
PyIntelOwl

Release 4.4.5

Matteo Lodi

Nov 16, 2023

USAGE

1	Installation	3
2	Usage as CLI	5
3	Usage as SDK/library	7
4	Index	9
4.1	Modules	9
4.2	Tests	9
5	Indices and tables	11
	Index	13

Robust Python **SDK** and **Command Line Client** for interacting with [IntelOwl](#) API.

INSTALLATION

```
$ pip install pyintelowl
```


USAGE AS CLI

On successful installation, The pyintelowl entryscript should be directly invocable. For example,

```
$ pyintelowl
Usage: pyintelowl [OPTIONS] COMMAND [ARGS]...

Options:
  -d, --debug  Set log level to DEBUG
  --version    Show the version and exit.
  -h, --help   Show this message and exit.

Commands:
  analyse          Send new analysis request
  analyzer-healthcheck  Send healthcheck request for an analyzer...
  config          Set or view config variables
  connector-healthcheck  Send healthcheck request for a connector
  get-analyzer-config  Get current state of `analyzer_config.json` from...
  get-connector-config  Get current state of `connector_config.json` from...
  get-playbook-config  Get current state of `playbook_config.json` from...
  jobs            Manage Jobs
  tags            Manage tags
```

Configuration:

You can use set to set the config variables and get to view them.

Listing 1: [View on asciinema](#)

```
$ pyintelowl config set -k 4bf03f20add626e7138f4023e4cf52b8 -u "http://localhost:80"
$ pyintelowl config get
```

Hint: The CLI would is well-documented which will help you navigate various commands easily. Invoke pyintelowl -h or pyintelowl <command> -h to get help.

USAGE AS SDK/LIBRARY

```
1 from pyintelowl import IntelOwl, IntelOwlClientException
2 obj = IntelOwl(
3     "4bf03f20add626e7138f4023e4cf52b8",
4     "http://localhost:80",
5     None,
6 )
7 """
8 obj = IntelOwl(
9     "<your_api_key>",
10    "<your_intelowl_instance_url>",
11    "optional<path_to_pem_file>"
12    "optional<proxies>"
13 )
14 """
15
16 try:
17     ans = obj.get_analyzer_configs()
18     print(ans)
19 except IntelOwlClientException as e:
20     print("Oh no! Error: ", e)
```

Tip: We very much **recommend** going through the `pyintelowl.pyintelowl.IntelOwl` docs.

4.1 Modules

4.1.1 IntelOwl class

4.1.2 IntelOwlClientException class

```
class pyintelowl.exceptions.IntelOwlClientException(*args, **kwargs)
    Bases: RequestException
    property error_detail: Union[Dict, AnyStr]
```

4.2 Tests

4.2.1 Configuration

Some tests require file samples, which can be found in the encrypted folder `tests/test_files.zip` (password: “infected”). Unzip the archive in `tests/test_files` folder before running the tests.

Please remember that these are dangerous malware! They come encrypted and locked for a reason! Do NOT run them unless you are absolutely sure of what you are doing! They are to be used only for launching specific tests that require them (`__send_analysis_request`)

- With the following constants in `__init__.py`, you can customize your tests:
 - **MOCKING_CONNECTIONS**: Mock connections to external API to test functions without a real connection or a valid API Key.
- If you prefer to use custom inputs for tests, you can change the following constants:
 - **TEST_JOB_ID**
 - **TEST_HASH**
 - **TEST_URL**
 - **TEST_IP**
 - **TEST_DOMAIN**
 - **TEST_GENERIC**
 - **TEST_FILE**
 - **TEST_FILE_HASH**

4.2.2 Launch Tests

- The test requirements are specified in the `test-requirements.txt` file. Install them using,

```
$ pip3 install -r test-requirements.txt
```

- Launch the tests using `tox`:

```
$ tox
```

INDICES AND TABLES

- genindex
- modindex

INDEX

E

`error_detail` (*pyintelowl.exceptions.IntelOwlClientException*
property), 9

I

`IntelOwlClientException` (*class in pyin-*
telowl.exceptions), 9