In Curation We Trust: Generating Contextual & Actionable Threat Intelligence

Michel Coene
Bart Parys

2022-07-01
TLP:White

Michel Coene — **Bart Parys**

Content

What we will dive into today

Slides 4-7	Problem Statement & Examples
Slides 8 - 14	The Curation Procedure… & More
Slides 15-17	Lessons Learned & Roadmap
Q&A	

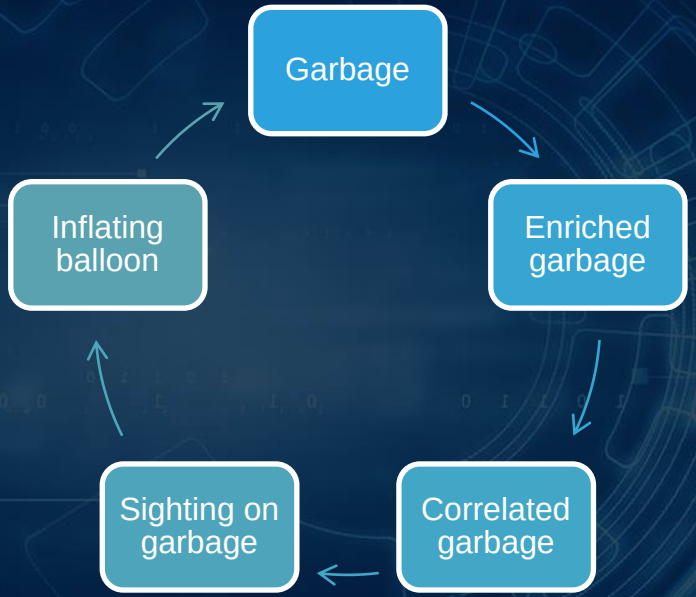
The problem

What everyone has ~~done~~ seen before, at least once

Indicator: 8.8.8.8

IDS: Enabled

SOC:



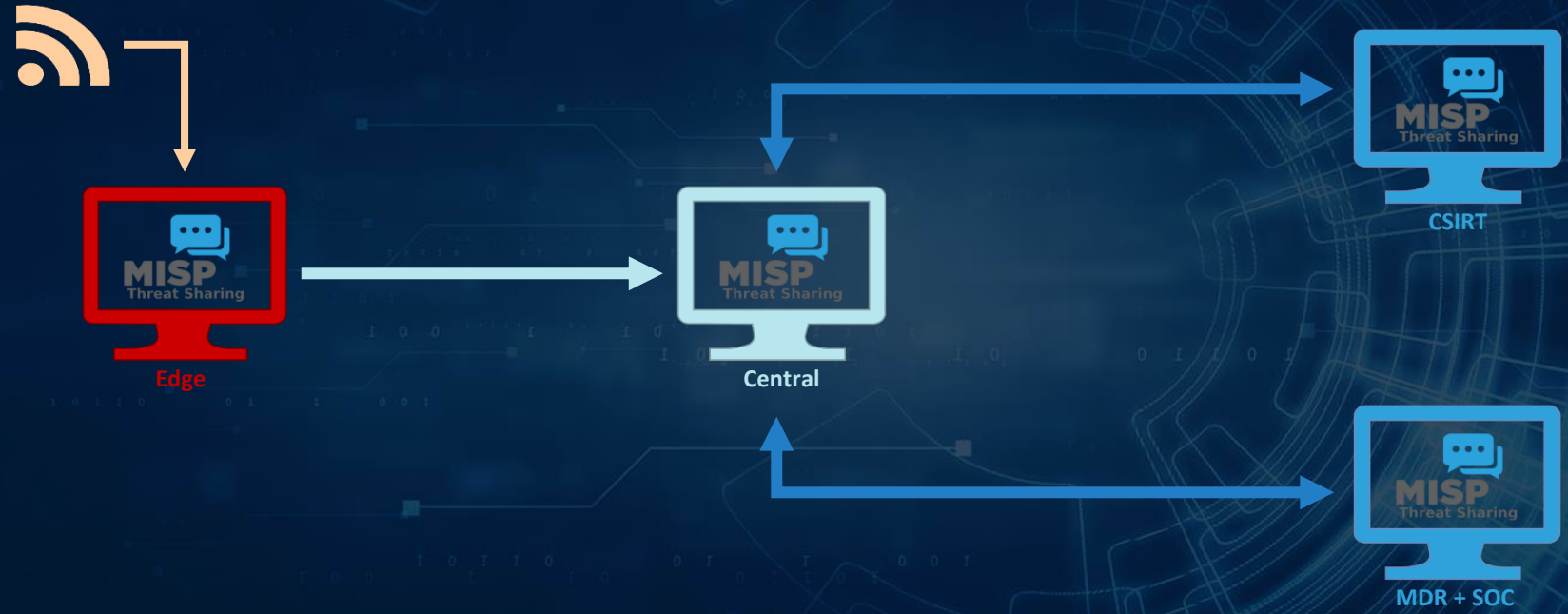
Threat Intelligence

Consuming & producing

NVISO MDR Managed Detect & Respond	NVISO CSIRT Incident Response Team	NVISO CTI Cyber Threat Intelligence
• **TI consumer** • **TI producer** • SOC monitoring of our clients • Anonymized threat data is fed back in the MDR MISP instance for future correlation • Threat hunting	• **TI consumer** • **TI producer** • Incident response cases • Malware analysis	• **TI consumer** • **TI producer** • Threat intelligence integrations • Threat Intel Feeds • Tailored threat briefings • Threat landscape reports • Adversary emulation plans • Vulnerability intelligence

The architecture

The first step in an attempt to solve the problem



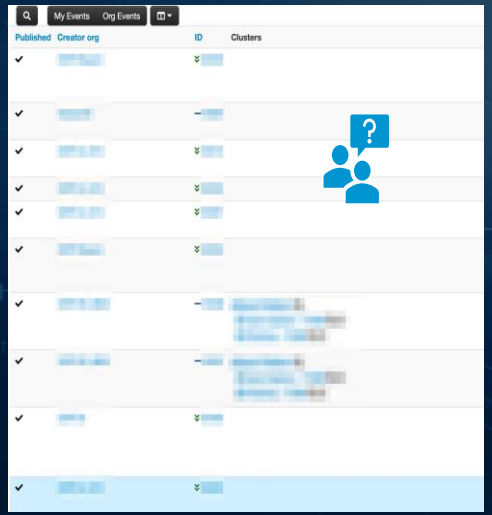
Problem exemplars

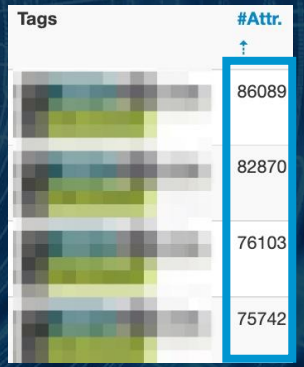
Recipe for sub-optimal detection

Potential false positives

Lack of contextualisation or inconsistencies

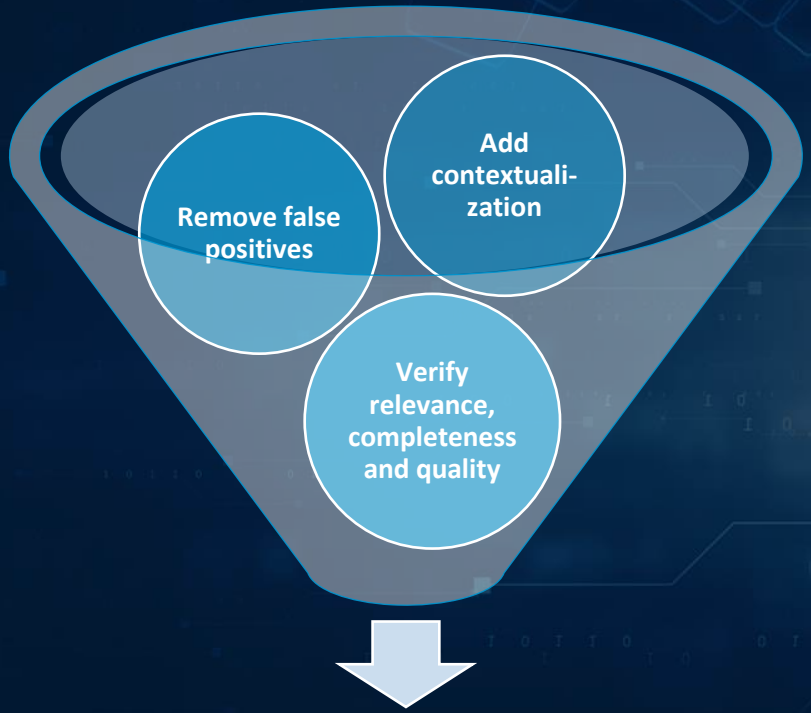
Actionable?





