Arigi REST API

2018-12-14

Table of Contents

1 Principle of Operation

In this document, to *set* means to create or update an object. Objects are both created and updated using the http put method and every request is idempotent. Set operations typically return `204 No Content` on success, unless otherwise specified. This also means that to update an attribute on an object you need to post the complete object.

Operations that are described as *get* will return a single JSON object. Operations that are described as *list* will instead return a JSON array of objects. In both cases the expected successful status is `200 OK` with other codes indicating an error.

All delete operations use the http delete method and return `204 No Content` on success. Delete operations are not idempotent, in that they will instead return `404 Not Found` if the object to be deleted does not exist. The client may choose to interpret this as success in that the object in question does not exist after the operation.

All data exchanged is in JSON format. All attribute names follow the `lowerCamelCase` convention.

It is useful to use the `arigi` CLI when exploring the REST API. Specifically, running `arigi --api-verbose ...any command...` will show, in addition to the command results, the exact operations performed, data sent, and data received.

2 Data Model

Arigi uses an internal data model based on the following objects:

Devices, representing Syncthing instances. These are identified by their (Syncthing) device ID, and must have an API key and API port.

Templates, which contain configuration fragments that can be evaluated and applied to devices.

Tags, that can be attached to devices and templates in order to tie them together.

3 Authentication

Authentication in the API uses tokens. A token is acquired by the `login` method using a regular Arigi username and password. The token is then passed in the `Authorization` header as the `Bearer`scheme.

`POST` **`/api/v1/login`**

```
{
  "username": "admin",
  "password": "arigi"
}
```

Login Request

```
{
  "result":"ok",
  "token":"MTUwOTE3NzMz...L8XrRbgCngaFabirdX0="
}
```

Login Response

Subsequent API requests should then include the following header:

```
Authorization: Bearer MTUwOTE3NzMz...L8XrRbgCngaFabirdX0=
```

Note that this scheme is only secure over a secure channel such as HTTPS. Arigi uses HTTPS by default.

4 Device Management

4.1 List Devices

`GET` **`/api/v1/devices`**

Returns a list of devices.

```
[
  {
      "deviceID": "T5ZNYVY-VDRQ2HH-FP26UHC-FO6NOJ3-YQWK7AV-...",
      "apiKey": "uRT9cSDVnaKe6F3GEN2YycEwiqZHtNjv",
      "apiPort": 8384,
      "apiAddress": "localhost",
      "label": "s1",
      "createdAt": "2017-10-28T10:06:48.087317052+02:00",
      "updatedAt": "2017-10-28T10:06:48.087317052+02:00"
  },
  ...
]
```

List Devices Response

4.2 Get Device

`GET` **`/api/v1/devices/<device ID>`**

```
{
  "deviceID": "T5ZNYVY-VDRQ2HH-FP26UHC-FO6NOJ3-YQWK7AV-...",
  "apiKey": "uRT9cSDVnaKe6F3GEN2YycEwiqZHtNjv",
  "apiPort": 8384,
  "apiAddress": "localhost",
  "label": "s1",
  "createdAt": "2017-10-28T10:06:48.087317052+02:00",
  "updatedAt": "2017-10-28T10:06:48.087317052+02:00"
}
```

Get Device Response

4.3 Set Device

`PUT` **`/api/v1/devices/<device ID>`**

```
{
  "deviceID": "T5ZNYVY-VDRQ2HH-FP26UHC-FO6NOJ3-YQWK7AV-...",
  "apiKey": "uRT9cSDVnaKe6F3GEN2YycEwiqZHtNjv",
  "apiPort": 8384,
  "apiAddress": "localhost",
  "label": "s1"
}
```

Set Device Response

The *apiAddress* attribute is optional - when blank or missing, Arigi will instead use dynamic discovery to find the device address. The *deviceID* attribute is optional but must then match the device ID in the URL if set.

4.4 Delete Device

`DELETE` **`/api/v1/devices/<device ID>`**

The response is `204 No Content` on successfull delete.

4.5 Get Device Status

`GET` **`/api/v1/devices/<device ID>/status`**

This endpoint mirrors Syncthing's API call to get device status.


```
{
  "myID": "T5ZNYVY-VDRQ2HH-FP26UHC-FO6NOJ3-YQWK7AV-...",
  "alloc": 15395616,
  "cpuPercent": 0.019787337440146772,
  "discoveryEnabled": true,
  "discoveryMethods": 8,
  "goRoutines": 65,
  "pathSeparator": "/",
  "sys": 33286392,
  "tilde": "/Users/jb",
  "uptime": 71,
  "createdAt": "2017-10-28T10:14:43.721492134+02:00",
  "updatedAt": "2017-10-28T10:14:43.721492134+02:00"
}
```


Get Device Status Response

4.6 Get Device Version

`GET` **`/api/v1/devices/<device ID>/version`**

This endpoint mirrors Syncthing's API call to get the device version.


```
{
  "device": "T5ZNYVY-VDRQ2HH-FP26UHC-FO6NOJ3-YQWK7AV-...",
  "arch": "amd64",
  "codename": "Dysprosium Dragonfly",
  "longVersion": "syncthing v0.14.40-rc.3+5-gf8015c9a ...",
  "os": "darwin",
  "version": "v0.14.40-rc.3+5-gf8015c9a",
  "createdAt": "2017-10-28T10:14:43.721879991+02:00",
  "updatedAt": "2017-10-28T10:14:43.721879991+02:00"
}
```


Get Device Version Response

4.7 List Device Tags

`GET` **`/api/v1/devices/`*`<device ID>`*`/tags`**

Returns the current set of tags for the device.

```
[
  "s1"
]
```

List Device Tags Response

4.8 Tag a Device

`PUT` **`/api/v1/devices/`*`<device ID>`*`/tags/`*`<tag>`**

The response contains the resulting set of tags for the device, same format as the list tags call.

