swiftsc Documentation

Release 0.7.0

Kouhei Maeda

October 08, 2015

Contents

1 swiftsc is simple client library of OpenStack Swift **3**
- 1.1 Requirements . . . 3
- 1.2 Setup . . . 3
- 1.3 See also . . . 4

2 Basic usage **5**
- 2.1 Initialize. . . . 5
- 2.2 Create container . . . 5
- 2.3 Create object . . . 6

3 References **7**

4 History **11**
- 4.1 0.7.0 (2015-08-30) . . . 11
- 4.2 0.6.5 (2015-05-17) . . . 11
- 4.3 0.6.4 (2015-05-16) . . . 11
- 4.4 0.6.3 (2015-03-17) . . . 11
- 4.5 0.6.2 (2015-03-11) . . . 12
- 4.6 0.6.1 (2015-03-09) . . . 12
- 4.7 0.6.0 (2015-03-06) . . . 12
- 4.8 0.5.5 (2014-11-19) . . . 12
- 4.9 0.5.4 (2014-11-16) . . . 12
- 4.10 0.5.3 (2014-11-15) . . . 12
- 4.11 0.5.2 (2014-05-10) . . . 12
- 4.12 0.5.1 (2013-11-06) . . . 13
- 4.13 0.5 (2013-07-27) . . . 13
- 4.14 0.4 (2013-06-13) . . . 13
- 4.15 0.3 (2013-06-03) . . . 13
- 4.16 0.2.2 (2013-05-20) . . . 13
- 4.17 0.2.1 (2013-05-17) . . . 13
- 4.18 0.2 (2013-05-10) . . . 13
- 4.19 0.1.3 (2013-05-08) . . . 13
- 4.20 0.1.2 (2013-05-07) . . . 13
- 4.21 0.1.1 (2013-05-05) . . . 14
- 4.22 0.1 (2013-05-02) . . . 14

5 Indices and tables **15**

Python Module Index **17**

Contents:

CHAPTER 1

swiftsc is simple client library of OpenStack Swift

This tool is simple client library of OpenStack Swift. This tool is intended to be used in the module and Python script other. The main purpose of this tool is used as a core module for backup tool.

1.1 Requirements

- Python 2.7 over or Python 3.3 over
- requests 0.12.1 later
- python-magic 5.x in debian package or python-magic 0.4.3 later of PyPI

1.2 Setup

```
$ pip install --user swiftsc
or
(venv)$ pip install swiftsc
```

1.2.1 workaround of Python 3.3

When not using debian package of python-magic, current version(0.4.3) is not support python 3.3. Python 3.3 is supported by committed after one of the tag of 0.3.

https://github.com/ahupp/python-magic/commit/d033eb46a8ace66cf795c54168a197228e47ce9e

So you must install from github until next version will release.:

```
$ git clone https://github.com/ahupp/python-magic
$ cd python-magic
$ sudo python setup.py install
$ cd
$ git clone https://github.com/mkouhei/swiftsc
$ cd swiftsc
$ sudo python setup.py install
```

1.3 See also

- OpenStack Object Storage Developer Guide
- Requests

CHAPTER 2

Basic usage

swiftsc is OpenStack Swift simple client.

2.1 Initialize.

2.1.1 Swift tempauth

Using Swift tempauth.

```
$ python
>>> from swiftsc import Client
>>> client = Client(auth_uri='https://swift.example.org/auth/v1.0',
... username='myaccount', password='passw0rd')
```

2.1.2 KeyStone Auth

Using Identity API v2.0

```
$ python
>>> from swiftsc import Client
>>> client = Client(auth_uri='https://keystone.example.org/v2.0/tokens',
... username='user_name', password='passw0rd', tenant_name='tenant_name')
```

Using Identity API (>= v0.7.0). You must pass `user id` to `username` and, `project id` to `tenant_name`.

```
$ python
>>> from swiftsc import Client
>>> client = Client(auth_uri='https://keystone.example.org/v3/auth/tokens',
... username='user_id', password='passw0rd', tenant_name='project_id')
```

