How to interact with MISP threat sharing platform using standard unix tools



CIRCL
Computer Incident
Response Center
Luxembourg

Team CIRCL
Gérard Wagener
TLP:WHITE

http://www.circl.lu/
Twitter: *@circl_lu*

1 March, 2019



Co-financed by the European Union
Connecting Europe Facility

MISP and starting from a practical use-case

- During a malware analysis workgroup in 2012, we discovered that we worked on the analysis of the same malware.
- We wanted to share information in an easy and automated way **to avoid duplication of work**.
- Christophe Vandeplas (then working at the CERT for the Belgian MoD) showed us his work on a platform that later became MISP.
- A first version of the MISP Platform was used by the MALWG and **the increasing feedback of users** helped us to build an improved platform.
- MISP is now **a community-driven development**.

Development based on practical user feedback

- There are many different types of users of an information sharing platform like MISP:
  - **Malware reversers** willing to share indicators of analysis with respective colleagues.
  - **Security analysts** searching, validating and using indicators in operational security.
  - **Intelligence analysts** gathering information about specific adversary groups.
  - **Law-enforcement** relying on indicators to support or bootstrap their DFIR cases.
  - **Risk analysis teams** willing to know about the new threats, likelyhood and occurences.
  - **Fraud analysts** willing to share financial indicators to detect financial frauds.