4.9 Untag a Device

`DELETE` **`/api/v1/devices/`*`<device ID>`*`/tags/`*`<tag>`**

The response contains the resulting set of tags for the device, which is possibly the empty list.

5 Template Management

Templates are the fragments that make up the configuration system. Each template has an *id* which is automatically assigned by Arigi and uniquely identifies the template in the API, and a *label* which is human readable and not necessarily unique (although that is probably best).

A template has the following attributes:

label

Human friendly label.

priority

Numeric priority index.

op

Operation to apply.

key

Key into the configuration object where we apply the template operation.

template

HJSON template.

5.1 List Templates

`GET` **`/api/v1/templates`**

Returns the set of template names.

```
[
  {
    "id": 1,
    "createdAt": "2017-10-28T10:20:05.535808929+02:00",
    "updatedAt": "2017-10-28T10:20:05.535808943+02:00",
    "label": "Disable NAT",
    "priority": 50,
    "key": "options.natEnabled",
    "op": "set",
    "template": "false\n"
  },
  ...
]
```

List Templates Response

5.2 Set Template

`POST` **`/api/v1/templates`**

`PUT` **`/api/v1/templates/`*`<template ID>`***

Creates or updates and existing template.

```
{
  "label": "Disable NAT",
  "priority": 50,
  "key": "options.natEnabled",
  "op": "set",
```

```
  "template": "false\n"
}
```

Set Template Response

5.3 Get Template

`GET` **`/api/v1/templates/`*`<template ID>`***

```
{
  "id": 1,
  "createdAt": "2017-10-28T10:20:05.535808929+02:00",
  "updatedAt": "2017-10-28T10:20:05.535808943+02:00",
  "label": "Disable NAT",
  "priority": 50,
  "key": "options.natEnabled",
  "op": "set",
  "template": "false\n"
}
```

Get Template Response

5.4 Delete Template

`GET` **`/api/v1/templates/`*`<template ID>`***

The response is `204 No Content` on successfull delete.

5.5 Tag a Template

`GET` **`/api/v1/templates/`*`<template ID>`*`/tags/`*`<tag>`***

The response contains the resulting set of tags for the template.

```
[
  "s1"
]
```

Tag Template Response

5.6 Untag a Template

GET **/api/v1/templates/*<template ID>*/tags/*<tag>***

The response contains the resulting set of tags for the template, which is possibly the empty list.

> TBD: Example

5.7 Evaluate a Template (Existing Template)

GET **/api/v1/templates/*<template ID>***

Evaluates the template against an automatically chosen example device and returns the results, or an error. The request body is empty.

> TBD: Example

5.8 Evaluate a Template (New Template)

GET **/api/v1/templates/evalue**

Evaluates the template against an automatically chosen example device and returns the results, or an error. The template to evaluate is sent as the request body.

> TBD: Example

6 Tag Management

6.1 List Tags

GET **/api/v1/tags**

Returns the complete set of tags currently in existence.

```
[
  "s1"
]
```

List Tags Response

6.2 Delete Tag

`GET` **`/api/v1/tags/`*`<tag>`***

The tag is removed from all devices and all templates. Returs the list of remaining tags.

6.3 List Devices for Tag

`GET` **`/api/v1/tags/`*`<tag>`*`/devices`**

> ⓘ TBD: Example

6.4 List Templates for Tag

`GET` **`/api/v1/tags/`*`<tag>`*`/templates`**

> ⓘ TBD: Example

7 Folder Status

7.1 List Folders

`GET` **`/api/v1/folders`**

Returns the list of folder IDs and their labels.

```
[
  {
    "folder": "2fyjq-jrfpr",
    "label": "Foton/2017"
  },
  {
    "folder": "4vcqq-sciqk",
```

```
    "label": "Transfer"
  },
  ...
]
```

7.2 List Folder Status

GET **/api/v1/folders/*<folder ID>***

Returns the list of folder summaries, one per device that shares the folder.

```
[
  {
    "device": "GB33MBV-EKIA4IA-RNNU2YP-MHYRNL7-AWPB5JR-...",
    "folder": "lightroom",
    "createdAt": "2017-09-27T20:30:39.824251Z",
    "updatedAt": "2017-10-28T09:04:51.017775Z",
    "globalBytes": 21775074403,
    "globalDeleted": 11,
    "globalDirectories": 21486,
    "globalFiles": 24727,
    "globalSymlinks": 0,
    "localBytes": 21775074403,
    "localDeleted": 11,
    "localDirectories": 21486,
    "localFiles": 24727,
    "localSymlinks": 0,
    "inSyncBytes": 21775074403,
    "inSyncFiles": 24727,
    "needBytes": 0,
    "needDeletes": 0,
    "needDirectories": 0,
    "needFiles": 0,
    "needSymlinks": 0,
    "ignorePatterns": false,
    "sequence": 95173,
    "version": 95173
  },
  ...
]
```

List Folder Status Response