2.2 Create container

```
>>> client.containers.create(name='mycontainer')
<Response [201]>
>>> client.containers.list().json()
[{'bytes': 1403088360, 'count': 5, 'name': 'container-a'},
 {'bytes': 393429510, 'count': 11, 'name': 'container-b'},
 {'bytes': 410389320, 'count': 11, 'name': 'container-c'},
```

```
 ...
 {'bytes': 9690876040, 'count': 57, 'name': 'container-x'},
 {'bytes': 0, 'count': 0, 'name': 'mycontainer'}]
>>> client.containers.detail('mycontainer')
<Response [204]>
```

2.3 Create object

```
>>> client.containers.container('mycontainer')
>>> with open('/tmp/dummy.txt', 'w') as f:
...     f.write('sample')
>>> client.containers.objects.create(name='dummy',
... file_path='/tmp/dummy.txt')
<Response [201]>
>>> client.containers.objects.list().json()
[{'bytes': 5,
  'content_type': 'text/plain',
  'hash': '275876e34cf609db118f3d84b799a790',
  'last_modified': '2015-03-18T09:32:40.311040',
  'name': 'dummy'}]
>>> client.containers.detail('mycontainer')
<Response [200]>
>>> client.containers.detail('mycontainer').text.split()
['dummy']
>>> client.containers.detail('dummy').text
'sample'
>>> client.containers.objects.copy('dummy', 'dummy2')
<Response [201]>
>>> client.containers.objects.delete('dummy')
<Response [204]>
>>> client.containers.objects.list().json()
[{u'bytes': 6,
  u'content_type': u'text/plain',
  u'hash': u'5e8ff9bf55ba3508199d22e984129be6',
  u'last_modified': u'2015-03-18T09:39:36.247430',
  u'name': u'dummy2'}]
```

CHAPTER 3

References

swiftsc.client module.

class swiftsc.client.**Client**(*auth_uri=None*, *uri=None*, *username=None*, *password=None*, *token=None*, *tenant_name=None*, *verify=True*, *timeout=5.0*)

Bases: object

The *Client* object.

This provides REST connection including tempauth or KeyStone Auth.:

```
>>> from swiftsc import Client
>>> client = Client(auth_uri='https://swift.example.org/auth/v1.0',
... username='swiftuser', password='passw0rd')
```

Parameters

- **auth_uri** (*str*) – tempauth URL or KeyStone URL
- **uri** (*str*) – Storage URL (required token)
- **username** (*str*) – tempauth or KeyStone username
- **password** (*str*) – tempauth or KeyStone password
- **token** (*str*) – Auth token
- **tenant_name** (*str*) – KeyStone tenant name

class swiftsc.client.**Container**(*obj*)

Bases: swiftsc.client._CRUD

Swift container resources.

```
>>> client.containers.list().json()
[{'bytes': 1403088360, 'count': 5, 'name': 'container-a'},
 {'bytes': 393429510, 'count': 11, 'name': 'container-b'},
 {'bytes': 410389320, 'count': 11, 'name': 'container-c'},
 ...
 {'bytes': 9690876040, 'count': 57, 'name': 'container-x'}]
```

change_timeout(*timeout*)

Change timeout.

Change timeout to other than "5.0"

container(*container_name*)
Set container name and create instances.

The instance has the attributes of *Container*, as follows.

•objects: *Object*

Parameters **container_name** (*str*) – container name

create(***kwargs*)
Create or replace resource.

Return type *requests.Response*

Returns Response of metadata single resource.

Parameters ****kwargs** – parameters

delete(*obj_id*)
Delete resource.

Return type *requests.Response*

Returns Response of deleting a single resource.

Parameters **obj_id** (*str*) – resource id (or resource name)

detail(*obj_id=None*)
Show/Get a single resource.

Return type *requests.Response*

Returns Response of detail single resource.

