

Module REST API

Modules and module clients can also expose their own REST API.

REST endpoint

It is mandatory that the module's REST API is accessed through C1 Core. Like this, there is one central point to connect to for all modules.

Path

The module's REST path in C1 Core always starts with `/api/v1/modules/<module name>/<custom path>`. The path passed to the module is derived from this REST path by removing `/modules/<module name>`. The path for module clients is very similar: `/api/v1/module_clients/<module client ID>/<custom path>`. Here `/module_clients/<module client ID>` is removed when passed to the module client.

Supported HTTP methods

The following HTTP methods are currently supported: `DELETE`, `GET`, `PATCH`, `POST` and `PUT`. More methods can be implemented if required.

Response format

The response must be encoded as JSON.

RPC tunneling

As modules are connected to C1 Core using a binary RPC protocol, REST requests must be tunneled through that. This is done by generating a RPC request from the REST call. The method name looks like this:

```
REST/<HTTP method><space character><module REST path>
```

So let's say the HTTP method is `POST` and the REST path accessed on C1 Core is `/api/v1/module/my-module/my/custom/path`. Then the method name will be:

```
REST/POST /api/v1/my/custom/path
```

A payload for the REST request is passed as first parameter to the method. The module or module client can translate back this RPC request to an actual REST request.

The original HTTP headers are passed as second parameter to this RPC method.