

# Turning data into actionable intelligence

advanced features in MISP supporting your analysts and tools

CIRCL / Team MISP Project



CIISI-EU





**circl.lu**  
Computer Incident  
Response Center  
LUXEMBOURG

- The Computer Incident Response Center Luxembourg (CIRCL) is a government-driven initiative designed to provide a systematic response facility to computer security threats and incidents. CIRCL is the CERT for the private sector, communes and non-governmental entities in Luxembourg and is operated by securitymadein.lu g.i.e.

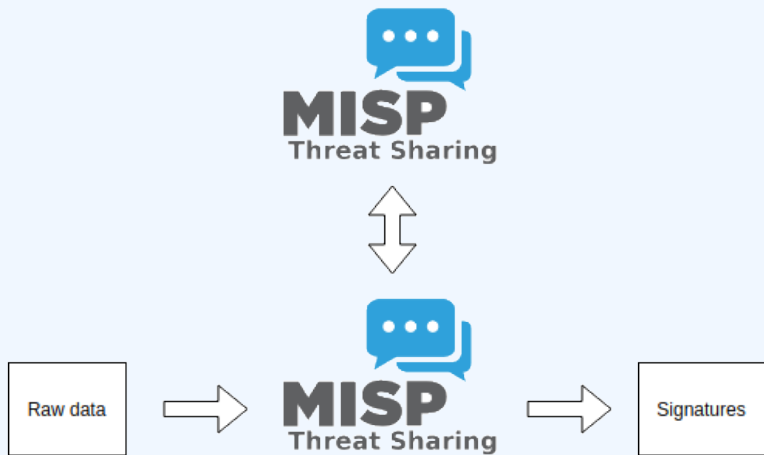
- CIRCL is mandated by the Ministry of Economy and acting as the Luxembourg National CERT for private sector.
- CIRCL leads the development of the Open Source MISP threat intelligence platform which is used by many military or intelligence communities, private companies, financial sector, National CERTs and LEAs globally.
- **CIRCL runs multiple large MISP communities performing active daily threat-intelligence sharing.**

# THE AIM OF THIS PRESENTATION

- To give some insight into what sort of an evolution of our various communities' have gone through as observed over the past 8 years
- Show the importance of **strong contextualisation...**
- ...and how that can be leveraged when trying to make our data **actionable**

- 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, likelihood and occurrences.
  - ▶ **Fraud analysts** willing to share financial indicators to detect financial frauds.

- **Extract information** during the analysis process
- Store and **correlate** these datapoints
- **Share** the data with partners
- Focus on technical indicators: IP, domain, hostname, hashes, filename, pattern in file/memory/traffic
- Generate protective signatures out of the data: snort, suricata, OpenIOC



# WHY WAS IT SO SIMPLISTIC?

- This was both a reflection of our maturity as a community
  - ▶ Capabilities for **extracting** information
  - ▶ Capabilities for **utilising** the information
  - ▶ Lack of **willingness** to share context
  - ▶ Lack of **co-operation** between teams doing technical analysis/monitoring and threat-intel
- The more growth we saw in maturity, the more we tried to match it with our data-model, often against pushback



- There were separate factors that made our data-sets less and less useful for detection/defense in general
  - ▶ **Growth of our communities**
  - ▶ Distinguish between information of interest and raw data
  - ▶ **False-positive** management
  - ▶ TTPs and aggregate information may be prevalent compared to raw data (risk assessment)
  - ▶ **Increased data volumes** leads to be able to prioritise

# OUR INITIAL SOLUTION

- Allow users to **tag any information** created in MISP
- We wanted to be **lax with what we accept** in terms of data, but be **strict on what we fed to our tools**, with strong filter options
- We had some ideas on how to potentially move forward...