The curation procedure

In practical terms, what do we do?



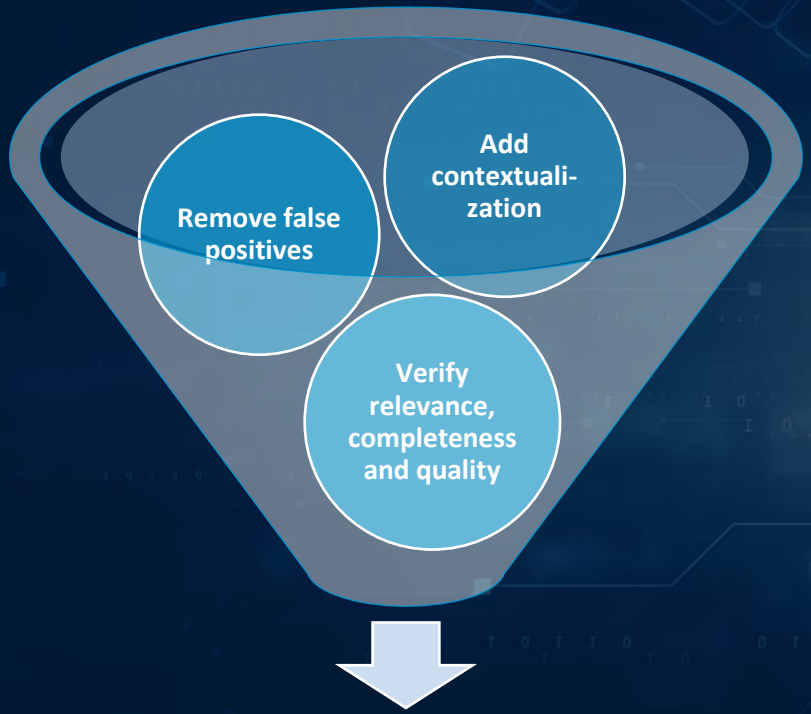
Remove **false positives**
- MISP warning lists
- Custom warning lists
- Analyst judgement

Add **contextualization**
- Mandatory TLP tags
- Intel source
- Relations, comments and objects
- Target info, threat actor, sectors, MITRE ATT&CK tags, ...

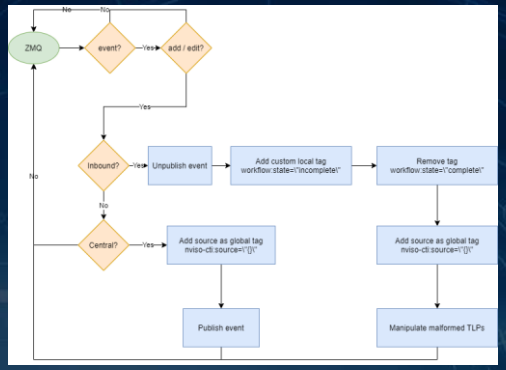
Verify **relevance**, completeness and **quality**
- "Useful"
- Sanity check
- If this alerts, is there enough context?

