PyIntelOwl

Release v3.1.4

Matteo Lodi

Oct 19, 2021

CONTENTS

Robust Python **SDK** and **Command Line Client** for interacting with IntelOwl API.

CHAPTER

ONE

INSTALLATION

```
$ pip install pyintelowl
```

CHAPTER

TWO

USAGE AS CLI

On successful installation, The `pyintelowl` entryscript should be directly invokable. 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
config               Set or view config variables
get-analyzer-config  Get current state of `analyzer_config.json` from the...
jobs                 Manage Jobs
tags                 Manage tags
```

2.1 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.

CHAPTER

THREE

USAGE AS SDK/LIBRARY

```python
from pyintelowl import IntelOwl, IntelOwlClientException
obj = IntelOwl(
    "4bf03f20add626e7138f4023e4cf52b8",
    "http://localhost:80",
    None,
)
"""
obj = IntelOwl(
    "<your_api_key>",
    "<your_intelowl_instance_url>",
    "optional<path_to_pem_file>"
)
"""

try:
    ans = obj.get_analyzer_configs(1)
    print(ans)
except IntelOwlClientException as e:
    print("Oh no! Error: ", e)
```

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

CHAPTER

FOUR

API CLIENT DOCS

4.1 pyintelowl modules

4.1.1 pyintelowl.pyintelowl module

class pyintelowl.pyintelowl.**IntelOwl**(*token: str*, *instance_url: str*, *certificate: Optional[str] = None*, *logger: Optional[logging.Logger] = None*, *cli: bool = False*)

Bases: object

ask_analysis_availability(*md5: str*, *analyzers_needed: List[str]*, *run_all_available_analyzers: bool = False*, *check_reported_analysis_too: bool = False*) → Dict

Search for already available analysis.

Endpoint: /api/ask_analysis_availability

Parameters

- **md5** (*str*) – md5sum of the observable or file
- **analyzers_needed** (*List[str]*) – list of analyzers to invoke
- **run_all_available_analyzers** (*bool, optional*) –
- **True, runs all compatible analyzers. Defaults to False.** (*If*) –
- **check_reported_analysis_too** (*bool, optional*) –
- **against all existing jobs. Defaults to False.** (*Check*) –

Raises *IntelOwlClientException* – on client/HTTP error

Returns JSON body

Return type Dict

create_tag(*label: str*, *color: str*)

Creates new tag by sending a POST Request Endpoint: /api/tags

Parameters

- **label** (*[str]*) – [Label of the tag to be created]
- **color** (*[str]*) – [Color of the tag to be created]

delete_job_by_id(*job_id: int*) → bool

Send delete job request.

Method: DELETE Endpoint: /api/jobs/{job_id}

Parameters **job_id** (*int*) – id of job to kill

Raises *IntelOwlClientException* – on client/HTTP error

Returns deleted or not

Return type Bool

delete_tag_by_id(*tag_id: int*) → bool

Send delete tag request.

Method: DELETE Endpoint: `/api/tags/{tag_id}`

Parameters **tag_id** (*int*) – id of tag to delete

Raises *IntelOwlClientException* – on client/HTTP error

Returns deleted or not

Return type Bool

edit_tag(*tag_id: Union[int, str]*, *label: str*, *color: str*)

Edits existing tag by sending PUT request Endpoint: `api/tags`

Parameters

- **id** (*[int]*) – [Id of the existing tag]
- **label** (*[str]*) – [Label of the tag to be created]
- **color** (*[str]*) – [Color of the tag to be created]

get_all_jobs() → List[Dict[str, Any]]

Fetch list of all jobs.

Endpoint: `/api/jobs`

Raises *IntelOwlClientException* – on client/HTTP error

Returns List of jobs

Return type List[Dict[str, Any]]

get_all_tags() → List[Dict[str, str]]

Fetch list of all tags.

Endpoint: `/api/tags`

Raises *IntelOwlClientException* – on client/HTTP error

Returns List of tags

Return type List[Dict[str, str]]

get_analyzer_configs()

Get current state of *analyzer_config.json* from the IntelOwl instance.

Endpoint: `/api/get_analyzer_configs`

get_job_by_id(*job_id: Union[int, str]*) → Dict[str, Any]

Fetch job info by ID. Endpoint: `/api/job/{job_id}`

Parameters **job_id** (*Union[int, str]*) – Job ID

Raises *IntelOwlClientException* – on client/HTTP error

Returns JSON body.

Return type Dict[str, Any]

static get_md5(*to_hash: AnyStr*, *type_='observable'*) → str

Returns md5sum of given observable or file object.

Parameters

