OTX to MISP

Release 1.4.2

May 11, 2018

Contents

1 Overview **1**
- 1.1 Installation . . . 1
- 1.2 Documentation . . . 1
- 1.3 Alienvault OTX SDK . . . 1

2 Installation **3**

3 Usage **5**
- 3.1 otx-misp . . . 5

4 Reference **7**
- 4.1 otx_misp . . . 7

5 Contributing **9**
- 5.1 Bug reports . . . 9
- 5.2 Documentation improvements . . . 9
- 5.3 Feature requests and feedback . . . 9
- 5.4 Development . . . 10

6 Authors **11**

7 Changelog **13**
- 7.1 1.4.2 (2018-05-11) . . . 13
- 7.2 1.4.1 (2017-08-25) . . . 13
- 7.3 1.4.0 (2017-08-14) . . . 13
- 7.4 1.3.0 (2017-07-04) . . . 13
- 7.5 1.2.1 (2017-03-31) . . . 14
- 7.6 1.2.0 (2017-03-31) . . . 14
- 7.7 1.1.1 (2017-01-28) . . . 14
- 7.8 1.1 (2016-12-04) . . . 14
- 7.9 1.0.3 (2016-09-10) . . . 14
- 7.10 1.0.2 (2016-09-02) . . . 14
- 7.11 1.0.1 (2016-09-01) . . . 14
- 7.12 1.0.0 (2016-06-21) . . . 14
- 7.13 0.3.0 (2016-06-20) . . . 15
- 7.14 0.2.0 (2016-06-14) . . . 15
- 7.15 0.1.0 (2016-06-14) . . . 15

8 Indices and tables 17

Python Module Index 19

CHAPTER 1

Overview

docs	
package	

Imports Alienvault OTX pulses to a MISP instance.

- Free software: Apache 2.0 license

1.1 Installation

Python 2 support will be dropped soon.

```
pip install otx-misp
```

1.2 Documentation

https://otx-misp.readthedocs.io/

1.3 Alienvault OTX SDK

All files in the *src/otx_misp/otx* directory are Copyright 2015 AlienVault, Inc. and under the Apache 2.0 license.

CHAPTER 2

Installation

At the command line:

```
pip install otx-misp
```

CHAPTER 3

Usage

To use OTX to MISP from the command line:

```
$ otx-misp --help
```

3.1 otx-misp

Downloads OTX pulses and add them to MISP.

```
usage: otx-misp [-h] [-o OTX] [-s SERVER] [-m MISP] [-t TIMESTAMP] [-c CONFIG]
                [-w] [-a] [-u] [-n] [-d] [-v] [--no-tlp] [--discover-tags]
                [--to-ids] [--distribution DISTRIBUTION]
                [--threat-level THREAT_LEVEL] [--analysis ANALYSIS]
                [--author-tag] [--bulk-tag BULK_TAG] [--dedup-titles]
                [--stop-on-error]
```

-h, --help
show this help message and exit

-o <otx>, **--otx** <otx>
Alienvault OTX API key

-s <server>, **--server** <server>
MISP server URL

-m <misp>, **--misp** <misp>
MISP API key

-t <timestamp>, **--timestamp** <timestamp>
Last import as Date/Time ISO format or UNIX timestamp

-c <config>, **--config-file** <config>

-w, --write-config
Write the configuration file

-a, --author
Add the Pulse author name in the MISP Info field

-u, --update-timestamp
Updates the timestamp in the configuaration file

-n, --no-publish
Don't publish the MISP event

-d, --dry-run
Fetch the pulses but don't create MISP events. Use -v[v] to see details.

-v, --verbose
Verbosity, repeat to increase the verbosity level.

--no-tlp
No Traffic Light Protocol tag

--discover-tags
Discover tags to add to MISP events

--to-ids
Mark IOCs as exportable to IDS

--distribution <distribution>
MISP distribution of events (organisation,community,connected,all), default: organisation

--threat-level <threat_level>
MISP threat level of events (high,medium,low,undefined), default: undefined

--analysis <analysis>
MISP analysis state of events (initial,ongoing,completed), default: completed

--author-tag
Add the pulse author as an event tag

--bulk-tag <bulk_tag>
Add a custom tag that will be added to all events (e.g. OTX)

--dedup-titles
Search MISP for an existing event title and update it, rather than create a new one