MISP Project Overview





Galaxy



warning-lists



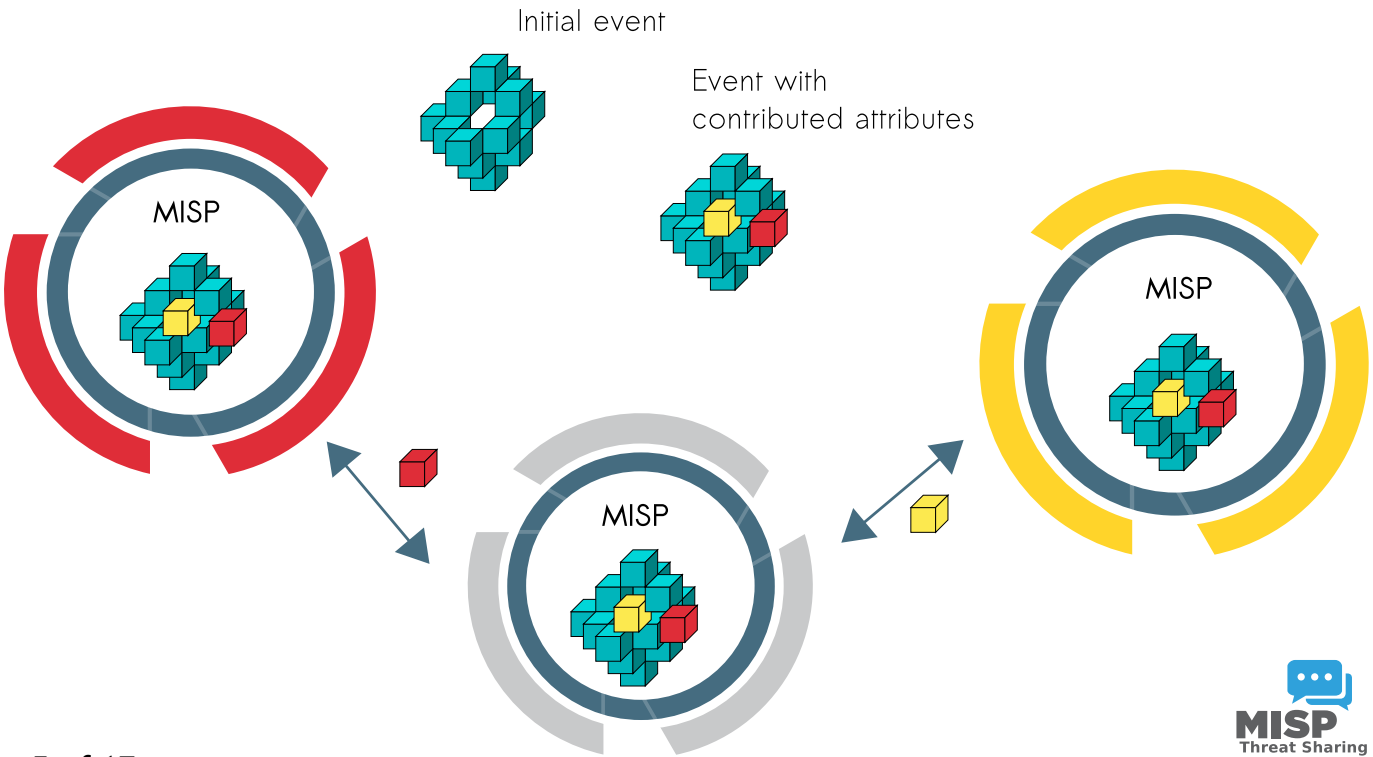
Taxonomies



modules (import, export, enrichment)

MISP core distributed sharing functionality

- MISPs' core functionality is sharing where everyone can be a consumer and/or a contributor/producer."
- Quick benefit without the obligation to contribute.
- Low barrier access to get acquainted to the system.



Events, Objects and Attributes in MISP

- MISP events are encapsulations for contextually linked information
- MISP attributes[1] initially started with a standard set of "cyber security" indicators.
- MISP attributes are purely **based on usage** (what people and organizations use daily).
- Evolution of MISP attributes is based on practical usage & users (e.g. the addition of **financial indicators** in 2.4).
- MISP objects are attribute compositions describing points of data using many facets, constructed along the lines of community and user defined templates.
- Galaxies granularly contextualise, classify & categorise data based on **threat actors**, **preventive measures**, tools used by adversaries.

[1] attributes can be anything that helps describe the intent of the event package from indicators, vulnerabilities or any relevant information

MISP features

- MISP[2] is a threat information sharing free & open source software.
- MISP has **a host of functionalities** that assist users in creating, collaborating & sharing threat information - e.g. flexible sharing groups, **automatic correlation**, free-text import helper, event distribution & proposals.
- Many export formats which support IDSes / IPSes (e.g. Suricata, Bro, Snort), SIEMs (eg CEF), Host scanners (e.g. OpenIOC, STIX, CSV, yara), analysis tools (e.g. Maltego), DNS policies (e.g. RPZ).
- A rich set of MISP modules[3] to add expansion, import and export functionalities.

[2] `https://github.com/MISP/MISP`

[3] `https://www.github.com/MISP/misp-modules`

Give IPs from yesterday not matching warning list

```
curl \
 -d '{ "timestamp": "1d",
       "returnFormat":"text",
       "type":["ip-src","ip-dst"],
       "enforceWarninglist":1}' \
\
 -H "Authorization:␣YOUR\_API\_KEY" \
 -H "Accept:␣application/json" \
 -H "Content-type:␣application/json" \
 -X POST https://your_misp_server.lu/attributes/
    restSearch > last_ips.txt
```

Searching IPs in your logs

```
cat last\_ips.txt | grep "\""␣|␣cut␣-d␣',’␣-f␣5␣|␣tr␣-d
    ␣’"’ > ips.txt

grep -f ips.txt /var/log/apache2/access.log >matches.
    txt

sort -u matches.txt | mail -s "MISP␣hits" info@circl.lu

rm matches.txt
```

Execute these commands as a daily cron job

Using sightings in MISP

- Sightings allow users to notify the community about the activities related to an indicator
- In recent MISP versions, the sighting system supports negative sigthings (FP) and expiration sightings.
- Sightings can be performed via the API, and the UI, even including the import of STIX sighting documents
- Many use-cases for scoring indicators based on users sighting

