Building Your Own Workflows in MISP

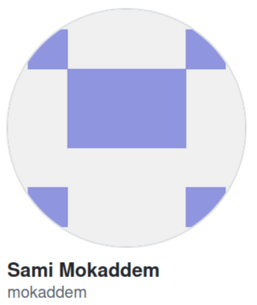
Overview of the feature

Sami Mokaddem & Alexandre Dulaunoy

MISP Project
https://www.misp-project.org/

2023-07-07

$ whoami



Sami Mokaddem

mokaddem

Belgian roads are terrible
Belgians
The Belgian Federal Government sucks
Belgians
The French invented fries
YOU BETTER WATCH YOUR MOUTH

Automation in MISP: What already exists?

MISP API / PyMISP

- Needs CRON Jobs in place
- Not realtime

∅ **PubSub channels**

- After the actions happen: No feedback to MISP
- Tougher to put in place & to share

→ No way to **prevent** behavior
→ Difficult to setup **hooks** to execute callbacks

What type of use-cases are we trying to support?

- **Prevent** default MISP behaviors to happen
  - Prevent **publication of events** not passing sanity checks
  - Prevent **querying** thrid-party **services** with sensitive information
  - ...
- **Hook** specific actions to run callbacks
  - **Automatically run** enrichment services
  - Modify data on-the-fly: False positives, enable CTI-Pipeline
  - Send notifications in a chat rooms
  - ...

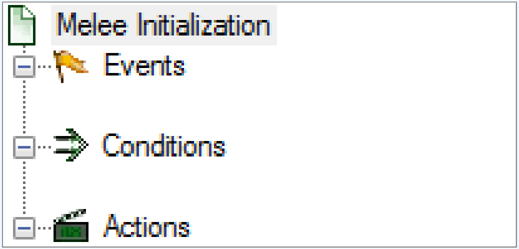
Simple automation in MISP made easy

- How?
  - **Drag & Drop** editor
  - Prevent actions **before they happen**
  - Flexible **Plug & Play** system
  - **Share** workflows, **debug** and **replay**

Workflow - Fundamentals

Objective: Start with the foundation to understand the basics

How does it work



Melee Initialization

Events

Conditions

Actions

1. An **event** happens in MISP
2. Check if all **conditions** are satisfied
3. Execute all **actions**

What kind of events?

Events

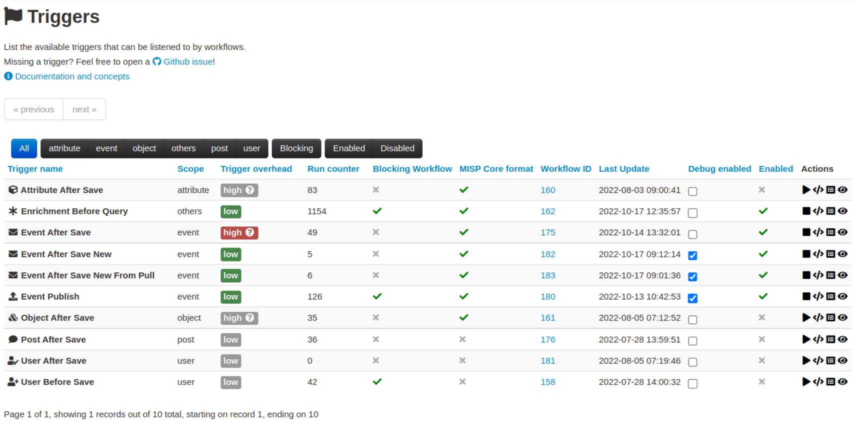
- New MISP Event
- Attribute has been saved
- New discussion post
- New user created
- Query against third-party services
- ...

❓ Supported events in MISP are called **Triggers**

❓ A **Trigger** is associated with **1-and-only-1 Workflow**

Triggers currently available

Currently 11 triggers can be hooked. 3 being **■ Blocking**.