- Try to capture different aspects of contextualisation into **normalised values** (threat level, source reliability, etc)
  - ▶ Didn't scale with needs other than our own
  - ▶ Incorporating new types of contextualisation would mean **the modification of the software**
  - ▶ Getting communities with **established naming conventions** to use anything but their go-to vocabularies was a pipe-dream
  - ▶ Heated arguments over numeric conversions

- We tried an alternate approach instead: Free tagging
  - ▶ Result was spectacularly painful, at least 7 different ways to spell tlp:amber
  - ▶ No canonisation for common terms lead to tagging ultimately becoming a highly flawed tool for filtering within a sharing community

TLP AMBER

TLP:AMBER

Threat tlp:Amber

tlp-amber

tlp::amber

tlp:amber

# HOW WE ENDED UP TACKLING THE ISSUE MORE SUCCESSFULLY

- We ended up with a mixed approach, currently implemented by the MISP-taxonomy system
  - ▶ Taxonomies are **vocabularies** of known tags
  - ▶ Tags would be in a **triple tag format**  
namespace:predicate="value"
  - ▶ Create your own taxonomies, recipients should be able to use data you tag with them without knowing it at the first place
  - ▶ Avoid any coding, stick to **JSON**
- Massive success, approaching 100 taxonomies
- Organisations can solve their own issues without having to rely on us

<input type="checkbox"/> Tag	Events	Attributes	Tags
<input type="checkbox"/> workflow:state="complete"	11	0	workflow:state="complete" ↩
<input type="checkbox"/> workflow:state="draft"	0	0	workflow:state="draft" ↩
<input type="checkbox"/> workflow:state="incomplete"	55	10	workflow:state="incomplete" ↩
<input type="checkbox"/> workflow:state="ongoing"	0	0	workflow:state="ongoing" ↩

# WE WERE STILL MISSING SOMETHING...

- Taxonomy tags often **non self-explanatory**
- Example: universal understanding of tlp:green vs APT 28
- For the latter, a single string was ill-suited
- So we needed something new in addition to taxonomies - **Galaxies**
  - ▶ Community driven **knowledge-base libraries used as tags**
  - ▶ Including descriptions, links, synonyms, meta information, etc.
  - ▶ Goal was to keep it **simple and make it reusable**
  - ▶ Internally it works the exact same way as taxonomies (stick to **JSON**)

🔗 Ransomware galaxy	
Galaxy ID	373
Name	Ransomware
Namespace	misp
Uuid	3f44af2e-1480-4b6b-9aa8-f9bb21341078
Description	Ransomware galaxy based on...
Version	4
Value ↓	<a href="#">Synonyms</a>
.CryptoHasYou.	
777	Sevleg
7ev3n	7ev3n-HONE\$T

# BROADENING THE SCOPE OF WHAT SORT OF CONTEXT WE ARE INTERESTED IN

- **Who** can receive our data? **What** can they do with it?
- **Data accuracy, source reliability**
- **Why** is this data relevant to us?
- **Who** do we think is behind it, **what tools** were used?
- What sort of **motivations** are we dealing with? Who are the **targets**?
- How can we **block/detect/remediate** the attack?
- What sort of **impact** are we dealing with?

# PARALLEL TO THE CONTEXTUALISATION EFFORTS: FALSE POSITIVE HANDLING

- Low quality / false positive prone information being shared
- Lead to **alert-fatigue**
- Exclude organisation xy out of the community?
- False positives are often obvious - **can be encoded**
- **Warninglist system**<sup>1</sup> aims to do that
- Lists of well-known indicators which are often false-positives like RFC1918 networks, ...

## LIST OF KNOWN IPV4 PUBLIC DNS RESOLVERS

Id	89
Name	List of known IPv4 public DNS resolvers
Description	Event contains one or more public IPv4 DNS resolvers as attribute with an IDS flag set
Version	20181114
Type	string
Accepted attribute types	ip-src, ip-dst, domain/ip
Enabled	Yes (disable)
Values	
	1.0.0.1
	1.1.1.1
	1.1.1.4