An attempt to solve the problem

Curation procedure



Threat Intelligence




```
class nviso_curate:
  '''
    Class to curate NVISO events
    Curation depends on the server we're being executed

    :param logger: logger object
    :param custom_tag: Add this tag to events
    :param remove_tag: Remove this tag from events

  '''

  def __init__(self, logger, custom_tag, remove_tag):
    self.misp = ExpandedPyMISP(misp_url, misp_key, misp_verifycert,
    self.custom_tag = custom_tag
    self.logger = logger
    self.remove_tag = remove_tag
    self.source_tag = "nviso-cti:source=\"{}\""
```

```
# Remove tag "complete"
if self.remove_tag:
  result_tag = self.misp.untag(uuid, self.remove_tag)
  self.logger.debug("Untag event {} - {} {}".format(uuid, misp_event.info, self.remove_tag))
```

ID ↑	Name	Version	Description	Category
145	List of known Limelight CDN IPs	1	List of known Limelight CDN IPs	False positive
144	NITRO URL False Positives	39	False Positive URLs as observed by the NVISO Intelligence and Threat Response Operations.	False positive
143	NITRO Hashes False Positives	41	False Positive Hashes as observed by the NVISO Intelligence and Threat Response Operations.	False positive
142	NITRO Domain False Positives	39	False Positive Domains as observed by the NVISO Intelligence and Threat Response Operations.	False positive
140	NITRO Registry False Positive	39	False Positive Registry Keys as observed by the NVISO Intelligence and Threat Response Operations.	False positive
139	List of known Azure IPs	2	Azure data centers, edge nodes etc.	False positive
136	NVISO False Positive List	6	Indicators that are found to be false positives by the CTI team	False positive

```
elif tag.name.strip() == "TLP:White" or tag.name.strip() == "TLP: White"  or tag.name.strip() == "TLP:WHI
  self.misp.untag(uuid, tag.id)
  result_tag = self.misp.tag(uuid, "tlp:white")
```

```
# Add source of this event
if include_source:
  servername = self.get_server(int(server))
  source_tag = self.source_tag.format(servername)
```

Automated Curation

Automation is key!

Edge

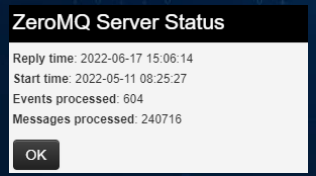
ZMQ → Unpublish → Workflow → Source → TLP

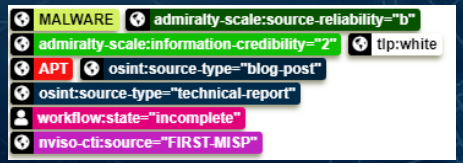
Subscribe

“Incomplete”

Who?

Sanitize





Manual Curation

Manual still comes into play

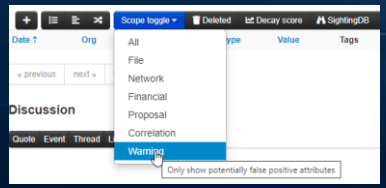
Edge

Central

Workflow → False Positives → Quality & Completeness → Context → Relevant

Unpublished

Warning lists

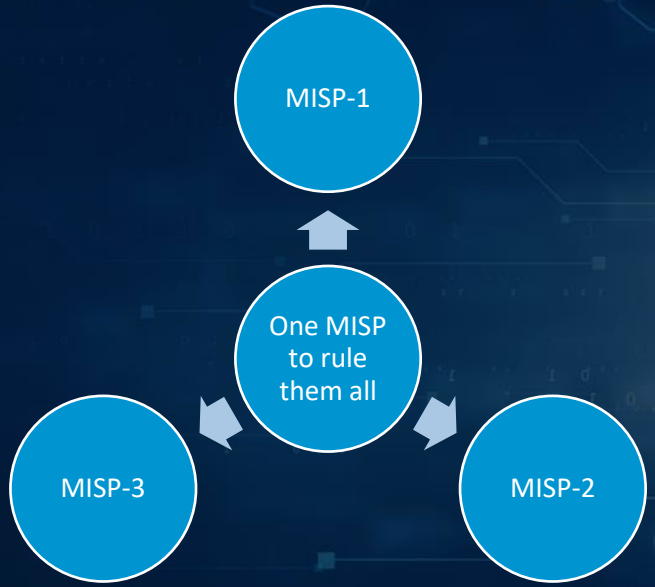


Target info

Usefulness

No need repeating

"Offender list" and blocklist



Synchronise offender lists between MISP servers

Organisation Blocklists

« previous | next »