Triggers

List the available triggers that can be listened to by workflows.
Missing a trigger? Feel free to open a Github issue!
Documentation and concepts

Trigger name	Scope	Trigger overhead	Run counter	Blocking Workflow	MISP Core format	Workflow ID	Last Update	Debug enabled	Enabled	Actions
Attribute After Save	attribute	high ?	83	×	✓	160	2022-08-03 09:00:41	☐	×	▶ </> ▤ ◉
Enrichment Before Query	others	low	1154	✓	✓	162	2022-10-17 12:35:57	☐	✓	■ </> ▤ ◉
Event After Save	event	high ?	49	×	✓	175	2022-10-14 13:32:01	☐	✓	■ </> ▤ ◉
Event After Save New	event	low	5	×	✓	182	2022-10-17 09:12:14	☑	✓	■ </> ▤ ◉
Event After Save New From Pull	event	low	6	×	✓	183	2022-10-17 09:01:36	☑	✓	■ </> ▤ ◉
Event Publish	event	low	126	✓	✓	180	2022-10-13 10:42:53	☑	✓	■ </> ▤ ◉
Object After Save	object	high ?	35	×	✓	161	2022-08-05 07:12:52	☐	×	▶ </> ▤ ◉
Post After Save	post	low	36	×	×	176	2022-07-28 13:59:51	☐	×	▶ </> ▤ ◉
User After Save	user	low	0	×	×	181	2022-08-05 07:19:46	☐	×	▶ </> ▤ ◉
User Before Save	user	low	42	✓	×	158	2022-07-28 14:00:32	☐	×	▶ </> ▤ ◉

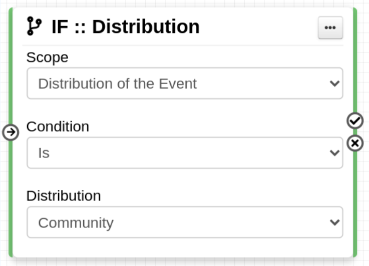
Page 1 of 1, showing 1 records out of 10 total, starting on record 1, ending on 10

What kind of conditions?

⇛ Conditions

- A MISP Event is tagged with `tlp:red`
- The distribution of an Attribute is a sharing group
- The creator organisation is `circl.lu`
- Or any other **generic** conditions

? These are also called **Logic modules**



IF :: Distribution

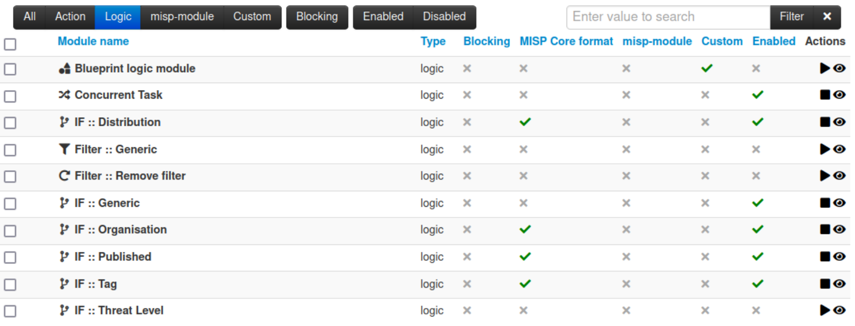
Scope: Distribution of the Event

Condition: Is

Distribution: Community

Workflow - Logic modules

- **logic** modules: Allow to redirect the execution flow.
  - IF conditions
  - Delay execution



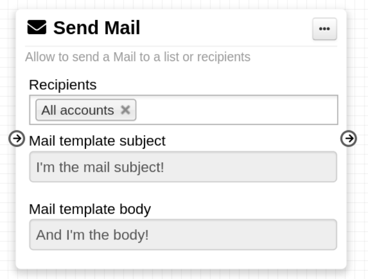
□	Module name	Type	Blocking	MISP Core format	misp-module	Custom	Enabled	Actions
□	Blueprint logic module	logic	✕	✕	✕	✓	✕	▶ 👁
□	Concurrent Task	logic	✕	✕	✕	✕	✓	■ 👁
□	IF :: Distribution	logic	✕	✓	✕	✕	✓	■ 👁
□	Filter :: Generic	logic	✕	✕	✕	✕	✕	▶ 👁
□	Filter :: Remove filter	logic	✕	✕	✕	✕	✕	▶ 👁
□	IF :: Generic	logic	✕	✕	✕	✕	✓	■ 👁
□	IF :: Organisation	logic	✕	✓	✕	✕	✓	■ 👁
□	IF :: Published	logic	✕	✓	✕	✕	✓	■ 👁
□	IF :: Tag	logic	✕	✓	✕	✕	✓	■ 👁
□	IF :: Threat Level	logic	✕	✕	✕	✕	✕	▶ 👁

What kind of actions?

Actions

- Send an email notification
- Perform enrichments
- Send a chat message on MS Teams
- Attach a local tag
- ...