### Warning: Potential false positives

List of known IPv4 public DNS resolvers

Top 1000 website from Alexa

List of known google domains

<sup>1</sup><https://github.com/MISP/misp-warninglists>



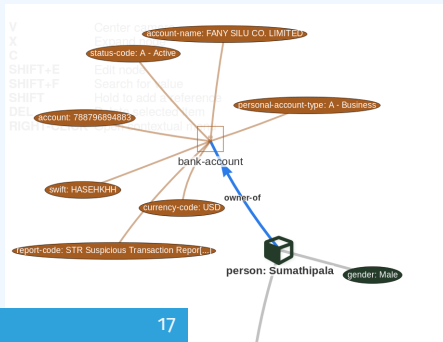
- Atomic attributes were a great starting point, but lacking in many aspects
- **MISP objects**<sup>2</sup> system
  - ▶ Simple **templating** approach
  - ▶ Use templating to build more complex structures
  - ▶ Decouple it from the core, allow users to **define their own** structures
  - ▶ MISP should understand the data without knowing the templates
  - ▶ Massive caveat: **Building blocks have to be MISP attribute types**
  - ▶ Allow **relationships** to be built between objects

---

<sup>2</sup><https://github.com/MISP/misp-objects>

# SUPPORTING SPECIFIC DATAMODEL

Date	Org	Category	Type	Value	Tags	Galaxies	Comment	Correlate	Related Events
Name: bank-account									
References: 0									
<input type="checkbox"/>	2018-09-28	Other	status-code:	A - Active	is	Add		<input type="checkbox"/>	
<input type="checkbox"/>	2018-09-28	Other	report-code:	STR Suspicious Transaction Report	is	Add		<input type="checkbox"/>	
<input type="checkbox"/>	2018-09-28	Other	personal-account-type:	A - Business	is	Add		<input type="checkbox"/>	
<input type="checkbox"/>	2018-09-28	Financial fraud	swift:	HASEH00H	is	Add		<input checked="" type="checkbox"/>	3849 11320 11584
<input type="checkbox"/>	2018-09-28	Financial fraud	account:	788796894883	is	Add		<input checked="" type="checkbox"/>	
<input type="checkbox"/>	2018-09-28	Other	account-name:	FANY SILU CO. LIMITED	is	Add		<input checked="" type="checkbox"/>	
<input type="checkbox"/>	2018-09-28	Other	currency-code:	USD	is	Add		<input type="checkbox"/>	



- Data ingested by MISP was in a sense frozen in time
- We had a creation data, but lacked a way to use the output of our detection
- Lead to the introduction of the **Sighting system**
- The community could sight indicators and convey the time of sighting
- Potentially powerful tool for IoC lifecycle management, clumsy query implementation default

# SUPPORTING SPECIFIC DATAMODEL

Events					
<input checked="" type="checkbox"/>	No				
<input checked="" type="checkbox"/>	No	Inherit			
<input checked="" type="checkbox"/>	No	Inherit	👍 👎 🔧		

Sightings

CIRCL: 2 (2017-03-19 16:17:59)

(2/0/0)

(0/0/0)

Tags	<b>+</b>
Date	2016-02-24
Threat Level	High
Analysis	Initial
Distribution	Connected communities
	freetext test
Sighting Details	<b>No</b>
MISP: 2	4 (2) - restricted to own organisation only.
CIRCL: 2	
	<b>- Discussion</b>

- Most obvious goal: Improve the way we query data
  - ▶ Unified all export APIs
  - ▶ Incorporate all contextualisation options into **API filters**
  - ▶ Allow for an **on-demand** way of **excluding potential false positives**
  - ▶ Allow users to easily **build their own** export modules feed their various tools

## EXAMPLE QUERY

```
/attributes/restSearch
```

```
{  
  "returnFormat": "netfilter",  
  "enforceWarninglist": 1,  
  "tags": {  
    "NOT": [  
      "tlp:white",  
      "type:OSINT"  
    ],  
    "OR": [  
      "misp-galaxy:threat-actor=\"Sofacy\"",  
      "misp-galaxy:sector=\"Chemical\""  
    ],  
  }  
}
```

- Make decisions on whom to share data with based on context
  - ▶ MISP by default decides based on the information creator's decision who data gets shared with
  - ▶ Community hosts should be able to **act as a safety net** for sharing
    - **Push filters** - what can I push?
    - **Pull filters** - what am I interested in?
    - **Local tags** allow for information flow control