Id	Organisation name	UUID	Created ↑	Comment
[redacted]	[redacted]	[redacted]	2022-05-09 18:33:59	Very low quality, extremely large events with very high amount of FP's
[redacted]	[redacted]	[redacted]	2022-04-05 09:54:08	Generic sandbox output, non-filtered, nonsensical
[redacted]	[redacted]	[redacted]	2021-10-13 08:31:00	Low quality intel, no attributes marked or markable as IDS true
[redacted]	[redacted]	[redacted]	2021-09-22 08:48:29	
[redacted]	[redacted]	[redacted]	2021-07-29 14:39:57	Lots of low quality cuckoo analysis events.
[redacted]	[redacted]	[redacted]	2021-05-12 10:42:53	Low quality
[redacted]	[redacted]	[redacted]	2021-04-02 11:57:14	Scanner / Source IPs > 1000 hits, low value

```
for uuid in diff_source_blocklist_uuids:
    payload = json.dumps({"returnFormat": "json", "uuid": uuid})
    search_results = requests.request("POST", search_dest_url, headers=dest_headers, data=payload, verify=False)
    if len(search_results.json()['response']) != 0:
        try:
            delete_response = requests.request("DELETE", delete_dest_url + uuid, headers=dest_headers, verify=False)
            delete_response.raise_for_status()
            deleted_count += 1
        except requests.exceptions.HTTPError as err:
            print(err)
```

Statistics

What have we seen so far?

3.219 events

2.041.477 attributes

16 blocklisted organisations

Other scripts to complement our curation

False positives

- Remove *or* tag NSRL matches
- Hashlookup CIRCL

Relevant indicators

- Inactivate indicators after a grace period
- Basic decaying of indicators

Scrape web sources

- Collect OSINT
- MISP reports

Bulk delete events

- Events with non-relevant information
- You need a backup plan

Lessons Learned

Key components to make this work

Tooling

- **Customisation is key**
- **Automate as much as possible**
- **Extend what is available**

- ZMQ and Python
- Platform features
- Taxonomies (workflow)
- Galaxies and clusters

Documentation

- **Server architecture**
- **MISP synchronization data flows**

- There is a limit to what you can automate
- Operating procedures for analysts
- Multiple analysts – but same procedure

Communication

- **Involve stakeholders at the appropriate time**
- **Let TI consumers signal the quality of TI**

- Rinse and repeat

Future State

Roadmap

Integrate MISP workflows (New MISP feat.)

- Shareable models

Contribute back to community

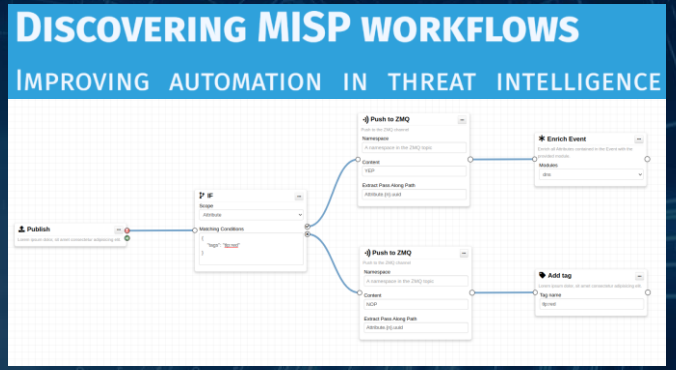
- Event proposals
- Sightings
- *Resources*
- *Mature process*

"Announcements"

- Create bot (Slack, Teams, …) that alert when new events are in

Open source scripts

- Generalise code for wider use
- https://github.com/NVISOsecurity/nviso-cti



THANK YOU!

Q&A

Michel Coene
Bart Parys

threatintel@nviso.eu

2022-07-01
TLP:White