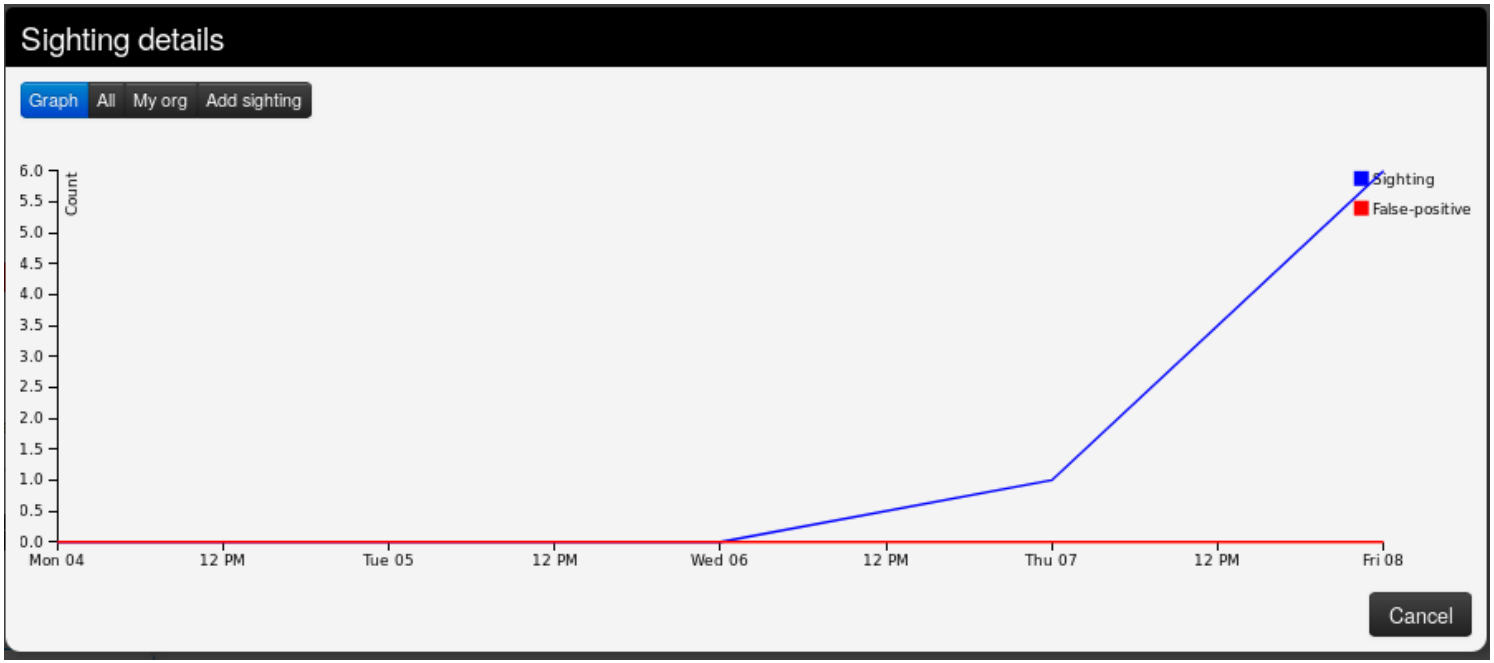
Recording the hits

```
grep -o -f ips.txt /var/log/apache2/access.log |
    sort -u > matches.txt
```

Using sightings in MISP

```
AR=‘cat matches.txt | jq -csR ’split("\n")’‘
TS=‘date +%s‘
QUERY="""
␣␣␣␣{\"values\":␣$AR␣,\"type\":\"0\",\"source\":\"
    webserver\",\"timestamp\":\"$TS\"}
"""
echo $QUERY > query.json
curl \
 -d @query.json \
 -H "Authorization:␣YOUR_API_KEY" \
 -H "Accept:␣application/json" \
 -H "Content-type:␣application/json" \
 -X POST http://127.0.0.1:8080/sightings/add
```

Using sightings in MISP





Checking network logs

Packet captures

- Use local filtering capacities of tools → BPF filter
- Do not under estimate textual representation overhead, i.e.json, xml

Create BPF rule from MISP search query

```
for ip in `cat ips.txt `; do
        echo -n "host␣$ip␣or␣" ;
done | sed 's/or $//g' > filter.txt

find /pcaps/daily -type f | sort | parallel 'zcat {} |
   tcpdump -tt -n -r - "`cat␣filter.txt`"'
```