Parameters **obj_id** (*str*) – resource id (or resource name)

list()
List collection of resources.

Return type *requests.Response*

Returns Response of list collection.

no_verify()
Ignore SSL Cert Verification.

Change `verify` to `False`.

show_metadata(*obj_id=None*)
Show metadata.

Return type *requests.Response*

Returns Response of metadata single resource.

Parameters **obj_id** (*str*) – resource id (or resource name)

update_metadata(*obj_id*, ***kwargs*)
Create, Update (or delete) metadata.

Return type *requests.Response*

Returns Response of updating a single resource.

Parameters

- **obj_id** (*str*) – resource id (or resource name)

- ****kwargs** – keyword arguments of method

class swiftsc.client.**Object**(*obj*)

Bases: swiftsc.client._CRUD

Objects resources.

```
>>> client.containers.container('container-a')
>>> client.containers.objects.list().json()
[{'bytes': 0,
  'content_type': 'application/octet-stream',
  'hash': 'd41d8cd98f00b204e9800998ecf8427e',
  'last_modified': '2015-03-05T07:57:17.450440',
  'name': 'test'},
 {'bytes': 225280,
  'content_type': 'application/x-tar',
  'hash': '9aa58f7a3fca9853c26a048eda407c71',
  'last_modified': '2013-07-18T09:56:30.989920',
  'name': 'test.tgz'},
 {'bytes': 22,
  'content_type': 'text/plain',
  'hash': '4cc6982f37c06ec4eb378e916cfbd289',
  'last_modified': '2015-03-05T09:34:42.935400',
  'name': 'test2'}]
```

change_timeout(*timeout*)

Change timeout.

Change timeout to other than "5.0"

copy(*src_object_name*, *dest_object_name*)

Copy object.

Return type *requests.Response*

Returns Response of copy object

create(***kwargs*)

Create object.

Parameters ****kwargs** – parameters for creating object

Return type *requests.Response*

Returns Response of create object

delete(*obj_id*)

Delete resource.

Return type *requests.Response*

Returns Response of deleting a single resource.

Parameters **obj_id** (*str*) – resource id (or resource name)

detail(*obj_id=None*)

Show/Get a single resource.

Return type *requests.Response*

Returns Response of detail single resource.

Parameters **obj_id** (*str*) – resource id (or resource name)

list()
List collection of resources.

Return type *requests.Response*

Returns Response of list collection.

no_verify()
Ignore SSL Cert Verification.

Change `verify` to `False`.

object(*object_name*)
Set object name.

Parameters **object_name** (*str*) – object name

show_metadata(*obj_id=None*)
Show metadata.

Return type *requests.Response*

Returns Response of metadata single resource.

Parameters **obj_id** (*str*) – resource id (or resource name)

update_metadata(*obj_id*, ***kwargs*)
Create, Update (or delete) metadata.

Return type *requests.Response*

Returns Response of updating a single resource.

Parameters

- **obj_id** (*str*) – resource id (or resource name)
- ****kwargs** – keyword arguments of method

swiftsc utility module.

swiftsc.utils.**check_mimetype**(*filepath*)
Check mimetype of file.

Return type str

Returns mimetype

Parameters **filepath** (*str*) – target filename path

swiftsc.utils.**check_mimetype_buffer**(*fileobj*)
Check mimetype of file.

Return type str

Returns mimetype

Parameters **fileobj** – target filename path

swiftsc.utils.**retrieve_info_from_buffer**(*file_object*)
Check mimetype of file object.

Return type tuple

Returns mimetype, content length, data

Parameters **file_object** – target file object

CHAPTER 4

History

4.1 0.7.0 (2015-08-30)

- Support Python 3.50-rc2.
- Discard the functions not related Client class.
- Supports Identity API v3 of OpenStack KeyStone.
- Fixes serviceCatlog key for the public cloud using OpenStack.
- Changes development status.
- Change Sphinx theme to sphinx_rtd_theme.

