string",
  "custom_id": 0
}
```

Clicking on "Model" will display details and descriptions of the request body parameters, if available.

Example Value

Model

put_lir

name	string example: TestLIR Name of the LIR resource
rir	string example: ARIN The RIR for the LIR. Accepted values are "ARIN", "RIPE", "LACNIC", "AFRINIC", "APNIC", and "1918".
asn	integer(\$int64) example: 20202 The ASN (Autonomous System Number) for the LIR
parent_id	integer(\$int64)
entities	> [...]
type	string example: entry Type of resource - Updating a LIR will always be "entry"
section	string Section of the resource object
custom_id	integer(\$int64)

5. Additional "Model" examples are available at the bottom of the page with additional descriptive information.

At the bottom of the page, click on "Models".

GET

/ipam/vlans/{id} Retrieve VLAN

GET

/ipam/{size}/sizes GET IPAM Sizes

Models

Then, click on the "Model" you

wish to view. Some models may contain additional information that you can expand to view, such as valid values for a parameter. In the example below, the circled "array" will display valid RIR values.

Models

```
retrieve_vlan_output > (...)
```

```
get_netblock_resource_hierarchy_output > (...)
```

oneOf_direct_assign_1 ✓

```
resource_id integer($int64)
example: 1234
The ID of the resource the block is assigned to

generic_code integer($int64)
example: Code 2
User-defined block code as defined in Admin-IPAM settings: Generic Code Per Block Name

lir_id integer($int64)
example: 101
The numeric ID of an LIR resource the block should be linked to

rirs string
Regional internet registry
Enum:
  > Array [ 6 ]
region_id integer($int64)
example: 2
The numeric ID of a region

tags string
example: customer,vpn
Come-separated list of tags to filter by. If used in conjunction with 'search', performs the search operation and then filters results by the provided tag. Use with tagsMode to specify filter approach.

tags_mode string
Denotes how the "tags" parameter is handled:
```

Testing Endpoints

You may test queries in Swagger by using the "Try it out" button for any call.

1. Navigate to the call that you want to try out.
2. Expand the call to view its details, then click the "Try it out" button.

GET /ipam/lirs/{id} Retrieve LIR

Returns information on a single LIR.

Parameters

Name	Description
id • required string (path)	ID of the LIR

Try it out

3. Input the desired parameters to test, and click "Execute".

GET /ipam/lirs/{id} Retrieve LIR

Returns information on a single LIR.

Parameters

Name	Description
id • required string (path)	ID of the LIR

1234

Execute

Responses

If the call is a method that uses a JSON request body, you will have the option to edit the body text in the "Example Value" box - when done, click "Execute".

Example Value Model

```
{
  "name": "TestLIR",
  "rir": "ARIN",
  "asn": 20202,
  "parent_id": 0,
  "entitlements": [
    "string"
  ],
  "type": "entry",
  "section": "string",
  "custom_id": 0
}
```

Execute

4. The example response will display under "Responses" after being executed.

Responses	
Curl curl -X GET "https://2-dev.6connect.com/qa-7.1.0-obf/api/v2/tpam/lirs/14591" -H "accept: */*"	
Request URL https://2-dev.6connect.com/qa-7.1.0-obf/api/v2/tpam/lirs/14591	
Server response	
Code	Details
200	Response body <pre>{ "id": "14591", "name": "abc", "slug": "abc-2", "rprt": "1234", "qso": "123", "entities": [{ "mnt_by": "abc", "redef_c": "abc", "secb_c": "abc", "mnt_by_password": "", "mnt_by_password": "*****" }] }</pre> Response headers <pre>access-control-allow-headers: access-control-allow-methods: GET, PUT, POST, PATCH, OPTIONS, DELETE access-control-allow-origin: * access-control-expose-headers: X-Total-Count, Location cache-control: no-store, no-cache, must-revalidate connection: Keep-Alive content-encoding: gzip content-length: 135 content-type: application/json;charset=utf-8 date: Fri, 22 Mar 2019 02:25:42 GMT expires: Thu, 19 Nov 1981 08:52:00 GMT keep-alive: timeout=5, max=100 pragma: no-cache server: Apache/2.4.18 (Ubuntu) vary: Accept-Encoding x-totals-check: valid</pre>