# THE EMERGENCE OF ATT&CK AND SIMILAR GALAXIES

- Standardising on high-level **TTPs** was a solution to a long list of issues
- Adoption was rapid, tools producing ATT&CK data, familiar interface for users
- A much better take on kill-chain phases in general
- Feeds into our **filtering** and **situational awareness** needs extremely well
- Gave rise to other, ATT&CK-like systems tackling other concerns
  - ▶ **attck4fraud** <sup>3</sup> by Francesco Bigarella from ING
  - ▶ **Election guidelines** <sup>4</sup> by NIS Cooperation Group

---

<sup>3</sup>[https://www.misp-project.org/galaxy.html#\\_attck4fraud](https://www.misp-project.org/galaxy.html#_attck4fraud)

<sup>4</sup>[https://www.misp-project.org/galaxy.html#\\_election\\_guidelines](https://www.misp-project.org/galaxy.html#_election_guidelines)

[https://www.misp-project.org/galaxy.html#\\_election\\_guidelines](https://www.misp-project.org/galaxy.html#_election_guidelines)



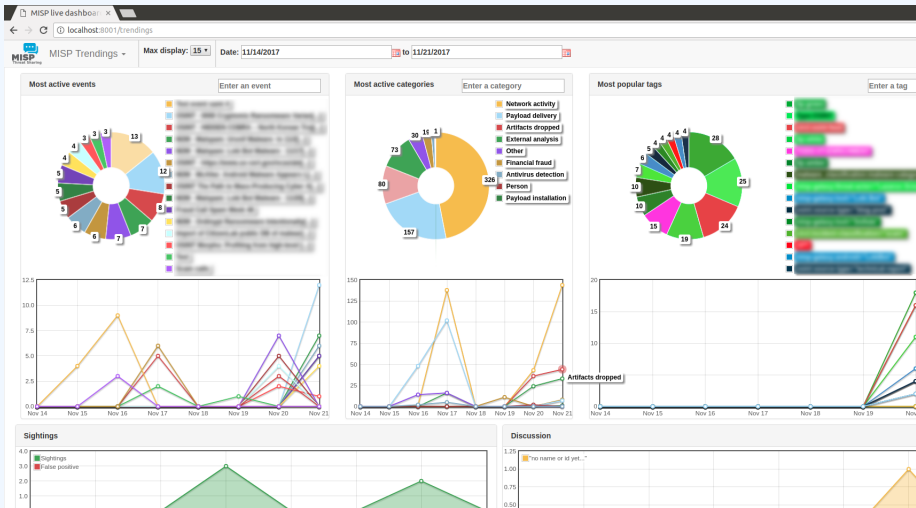
## EXAMPLE QUERY TO GENERATE ATT&CK HEATMAPS

```
/events/restSearch
{
  "returnFormat": "attack",
  "tags": [
    "misp-galaxy:sector=\"Chemical\""
  ],
  "timestamp": "365d"
}
```

