

---

# **Apache Libcloud Documentation**

*Release 1.10.0*

**The Apache Software Foundation**

October 02, 2015



<b>1</b>	<b>Installation</b>	<b>3</b>
<b>2</b>	<b>Usage</b>	<b>5</b>
<b>3</b>	<b>Using a custom CA certificate bundle</b>	<b>7</b>
<b>4</b>	<b>API Documentation</b>	<b>9</b>
<b>5</b>	<b>Changelog</b>	<b>11</b>
<b>6</b>	<b>License</b>	<b>13</b>



---

**Note:** This is an unofficial client library which is not affiliated with Yubico in any way.

---

Yubico Client is a Python library for verifying Yubikey One Time Passwords (OTPs) based on the validation protocol version 2.0.

- Yubikey website: <http://www.yubico.com>
- Yubikey documentation: <http://www.yubico.com/developers/intro/>
- Validation Protocol Version 2.0 FAQ: <http://www.yubico.com/develop/open-source-software/web-api-clients/server/>
- Validation Protocol Version 2.0 description: <https://github.com/Yubico/yubikey-val/wiki/ValidationProtocolV20>



---

## Installation

---

---

**Note:** Package has been recently renamed from *yubico* to *yubico-client* and the main module has been renamed from *yubico* to *yubico\_client*. This was done to avoid naming conflicts and make creation of distribution specific packages easier.

---

Latest stable version can be installed from PyPi using pip:

```
pip install yubico-client
```

If you want to install latest development version, you can install it from this Git repository:

```
pip install -e https://github.com/Kami/python-yubico-client.git@master#egg=yubico_client
```





---

## Usage

---

---

**Note:** It may take up until 5 minutes until all validation servers know about your newly generated client.

---

1. Generate your client id and secret key. This can be done on the [Yubico website](#).
2. Use the generated client id and key to instantiate the client

Keep in mind that using a secret key is optional, but you are strongly recommended to do so. If a secret key is not specified, you can still verify the token, but the response message signature won't be verified.

Single mode:

```
from yubico_client import Yubico

client = Yubico('client id', 'secret key')
client.verify('otp')
```

The `yubico_client.Yubico.verify()` method will return `True` if the provided OTP is valid (`STATUS=OK`).

Multi mode:

```
from yubico_client import Yubico

client = Yubico('client id', 'secret key')
client.verify_multi(['otp 1', 'otp 2', 'otp 3'])
```

The `yubico_client.Yubico.verify()` method will return `True` if all of the provided OTPs are valid (`STATUS=OK`).

Both methods can also throw one of the following exceptions:

- `StatusCodeError` - server returned `REPLAYED_OTP` status code
- `SignatureVerificationError` - server response message signature verification failed
- `InvalidClientIdError` - client with the specified id does not exist (server returned `NO_SUCH_CLIENT` status code)
- `Exception` - server returned one of the following status values: `BAD_OTP`, `BAD_SIGNATURE`, `MISSING_PARAMETER`, `OPERATION_NOT_ALLOWED`, `BACKEND_ERROR`, `NOT_ENOUGH_ANSWERS`, `REPLAYED_REQUEST` or no response was received from any of the servers in the specified time frame (default `timeout = 10` seconds)



---

## Using a custom CA certificate bundle

---

**Note:** This functionality changed slightly in version 1.9.0. Before this version, you could use a custom CA bundle by setting `yubico_client.yubico.CA_CERTS_BUNDLE_PATH` module level variable.

---

By default, the library will try to find and use a standard CA bundle which is available on your operating system for verifying the server SSL certificate.

If you want to use a custom CA bundle (e.g. you want to use a more restrictive bundle which only contains root CA certificate for `api.yubico.com` domain), you can do so by passing `ca_certs_bundle_path` argument to the `Yubico` class constructor.

Version 1.9.0 and above:

```
from yubico_client import Yubico

client = Yubico('client id', 'secret key',
               ca_certs_bundle_path='/home/to/my/bundle.pem')
```

Versions prior to 1.9.0:

```
from yubico_client import yubico

yubico.CA_CERTS_BUNDLE_PATH = '/home/to/my/bundle.pem'
client = yubico.Yubico('client id', 'secret key')
```

Keep in mind that this bundle needs to be in PEM format.



---

## API Documentation

---

For API documentation, please see the [API Documentation](#) page.



---

**Changelog**

---

For changelog, please see the [Changelog](#) page.





---

**License**

---

Yubico Client is distributed under the [3-Clause BSD License](#).