--stop-on-error
Stop import when an exception is raised

To use OTX to MISP in a project:

```
import otx_misp
```

CHAPTER 4

Reference

4.1 otx_misp

exception otx_misp.**ImportException**

otx_misp.**create_events**(*pulse_or_list*, *author=False*, *server=False*, *key=False*, *misp=False*, *distribution=0*, *threat_level=4*, *analysis=2*, *publish=True*, *tlp=True*, *discover_tags=False*, *to_ids=False*, *author_tag=False*, *bulk_tag=None*, *dedup_titles=False*, *stop_on_error=False*)

Parse a Pulse or a list of Pulses and add it/them to MISP if server and key are present

Parameters

- **pulse_or_list** – a Pulse or list of Pulses as returned by *get_pulses*
- **author** (*Boolean*) – Prepend the author to the Pulse name
- **server** – MISP server URL
- **key** – MISP API key
- **misp** (pymisp.PyMISP) – MISP connection object
- **distribution** – distribution of the MISP event (0-4)
- **threat_level** – threat level of the MISP object (1-4)
- **analysis** – analysis stae of the MISP object (0-2)
- **publish** (*Boolean*) – Is the MISP event should be published?
- **tlp** (*Boolean*) – Add TLP level tag to event
- **discover_tags** (*Boolean*) – discover MISP tags from Pulse tags
- **to_ids** (*Boolean*) – Flag pulse attributes as being sent to an IDS
- **author_tag** (*Boolean*) – Add the pulse author as an event tag
- **bulk_tag** (*String*) – A tag that will be added to all events for categorization (e.g. OTX)

- **dedup_titles** (*Boolean*) – Search MISP for an existing event title and update it, rather than create a new one

Returns a dict or a list of dict with the selected attributes

`otx_misp.`**`get_pulses`**(*otx_api_key*, *from_timestamp=None*)

Get the Pulses from Alienvault OTX

Parameters

- **otx_api_key** (*string*) – Alienvault OTX API key
- **from_timestamp** (`datetime.datetime` or ISO string or Unix timestamp) – only download Pulses after this date/time (None for all Pulses)

Returns a list of Pulses (dict)

`otx_misp.`**`get_pulses_iter`**(*otx_api_key*, *from_timestamp=None*)

Get the Pulses from Alienvault OTX and returns a generator

Parameters

- **otx_api_key** (*string*) – Alienvault OTX API key
- **from_timestamp** (`datetime.datetime` or ISO string or Unix timestamp) – only download Pulses after this date/time (None for all Pulses)

Returns a generator of Pulses (dict)

`otx_misp.`**`misp_server_version`**(*misp*)

Retrieve the MISP instance version

Parameters **misp** (`pymisp.PyMISP`) – MISP connection object

Returns MISP instance version as string

`otx_misp.`**`tag_event`**(*misp*, *event*, *tag*)

Add a tag to a MISP event

Parameters

- **misp** (`pymisp.PyMISP`) – MISP connection object
- **event** – a MISP event
- **tag** – tag to add

Returns None

CHAPTER 5

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

5.1 Bug reports

When reporting a bug please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

5.2 Documentation improvements

OTX to MISP could always use more documentation, whether as part of the official OTX to MISP docs, in docstrings, or even on the web in blog posts, articles, and such.

5.3 Feature requests and feedback

The best way to send feedback is to file an issue at https://github.com/gcrahay/otx_misp/issues.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that code contributions are welcome :)

5.4 Development

To set up *otx_misp* for local development:

1. Fork otx_misp (look for the "Fork" button).
2. Clone your fork locally:

```
git clone git@github.com:your_name_here/otx_misp.git
```

3. Create a branch for local development:

```
git checkout -b name-of-your-bugfix-or-feature
```

   Now you can make your changes locally.

4. Commit your changes and push your branch to GitHub:

```
git add .
git commit -m "Your detailed description of your changes."
git push origin name-of-your-bugfix-or-feature
```

5. Submit a pull request through the GitHub website.

5.4.1 Pull Request Guidelines

If you need some code review or feedback while you're developing the code just make the pull request.

For merging, you should:

1. Update documentation when there's new API, functionality etc.
2. Add a note to `CHANGELOG.rst` about the changes.
3. Add yourself to `AUTHORS.rst`.

