OpenCTI client for Python

Release 3.2.4

May 14, 2020

Contents:

The pycti library is designed to help OpenCTI users and developers to interact with the OpenCTI platform GraphQL API.

The Python library requires Python >= 3.

CHAPTER 1

Getting Started

1.1 Installation

Please install the latest pycti version available from PyPI:

```
$ pip3 install pycti
```

1.2 Initialization

The main class contains all what you need to interact with the platform, you just have to initialize it:

```
# coding: utf-8
from pycti import OpenCTIApiClient

# OpenCTI initialization
opencti_api_client = OpenCTIApiClient(api_url, api_token, log_level, ssl_verify)
```

CHAPTER 2

pycti

2.1 Functions

- *get_config_variable()*: [summary]

pycti.**get_config_variable**(*env_var*, *yaml_path*, *config={}*, *isNumber=False*)
[summary]

Parameters

- **env_var** (str) – environnement variable name
- **yaml_path** (str) – path to yaml config
- **config** (Dict) – client config dict, defaults to {}
- **isNumber** (Optional[bool]) – specify if the variable is a number, defaults to False

Return type Union[bool, int, None, str]

2.2 Classes

- *OpenCTIApiClient*: Main API client for OpenCTI
- *OpenCTIApiConnector*: OpenCTIApiConnector
- *OpenCTIApiJob*: OpenCTIApiJob
- *ConnectorType*: An enumeration.

- `OpenCTIConnector`: Main class for OpenCTI connector
- `OpenCTIConnectorHelper`: Python API for OpenCTI connector
- `Tag`: Undocumented.
- `MarkingDefinition`: Undocumented.
- `ExternalReference`: Undocumented.
- `KillChainPhase`: Undocumented.
- `StixEntity`: Undocumented.
- `StixDomainEntity`: Undocumented.
- `StixObservable`: Undocumented.
- `StixRelation`: Undocumented.
- `StixSighting`: Undocumented.
- `StixObservableRelation`: Undocumented.
- `Identity`: Undocumented.
- `ThreatActor`: Undocumented.
- `IntrusionSet`: Undocumented.
- `Campaign`: Undocumented.
- `Incident`: Undocumented.
- `Malware`: Undocumented.
- `Tool`: Undocumented.
- `Vulnerability`: Undocumented.
- `AttackPattern`: Undocumented.
- `CourseOfAction`: Undocumented.
- `Report`: Undocumented.
- `Note`: Undocumented.
- `Opinion`: Undocumented.
- `Indicator`: Undocumented.
- `OpenCTIStix2`: Python API for Stix2 in OpenCTI
- `ObservableTypes`: These are the possible values for OpenCTI's observable types.
- `CustomProperties`: These are the custom properties used by OpenCTI.

class pycti.**OpenCTIApiClient**(*url*, *token*, *log_level='info'*, *ssl_verify=False*)

Main API client for OpenCTI

Parameters

- **url** (*str*) – OpenCTI API url
- **token** (*str*) – OpenCTI API token
- **log_level** (*str, optional*) – log level for the client
- **ssl_verify** (*bool, optional*) –

Inheritance

OpenCTIApiClient

fetch_opencti_file(*fetch_uri*, *binary=False*)
get file from the OpenCTI API

Parameters

- **fetch_uri** (*str*) – download URI to use
- **binary** (*bool, optional*) – [description], defaults to False

Returns returns either the file content as text or bytes based on *binary*

Return type str or bytes

get_logs_worker_config()
get the logsWorkerConfig

return: the logsWorkerConfig rtype: dict

get_token()
Get the API token

Returns returns the configured API token

Return type str

health_check()
submit an example request to the OpenCTI API.