- **to_hash** (*AnyStr*) – either an observable string, file contents as bytes or path to a file
- **type** (*Union["observable", "binary", "file"], optional*) – *observable*, *binary*, *file*. Defaults to "observable".

Raises *IntelOwlClientException* – on client/HTTP error

Returns md5sum

Return type str

get_tag_by_id(*tag_id: Union[int, str]*) → Dict[str, str]

Fetch tag info by ID.

Endpoint: `/api/tag/{tag_id}`

Parameters **tag_id** (*Union[int, str]*) – Tag ID

Raises *IntelOwlClientException* – on client/HTTP error

Returns Dict with 3 keys: *id*, *label* and *color*.

Return type Dict[str, str]

kill_running_job(*job_id: int*) → bool

Send kill_running_job request.

Method: PATCH Endpoint: `/api/jobs/{job_id}/kill`

Parameters **job_id** (*int*) – id of job to kill

Raises *IntelOwlClientException* – on client/HTTP error

Returns killed or not

Return type Bool

logger: logging.Logger

send_analysis_batch(*rows: List[Dict]*)

Send multiple analysis requests. Can be mix of observable or file analysis requests.

Used by the pyintelowl CLI.

Parameters **rows** (*List[Dict]*) – Each row should be a dictionary with keys, *value*, *type*, *analyzers_list*, *run_all force_privacy*, *private_job*, *disable_external_analyzers*, *check*.

send_file_analysis_request(*analyzers_requested: List[str]*, *filename: str*, *binary: bytes*, *force_privacy: bool = False*, *private_job: bool = False*, *disable_external_analyzers: bool = False*, *run_all_available_analyzers: bool = False*, *runtime_configuration: Optional[Dict] = None*, *tags: Optional[List[int]] = None*) → Dict

Send analysis request for a file.

Endpoint: `/api/send_analysis_request`

Parameters

- **analyzers_requested** (*List[str]*) – List of analyzers to invoke
- **filename** (*str*) – Filename
- **binary** (*bytes*) – File contents as bytes

- **force_privacy** (*bool, optional*) – Disable analyzers that can leak info. Defaults to `False`.
- **private_job** (*bool, optional*) – Limit view permissions to your groups . Defaults to `False`.
- **disable_external_analyzers** (*bool, optional*) – Disable analyzers that use external services. Defaults to `False`.
- **tags** (*List[int]*) – List of tags associated with this job
- **run_all_available_analyzers** (*bool, optional*) – If True, runs all compatible analyzers. Defaults to `False`.
- **runtime_configuration** (*Dict, optional*) – Overwrite configuration for analyzers. Defaults to `{}`.

Raises *IntelOwlClientException* – on client/HTTP error

Returns JSON body

Return type Dict

send_observable_analysis_request(*analyzers_requested: List[str], observable_name: str, force_privacy: bool = False, private_job: bool = False, disable_external_analyzers: bool = False, run_all_available_analyzers: bool = False, runtime_configuration: Optional[Dict] = None, tags: Optional[List[int]] = None*) → Dict

Send analysis request for an observable.

Endpoint: `/api/send_analysis_request`

Parameters

- **analyzers_requested** (*List[str]*) – List of analyzers to invoke
- **observable_name** (*str*) – Observable value
- **force_privacy** (*bool, optional*) – Disable analyzers that can leak info. Defaults to `False`.
- **private_job** (*bool, optional*) – Limit view permissions to your groups . Defaults to `False`.
- **disable_external_analyzers** (*bool, optional*) – Disable analyzers that use external services. Defaults to `False`.
- **tags** (*List[int]*) – List of tags associated with this job
- **run_all_available_analyzers** (*bool, optional*) – If True, runs all compatible analyzers. Defaults to `False`.
- **runtime_configuration** (*Dict, optional*) – Overwrite configuration for analyzers. Defaults to `{}`.

Raises *IntelOwlClientException* – on client/HTTP error

Returns JSON body

Return type Dict

property session: requests.sessions.Session

Internal use only.

4.1.2 pyintelowl.exceptions module

exception pyintelowl.exceptions.**IntelOwlClientException**

Bases: Exception

exception pyintelowl.exceptions.**IntelOwlInvalidAPITokenException**

Bases: Exception

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex

PYTHON MODULE INDEX

p

INDEX