CHAPTER 6

Authors

- Gaetan Crahay - https://github.com/gcrahay
- Nick Driver - https://github.com/TheDr1ver
- KALRONG - https://github.com/KALRONG
- obert01 - https://github.com/obert01

CHAPTER 7

Changelog

7.1 1.4.2 (2018-05-11)

- Fix typo in logger name (@TheDr1ver)
- Don't add already attached tag to events
- Tested with Python 3.5 and MISP 2.4.89

7.2 1.4.1 (2017-08-25)

- Fix MISP tag name parsing (@KALRONG)
- Use pulse 'created' date in MISP event (@obert01)

7.3 1.4.0 (2017-08-14)

- Add YARA indicator support

7.4 1.3.0 (2017-07-04)

- Fix dedup function
- Fix TLP tag import
- Don't stop on import error
- Python 2 support warning
- Tested with Python 3.5, MISP 2.4.[71-76], PyMISP 2.4.71

7.5 1.2.1 (2017-03-31)

- Fix Python 3 compatibility

7.6 1.2.0 (2017-03-31)

- Fixes event tagging
- Adds additional tagging options
- Handles empty reference field in OTX pulses

7.7 1.1.1 (2017-01-28)

- Improve Pulse modified field parsing

7.8 1.1 (2016-12-04)

- Fix compatibility with PyMISP >= 2.4.53
- Improve Python 3 support

7.9 1.0.3 (2016-09-10)

- Fix new configuration cloning bug

7.10 1.0.2 (2016-09-02)

- Fix compatibility issue with Python 2.7.6

7.11 1.0.1 (2016-09-01)

- Catch exceptions when disabling SSL warnings

7.12 1.0.0 (2016-06-21)

- First stable version
- Pulse Traffic Light Protocol level added as tag in MISP event
- If the last part of a MISP tag and a Pulse tag are the same, tag the MISP event
- MISP attributes *to_ids* field

7.13 0.3.0 (2016-06-20)

- Fix default handling for distribution, threat_level and analysis parameters
- Better performance: Use OTXv2 generator API and remove some delays

7.14 0.2.0 (2016-06-14)

- Integrate OTXv2 as a subtree.

7.15 0.1.0 (2016-06-14)

- First release on PyPI.

CHAPTER 8

Indices and tables

- genindex
- modindex
- search

Python Module Index

o

otx_misp, 7

Index

Symbols

–analysis <analysis>
    otx-misp command line option, 6
–author-tag
    otx-misp command line option, 6
–bulk-tag <bulk_tag>
    otx-misp command line option, 6
–dedup-titles
    otx-misp command line option, 6
–discover-tags
    otx-misp command line option, 6
–distribution <distribution>
    otx-misp command line option, 6
–no-tlp
    otx-misp command line option, 6
–stop-on-error
    otx-misp command line option, 6
–threat-level <threat_level>
    otx-misp command line option, 6
–to-ids
    otx-misp command line option, 6
-a, –author
    otx-misp command line option, 5
-c <config>, –config-file <config>
    otx-misp command line option, 5
-d, –dry-run
    otx-misp command line option, 6
-h, –help
    otx-misp command line option, 5
-m <misp>, –misp <misp>
    otx-misp command line option, 5
-n, –no-publish
    otx-misp command line option, 6
-o <otx>, –otx <otx>
    otx-misp command line option, 5
-s <server>, –server <server>
    otx-misp command line option, 5
-t <timestamp>, –timestamp <timestamp>
    otx-misp command line option, 5
-u, –update-timestamp
    otx-misp command line option, 6
-v, –verbose
    otx-misp command line option, 6
-w, –write-config
    otx-misp command line option, 5

C

create_events() (in module otx_misp), 7

G

get_pulses() (in module otx_misp), 8
get_pulses_iter() (in module otx_misp), 8

I

ImportException, 7

M

misp_server_version() (in module otx_misp), 8

O

otx-misp command line option
    –analysis <analysis>, 6
    –author-tag, 6
    –bulk-tag <bulk_tag>, 6
    –dedup-titles, 6
    –discover-tags, 6
    –distribution <distribution>, 6
    –no-tlp, 6
    –stop-on-error, 6
    –threat-level <threat_level>, 6
    –to-ids, 6
    -a, –author, 5
    -c <config>, –config-file <config>, 5
    -d, –dry-run, 6
    -h, –help, 5
    -m <misp>, –misp <misp>, 5
    -n, –no-publish, 6
    -o <otx>, –otx <otx>, 5

-s <server>, –server <server>, 5
-t <timestamp>, –timestamp <timestamp>, 5
-u, –update-timestamp, 6
-v, –verbose, 6
-w, –write-config, 5
otx_misp (module), 7

T

tag_event() (in module otx_misp), 8