Returns returns *True* if the health check has been successful

Return type bool

log(*level*, *message*)
log a message with defined log level

Parameters

- **level** (*str*) – must be a valid logging log level (debug, info, warning, error)
- **message** (*str*) – the message to log

not_empty(*value*)
check if a value is empty for str, list and int

Parameters **value** (*str or list or int*) – value to check

Returns returns *True* if the value is one of the supported types and not empty

Return type bool

process_multiple(*data*, *with_pagination=False*)
processes data returned by the OpenCTI API with multiple entities

Parameters

- **data** – data to process
- **with_pagination** (*bool, optional*) – whether to use pagination with the API, defaults to False

Returns returns either a dict or list with the processes entities

Return type list or dict

process_multiple_fields (*data*)
processes data returned by the OpenCTI API with multiple fields

Parameters **data** (*dict*) – data to process

Returns returns the data dict with all fields processed

Return type dict

process_multiple_ids (*data*)
processes data returned by the OpenCTI API with multiple ids

Parameters **data** – data to process

Returns returns a list of ids

Return type list

query (*query*, *variables={}*)
submit a query to the OpenCTI GraphQL API

Parameters

- **query** (*str*) – GraphQL query string
- **variables** (*dict, optional*) – GraphQL query variables, defaults to {}

Returns returns the response json content

Return type Any

resolve_role (*relation_type*, *from_type*, *to_type*)
resolves the role for a specified entity

Parameters

- **relation_type** (*str*) – input relation type
- **from_type** (*str*) – entity type
- **to_type** (*str*) – entity type

Returns returns the role mapping

Return type dict

set_token (*token*)
set the request header with the specified token

Parameters **token** (*str*) – OpenCTI API token

upload_file (***kwargs*)
upload a file to OpenCTI API

Parameters ****kwargs** – arguments for file upload (required: *file_name* and *data*)

Returns returns the query respons for the file upload

Return type dict

class pycti.**OpenCTIApiConnector**(*api*)

Inheritance

OpenCTIApiConnector

list()

list available connectors

Returns return dict with connectors

Return type dict

ping(*connector_id*, *connector_state*)

pings a connector by id and state

Parameters

- **connector_id** (*str*) – the connectors id
- **connector_state** (Any) – state for the connector

Returns the response pingConnector data dict

Return type dict

register(*connector*)

register a connector with OpenCTI

Parameters **connector** (OpenCTIConnector) – *OpenCTIConnector* connector object

Returns the response registerConnector data dict

Return type dict

class pycti.**OpenCTIApiJob**(*api*)

Inheritance

OpenCTIApiJob

initiate_job(*work_id*)

initiate a job with the API

Parameters **work_id** (*str*) – id for the job

Returns the id for the initiateJob

Return type str

update_job(*job_id*, *status*, *messages*)
update a job with the API

Parameters

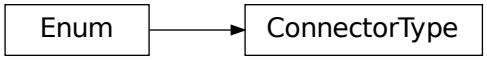
- **job_id** (*str*) – job id
- **status** (*str*) – job status
- **messages** (*list*) – job messages

Returns the id for the updateJob

Return type str

class pycti.**ConnectorType**
An enumeration.

Inheritance



class pycti.**OpenCTIConnector**(*connector_id*, *connector_name*, *connector_type*, *scope*)
Main class for OpenCTI connector

Parameters

- **connector_id** (*str*) – id for the connector (valid uuid4)
- **connector_name** (*str*) – name for the connector
- **connector_type** (*str*) – valid OpenCTI connector type (see *ConnectorType*)
- **scope** (*str*) – connector scope

Raises **ValueError** – if the connector type is not valid

Inheritance

to_input()
connector input to use in API query

Returns dict with connector data

Return type dict

class pycti.**OpenCTIConnectorHelper**(*config*)
Python API for OpenCTI connector

Parameters **config** (*dict*) – Dict standard config

Inheritance

OpenCTIConnectorHelper

static check_max_tlp(*tlp*, *max_tlp*)
check the allowed TLP levels for a TLP string

Parameters

- **tlp** (*str*) – string for TLP level to check
- **max_tlp** (*str*) – the highest allowed TLP level