Daily search for hashes on your server

```
find /home/ -type f | parallel 'md5sum {}; sha1sum {};
    sha256sum {}' >hashed_files.txt
```

```
curl -d '{"returnFormat":"text","type":["md5", "sha1",
     "sha256"],"timestamp":"1d","enforceWarninglist":"1
    ","to_ids":"1"}' -H 'Authorization: YOUR_API_KEY' -
    H 'Accept: application/json' -H 'Content-type:
    application/json' -X POST https://127.0.0.1:8080/
    attributes/restSearch > known_hashes.txt
```

```
grep -f known_hashes.txt hashed_files.txt | sort -u >
    matches.txt
```

Execute it in a daily cron job

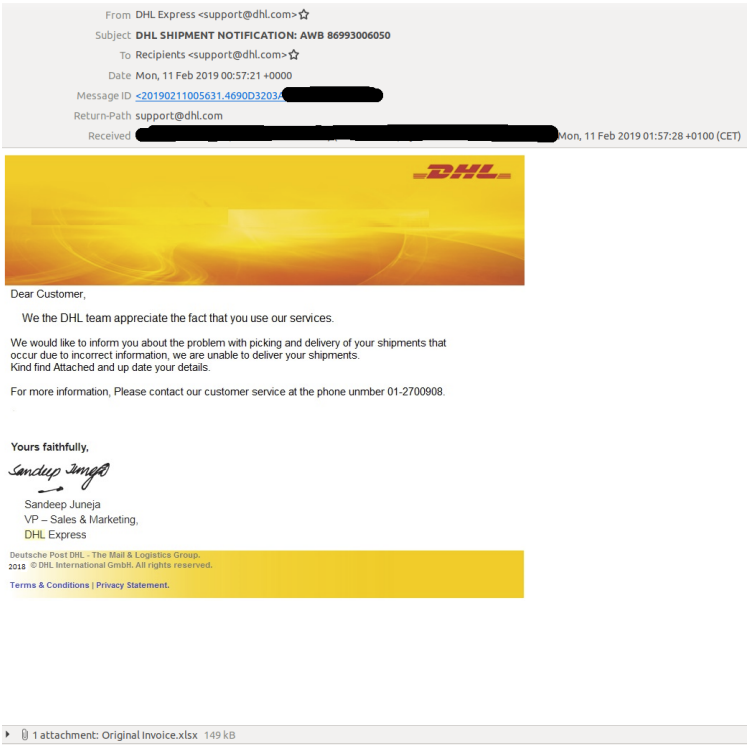
Using sightings in MISP

Pitfalls

- Sighting values $\rightarrow$ scalability issues due to duplicates
  - Do sighting on attribute x
  - x is included in z events
  - z sightings are added
- Sighting uuids of attributes
  - Only a given attribute in an event is sighted
  - Example: attribute 8.8.8.8 is sighted only in your event
  - Sighting third party events is complex
  - Need publisher right to republish the sightings

Simple information sharing example

Malicious emails



From DHL Express <support@dhl.com>
Subject **DHL SHIPMENT NOTIFICATION: AWB 86993006050**
To Recipients <support@dhl.com>
Date Mon, 11 Feb 2019 00:57:21 +0000
Message ID <20190211005631.4690D3203[redacted]
Return-Path support@dhl.com
Received [redacted] Mon, 11 Feb 2019 01:57:28 +0100 (CET)

Dear Customer,

We the DHL team appreciate the fact that you use our services.

We would like to inform you about the problem with picking and delivery of your shipments that occur due to incorrect information, we are unable to deliver your shipments.
Kind find Attached and up date your details.

For more information, Please contact our customer service at the phone unmber 01-2700908.

Yours faithfully,

Sandeep Juneja
VP – Sales & Marketing,
DHL Express

Deutsche Post DHL - The Mail & Logistics Group.
2018 © DHL International GmbH, All rights reserved.
Terms & Conditions | Privacy Statement.

1 attachment: Original Invoice.xlsx 149 kB

Forward the email and use mail2misp

Conclusion

- MISP is just a tool. What matters is your sharing practices. The tool should be as transparent as possible to support you
- Consume data with standard Unix tools is demonstrated
- The demonstration of all features of MISP is a two days training[4]
- Enable users to customize MISP to meet their community's use-cases
- MISP project combines open source software, open standards, best practices and communities to make information sharing a reality

[4] https://www.circl.lu/services/misp-training-materials/