# A SAMPLE RESULT FOR THE ABOVE QUERY

Initial access	Execution	Persistence	Privilege escalation	Defense evasion	Credential access	Discovery	Lateral movement	Collection	Exfiltration	Command and control
Spearphishing Attachment	Scripting	Screensaver	File System Permissions Weakness	Process Hollowing	Secured Memory	Password Policy Discovery	AppleScript	Data from Information Repositories	Exfiltration Over Alternative Protocol	Standard Application Layer Protocol
Spearphishing via Service	Command-Line Interface	Login Item	AppCert DLLs	Code Signing	Input Capture	System Network Configuration Discovery	Distributed Component Object Model	Data from Removable Media	Exfiltration Over Command and Control Channel	Communication Through Removable Media
Trusted Relationship	User Execution	Trap	Application Shimming	Rookit	Bash History	Process Discovery	Pass the Hash	Man in the Browser	Data Compressed	Custom Command and Control Protocol
Replication Through Removable Media	Regsvcs/Regasm	System Firmware	Scheduled Task	NTFS File Attributes	Exploitation for Credential Access	Network Share Discovery	Exploitation of Remote Services	Data Staged	Automated Exfiltration	Multi-Stage Channels
Exploit Public Facing Application	Trusted Developer Utilities	Registry Run Keys / Start Folder	Startup Items	Exploitation for Defense Evasion	Private Keys	Peripheral Device Discovery	Remote Desktop Protocol	Screen Capture	Scheduled Transfer	Remote Access Tools
Spearphishing Link	Windows Management Instrumentation	LC_LOAD_DYLIB Addition	New Service	Network Share Connection Removal	Brute Force	Account Discovery	Pass the Ticket	Email Collection	Data Encrypted	Uncommonly Used Port
Valid Accounts	Service Execution	LSASS Driver	Sudo Caching	Process Doppelganging	Password Filter DLL	System Information Discovery	Windows Remote Management	Clipboard Data	Exfiltration Over Other Network Medium	Multi-layer Encryption
Supply Chain Compromise	CMSTP	Rc.common	Process Injection	Disabling Security Tools	Two-Factor Authentication Interception	System Network Connections Discovery	Windows Admin Shares	Video Capture	Exfiltration Over Physical Medium	Domain Fronting
Drive-by Compromise	Control Panel Items	Authentication Package	Bypass User Account Control	Timestamp	LLMNR/NBT-NS Poisoning	Network Service Scanning	Remote Services	Audio Capture	Data Transfer Size Limits	Data Obfuscation
Hardware Additions	Dynamic Data Exchange	Component Firmware	Extra Window Memory Injection	Modify Registry	Credentials in Files	File and Directory Discovery	Taint Shared Content	Data from Network Shared Drive		Connection Proxy
	Source	Windows Management Instrumentation Event Subscription	Setuid and Setgid	Indicator Removal from Tools	Forced Authentication	Security Software Discovery	Application Deployment Software	Data from Local System		Commonly Used Port
	Space after Filename	Change Default File	Launch Daemon	Hidden Window	Keychain	System Service Discovery	Third-party Software	Automated Collection		Data Encoding

# MONITOR TRENDS OUTSIDE OF MISP (EXAMPLE: DASHBOARD)



- We were still missing a way to use all of these systems in combination to decay indicators
- Move the decision making **from complex filter options to complex decay models**
- Decay models would take into account various **taxonomies, sightings**, the **type** of each indicator **Sightings** and **Creation date**
- The first iteration of what we have in MISP now took:
  - ▶ 2 years of research
  - ▶ 3 published research papers
  - ▶ A lot of prototyping

$$\text{score}(\text{Attribute}) = \text{base\_score}(\text{Attribute}, \text{Model}) \bullet \text{decay}(\text{Model}, \text{time})$$

Where,

- $\text{score} \in [0, 100]$
- $\text{base\_score} \in [0, 100]$
- $\text{decay}$  is a function defined by model's parameters controlling decay speed
- $\text{Attribute}$  Contains *Attribute's* values and metadata (*Taxonomies, Galaxies, ...*)
- $\text{Model}$  Contains the *Model's* configuration

# IMPLEMENTATION IN MISP: Event/view

The screenshot displays the MISP interface for viewing an event. At the top, there are navigation tabs: "Photos", "Galaxy", "Event graph", "Correlation graph", "ATTACK matrix", "Attributes", and "Discussion". Below these is a blue button labeled "45: Decay...". A search bar labeled "Galaxies" is present. The main content area shows a table of events with the following columns: Date, Org, Category, Type, Value, Tags, Galaxies, Comment, Correlate, Related Events, Feed hits, IDS Distribution, Sightings, Activity, Score, and Actions. The "Decay score" toggle button is highlighted in blue. The table contains five rows of event data, each with a score and a "Model 5" button.