Returns list of allowed TLP levels

Return type list

date_now()
get the current date (UTC)

Returns current datetime for utc

Return type datetime

get_state()
get the connector state

Returns returns the current state of the connector if there is any

Return type

listen(*message_callback*)
listen for messages and register callback function

Parameters **message_callback** (*Callable[[Dict], List[str]]*) – callback function to process messages

Return type None

send_stix2_bundle(*bundle*, *entities_types=None*, *update=False*, *split=True*)
send a stix2 bundle to the API

Parameters

- **bundle** – valid stix2 bundle

- **entities_types** (*list, optional*) – list of entities, defaults to None
- **update** (*bool, optional*) – whether to updated data in the database, defaults to False
- **split** (*bool, optional*) – whether to split the stix bundle before processing, defaults to True

Raises **ValueError** – if the bundle is empty

Returns list of bundles

Return type list

set_state (*state*)
sets the connector state

Parameters **state** (*dict*) – state object

Return type None

split_stix2_bundle (*bundle*)
splits a valid stix2 bundle into a list of bundles

Parameters **bundle** – valid stix2 bundle

Raises **Exception** – if data is not valid JSON

Returns returns a list of bundles

Return type list

static stix2_create_bundle (*items*)
create a stix2 bundle with items

Parameters **items** – valid stix2 items

Returns JSON of the stix2 bundle

Return type

static stix2_deduplicate_objects (*items*)
deduplicate stix2 items

Parameters **items** – valid stix2 items

Returns de-duplicated list of items

Return type list

stix2_get_embedded_objects (*item*)
gets created and marking refs for a stix2 item

Parameters **item** – valid stix2 item

Returns returns a dict of created_by_ref of object_marking_refs

Return type dict

stix2_get_entity_objects (*entity*)
process a stix2 entity

Parameters **entity** – valid stix2 entity

Returns entity objects as list

Return type list

stix2_get_relationship_objects(*relationship*)
get a list of relations for a stix2 relationship object

Parameters **relationship** – valid stix2 relationship

Returns list of relations objects

Return type list

stix2_get_report_objects(*report*)
get a list of items for a stix2 report object

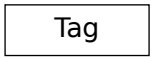
Parameters **report** – valid stix2 report object

Returns list of items for a stix2 report object

Return type list

class pycti.**Tag**(*opencti*)

Inheritance



class pycti.**MarkingDefinition**(*opencti*)

Inheritance

class pycti.**ExternalReference**(*opencti*)

Inheritance

ExternalReference

class pycti.**KillChainPhase**(*opencti*)

Inheritance

KillChainPhase

class pycti.**StixEntity**(*opencti*)

Inheritance

StixEntity

class pycti.**StixDomainEntity**(*opencti*, *file*)

Inheritance

StixDomainEntity

class pycti.**StixObservable**(*opencti*)

Inheritance

StixObservable

class pycti.**StixRelation**(*opencti*)

Inheritance

StixRelation

class pycti.**StixSighting**(*opencti*)

Inheritance

StixSighting

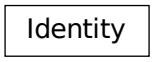
class pycti.**StixObservableRelation**(*opencti*)

Inheritance

StixObservableRelation

class pycti.**Identity**(*opencti*)

Inheritance

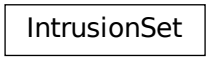


class pycti.**ThreatActor**(*opencti*)

Inheritance

class pycti.**IntrusionSet**(*opencti*)

Inheritance

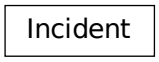


class pycti.**Campaign**(*opencti*)

Inheritance

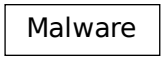
class pycti.**Incident**(*opencti*)

Inheritance



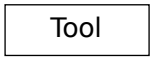
class pycti.**Malware**(*opencti*)

Inheritance



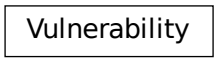
class pycti.**Tool**(*opencti*)

Inheritance



class pycti.**Vulnerability**(*opencti*)