4.2 0.6.5 (2015-05-17)

- Changes HTTpretty to requests_mock

4.3 0.6.4 (2015-05-16)

- Supports wheel
- Adds extras_require to setup.py
- Applies pep257
- Pins httpretty version to 0.8.6
- Workaround Read the docs theme
- Changed to generate requirements.txt automatically
- Adds basic usage new API

4.4 0.6.3 (2015-03-17)

- Fail creating object from stdin

4.5 0.6.2 (2015-03-11)

- Suppressed InsecureRequestWarning

4.6 0.6.1 (2015-03-09)

- Fixed storage uri
- Supported PyPy
- Updated pre-commit hook script
- Applied inherited-members to Sphinx documentation
- Changed attribute of automodule

4.7 0.6.0 (2015-03-06)

- Added new client API
- Added Sphinx documentation

4.8 0.5.5 (2014-11-19)

- Appended argument timeout to change value

4.9 0.5.4 (2014-11-16)

- Bump version

4.10 0.5.3 (2014-11-15)

- Unsuppored Python3.2
- Fixed #12 Read timed out
- Integrated pylint, pychecker to tox
- Fixed dependencies
- Fixed almost violations of pylint

4.11 0.5.2 (2014-05-10)

- refactoring
- support python 3.4, PyPI
- apply tox for unit test

4.12 0.5.1 (2013-11-06)

- fixes fail to load _io module in Python 2.6
- support Python 2.6

4.13 0.5 (2013-07-27)

- support input file from stdin pipe, redirect
- detect “python-magic” debian package in setup.py

4.14 0.4 (2013-06-13)

- support Python 3.2, 3.3

4.15 0.3 (2013-06-03)

- support auth of keystone

4.16 0.2.2 (2013-05-20)

- support to ignore verifying the SSL certficate

4.17 0.2.1 (2013-05-17)

- change api of retrieve_object(), response inserted boolean before content

4.18 0.2 (2013-05-10)

- add is_object method
- change api of is_container, response is changed status code to boolean

4.19 0.1.3 (2013-05-08)

- fixes the response is not invalid with Response.json in requests 1.0 later

4.20 0.1.2 (2013-05-07)

- set default timeout as 5.0

4.21 0.1.1 (2013-05-05)

- fixes failed to upload without "Content-Length" when uploading empty file

4.22 0.1 (2013-05-02)

- first release

CHAPTER 5

Indices and tables

- genindex
- modindex
- search

S

swiftsc.__init__, 5
swiftsc.client, 7
swiftsc.utils, 10

C

change_timeout() (swiftsc.client.Container method), 7
change_timeout() (swiftsc.client.Object method), 9
check_mimetype() (in module swiftsc.utils), 10
check_mimetype_buffer() (in module swiftsc.utils), 10
Client (class in swiftsc.client), 7
Container (class in swiftsc.client), 7
container() (swiftsc.client.Container method), 7
copy() (swiftsc.client.Object method), 9
create() (swiftsc.client.Container method), 8
create() (swiftsc.client.Object method), 9

D

delete() (swiftsc.client.Container method), 8
delete() (swiftsc.client.Object method), 9
detail() (swiftsc.client.Container method), 8
detail() (swiftsc.client.Object method), 9

L

list() (swiftsc.client.Container method), 8
list() (swiftsc.client.Object method), 9

N

no_verify() (swiftsc.client.Container method), 8
no_verify() (swiftsc.client.Object method), 10

O

Object (class in swiftsc.client), 9
object() (swiftsc.client.Object method), 10

R

retrieve_info_from_buffer() (in module swiftsc.utils), 10

S

show_metadata() (swiftsc.client.Container method), 8
show_metadata() (swiftsc.client.Object method), 10
swiftsc.__init__ (module), 5
swiftsc.client (module), 7
swiftsc.utils (module), 10

U

update_metadata() (swiftsc.client.Container method), 8
update_metadata() (swiftsc.client.Object method), 10