Date	Org	Category	Type	Value	Tags	Galaxies	Comment	Correlate	Related Events	Feed hits	IDS Distribution	Sightings	Activity	Score	Actions
2019-09-12		Network activity	ip-src	5.5.5.5							Inherit	(0/0)		NIDS Simple Decaying ... 65.26 Model 5 79.88	
2019-08-13		Network activity	ip-src	8.8.8.8	admiralty-scale:source-reliability="A" x retention:expired x				1 2 2 2 Show S1.1 S1.2 11 more...		Inherit	(5/0)		NIDS Simple Decaying ... 54.6 Model 5 52.69	
2019-08-13		Network activity	ip-src	9.9.9.9	admiralty-scale:source-reliability="C" x misp:confidence-level="completely-confident" x Ipnumber				1 3 1 9 Show S1.1 28 more...		Inherit	(4/1)		NIDS Simple Decaying ... 37.43 Model 5 0	
2019-08-13		Network activity	ip-src	7.7.7.7	admiralty-scale:information-credibility="4" x retention:20 x				41		Inherit	(3/0)		NIDS Simple Decaying ... 37.41 Model 5 0	
2019-07-18		Network activity	ip-src	6.6.6.6					41		Inherit	(0/0)		NIDS Simple Decaying ... 23.31 Model 5 0	

## ■ Decay score toggle button

- ▶ Shows Score for each Models associated to the Attribute type

## IMPLEMENTATION IN MISP: API RESULT









/attributes/restSearch

```
"Attribute": [  
  {  
    "category": "Network activity",  
    "type": "ip-src",  
    "to_ids": true,  
    "timestamp": "1565703507",  
    [...]  
    "value": "8.8.8.8",  
    "decay_score": [  
      {  
        "score": 54.475223849544456,  
        "decayed": false,  
        "DecayingModel": {  
          "id": "85",  
          "name": "NIDS Simple Decaying Model"  
        },  
      }  
    ]  
  }  
]
```

# IMPLEMENTATION IN MISP: INDEX

## Decaying Models

« previous next »

All Models	My Models	Shared Models	Default Models	ID	Organization	Usable to everyone	Name	Description	Parameters { }	Formula	# Assigned Types	Version	Enabled	Actions
				29	1	✓	Phishing model	Simple model to rapidly decay phishing website.	<pre>{   "lifetime": 3,   "decay_speed": 2.3,   "threshold": 30,   "default_base_score": 80,   "base_score_config": {     "estimative-language": 0.5,     "phishing": 0.5   } }</pre>	Polynomial	9	1	✓	   
				85	1	✗	NIDS Simple Decaying Model MISP	Simple decaying model for Network Intrusion Detection System (NIDS).	<pre>{   "lifetime": 120,   "decay_speed": 2,   "threshold": 30,   "default_base_score": 80,   "base_score_config": {     "estimative-language": 0.25,     "priority-level": 0.25,     "retention": 0.25,     "targeted-threat-index": 0.125,     "false-positive": 0.125   } }</pre>	Polynomial	13	1	✓	   

Page 1 of 1, showing 2 records out of 2 total, starting on record 1, ending on 2

« previous next »

View, update, add, create, delete, enable, export, import



# IMPLEMENTATION IN MISP: FINE TUNING TOOL

**Decaying Of Indicator Fine Tuning Tool**

Attributes Table:

Attribute Type	Category	Model ID
aba-rtn	Financial fraud	
authenthash	Payload delivery	
bank-account-iv	Financial fraud	
bc	Financial fraud	
bin	Financial fraud	
brn	Network activity	10 11
bc	Financial fraud	11
cc-number	Financial fraud	
cdhash	Payload delivery	
community-id	Network activity	
domain	Network activity	
domainip	Network activity	10 94
email-attachment	Payload delivery	
email-dst	Network activity	11
email-enc	Payload delivery	
headers	Payload delivery	
headers/authenthash	Payload delivery	
headers/iphuzzy	Payload delivery	
headers/iphash	Payload delivery	
headers/ipv4	Payload delivery	13
headers/iphash	Payload delivery	13
headers/ipv4	Payload delivery	13

Control Panel:

- Polynomial (Model Type)
- Lifetime: 3 days
- Decay speed: 2.3
- Cutoff threshold: 30
- Expire after (lifetime): 1 days and 7 hours
- Score halved after (Half-life): 0 day and 6 hours
- Buttons: Adjust base score, Simulate this model
- Model: Phishing model (Simple model to rapidly decay)
- Parameters table:

Parameters													
ID	Model Name	Org ID	Description	Formula	Lifetime	Decay speed	Threshold	Default basescore	Basescore config	Settings	# Types	Enabled	Action
29	Phishing model	1	Simple model to rapidly decay phishing website	Polynomial	3	2.3	30	80	estimate-language phishing	0.5	0.5	✓	Test model

Create, modify, visualise, perform mapping

# IMPLEMENTATION IN MISP: base\_score TOOL

Search Taxonomy  x

Default basescore 80

**3 not having numerical value**

Taxonomies	Weight
<b>admiralty-scale</b>	
source-reliability	31
information-credibility	30
<b>priority-level</b>	
priority-level	53
<b>retention</b>	
retention	0
<b>estimative-language</b>	
likelihood-probability	0
confidence-in-analytic-judgment	0
<b>misp</b>	
confidence-level	0
threat-level	0
automation-level	0
<b>phishing</b>	
state	0
psychological-acceptability	0
<b>Excluded</b>	

Placeholder for "Organisation source confidence"

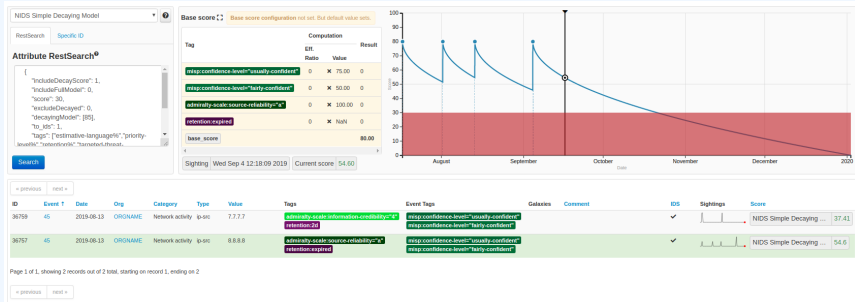
### Example

Attribute	Tags	Base score
Tag your attribute	+	
Attribute 1	admiralty-scale:information-credibility="3"	0.0
Attribute 2	priority-level:baseline-minor admiralty-scale:source-reliability="d" admiralty-scale:information-credibility="2"	38.2
Attribute 3	priority-level:severe admiralty-scale:information-credibility="2"	84.6

### Computation steps

Tag	Computation		Result
	Eff. Ratio	Value	
priority-level:baseline-minor	0.46	* 25.00	11.62
admiralty-scale:source-reliability="d"	0.27	* 25.00	6.80

# IMPLEMENTATION IN MISP: SIMULATION TOOL



Simulate *Attributes* with different *Models*

```
/attributes/restSearch
{
  "includeDecayScore": 1,
  "includeFullModel": 0,
  "excludeDecayed": 0,
  "decayingModel": [85],
  "modelOverrides": {
    "threshold": 30
  }
  "score": 30,
}
```

- Massive rise in **user capabilities**
- Growing need for truly **actionable threat intel**
- Lessons learned:
  - ▶ **Context is king** - Enables better decision making
  - ▶ **Intelligence and situational awareness** are natural by-products of context
  - ▶ Don't lock users into your **workflows**, build tools that enable theirs

## ■ Contact us

- ▶ [https://twitter.com/mokaddem\\_sami](https://twitter.com/mokaddem_sami)
- ▶ <https://twitter.com/iglowska>

## ■ Contact CIRCL

- ▶ [info@circl.lu](mailto:info@circl.lu)
- ▶ [https://twitter.com/circl\\_lu](https://twitter.com/circl_lu)
- ▶ <https://www.circl.lu/>

## ■ Contact MISPProject

- ▶ <https://github.com/MISP>
- ▶ <https://gitter.im/MISP/MISP>
- ▶ <https://twitter.com/MISPProject>