Inheritance



class pycti.**AttackPattern**(*opencti*)

Inheritance

AttackPattern

class pycti.**CourseOfAction**(*opencti*)

Inheritance

CourseOfAction

class pycti.**Report**(*opencti*)

Inheritance

Report

class pycti.**Note**(*opencti*)

Inheritance

Note

class pycti.**Opinion**(*opencti*)

Inheritance

Opinion

class pycti.**Indicator**(*opencti*)

Inheritance

Indicator

class pycti.**OpenCTIStix2**(*opencti*)

Python API for Stix2 in OpenCTI

Parameters **opencti** – OpenCTI instance

Inheritance

OpenCTIStix2

check_max_marking_definition(*max_marking_definition_entity*, *entity_marking_definitions*)

checks if a list of marking definitions conforms with a given max level

Parameters

- **max_marking_definition_entity**(*str, optional*) – the maximum allowed marking definition level
- **entity_marking_definitions** (*list*) – list of entities to check

Returns *True* if the list conforms with max marking definition

Return type bool

convert_markdown(*text*)

converts input text to markdown style code annotation

Parameters **text** (*str*) – input text

Returns sanitized text with markdown style code annotation

Return type str

extract_embedded_relationships (*stix_object*, *types=None*)
extracts embedded relationship objects from a stix2 entity

Parameters

- **stix_object** – valid stix2 object
- **types** (*list, optional*) – list of stix2 types, defaults to None

Returns embedded relationships as dict

Return type dict

filter_objects (*uuids*, *objects*)
filters objects based on UUIDs

Parameters

- **uuids** (*list*) – list of UUIDs
- **objects** (*list*) – list of objects to filter

Returns list of filtered objects

Return type list

format_date (*date*)
converts multiple input date formats to OpenCTI style dates

Parameters **date** – input date

Returns OpenCTI style date

Return type datetime

import_bundle_from_file (*file_path*, *update=False*, *types=None*)
import a stix2 bundle from a file

Parameters

- **file_path** (*str*) – valid path to the file
- **update** (*bool, optional*) – whether to updated data in the database, defaults to False
- **types** (*list, optional*) – list of stix2 types, defaults to None

Returns list of imported stix2 objects

Return type List

import_bundle_from_json (*json_data*, *update=False*, *types=None*)
import a stix2 bundle from JSON data

Parameters

- **json_data** – JSON data
- **update** (*bool, optional*) – whether to updated data in the database, defaults to False
- **types** (*list, optional*) – list of stix2 types, defaults to None

Returns list of imported stix2 objects

Return type List

import_object (*stix_object*, *update=False*, *types=None*)
import a stix2 object

Parameters

- **stix_object** – valid stix2 object
- **update** (*bool, optional*) – whether to updated data in the database, defaults to False
- **types** (*list, optional*) – list of stix2 types, defaults to None

Returns list of imported stix2 objects

Return type list

pick_aliases (*stix_object*)
check stix2 object for multiple aliases and return a list

Parameters **stix_object** – valid stix2 object

Returns list of aliases

Return type list

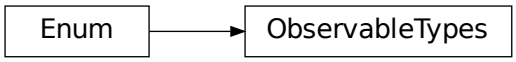
class pycti.**ObservableTypes**
These are the possible values for OpenCTI's observable types.

Use in conjunction with the STIX custom property *x_opencti_observable_type*.

ref: https://github.com/OpenCTI-Platform/opencti/blob/8854c2576dc17da9da54e54b116779bd2131617c/opencti-front/src/private/components/report/ReportAddObservable.js

NOTE: should this be a mapping between the stix2 SDO objects (i.e. stix2/v20/sdo.py)?

Inheritance



class pycti.**CustomProperties**
These are the custom properties used by OpenCTI.

Inheritance

CHAPTER 3

Indices and tables

- genindex
- modindex
- search

Python Module Index

p

Index

P

Q

R

S

T

U

V