These are also called **Action modules**



Send Mail

Allow to send a Mail to a list or recipients

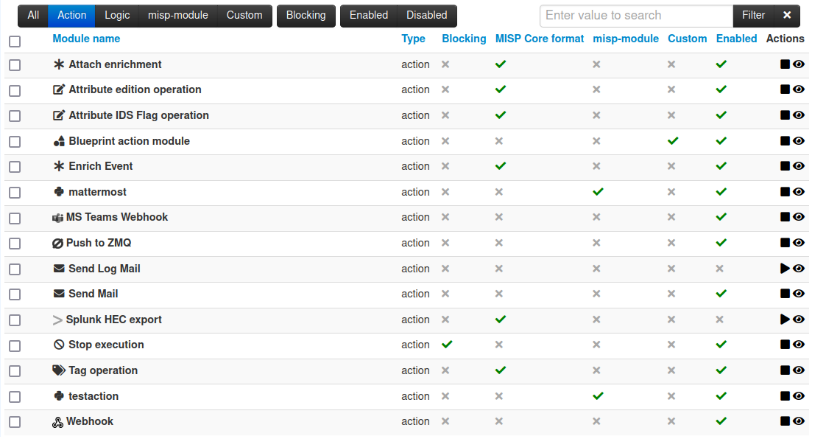
Recipients: All accounts ✕

Mail template subject: I'm the mail subject!

Mail template body: And I'm the body!

Workflow - Action modules

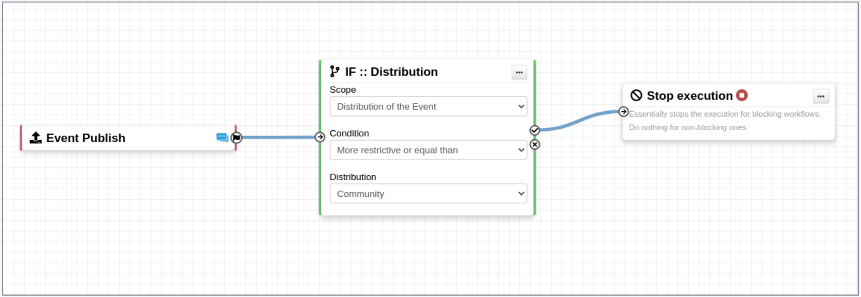
- **action** modules: Allow to executes operations
  - Tag operations
  - Send notifications
  - Webhooks & Custom scripts



□	Module name	Type	Blocking	MISP Core format	misp-module	Custom	Enabled	Actions
□	Attach enrichment	action	✗	✓	✗	✗	✓	■ 👁
□	Attribute edition operation	action	✗	✓	✗	✗	✓	■ 👁
□	Attribute IDS Flag operation	action	✗	✓	✗	✗	✓	■ 👁
□	Blueprint action module	action	✗	✗	✗	✓	✓	■ 👁
□	Enrich Event	action	✗	✓	✗	✗	✓	■ 👁
□	mattermost	action	✗	✗	✓	✗	✓	■ 👁
□	MS Teams Webhook	action	✗	✗	✗	✗	✓	■ 👁
□	Push to ZMQ	action	✗	✗	✗	✗	✓	■ 👁
□	Send Log Mail	action	✗	✗	✗	✗	✗	▶ 👁
□	Send Mail	action	✗	✗	✗	✗	✓	■ 👁
□	Splunk HEC export	action	✗	✓	✗	✗	✗	▶ 👁
□	Stop execution	action	✓	✗	✗	✗	✓	■ 👁
□	Tag operation	action	✗	✓	✗	✗	✓	■ 👁
□	testaction	action	✗	✗	✓	✗	✓	■ 👁
□	Webhook	action	✗	✗	✗	✗	✓	■ 👁

What is a MISP Workflow?

- Sequence of all nodes to be executed in a specific order



Event Publish

IF :: Distribution

Scope: Distribution of the Event

Condition: More restrictive or equal than

Distribution: Community

Stop execution

Essentially stops the execution for blocking workflows.
Do nothing for non-blocking ones

Workflow execution for Event publish

An Event is about to be published

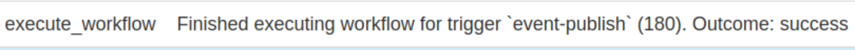
- The workflow for the `event-publish` trigger starts

Conditions are evaluated

- They might change the path taken during the execution

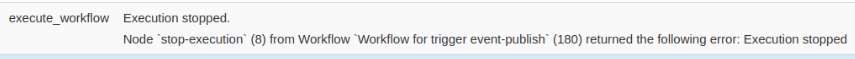
Actions are executed

- **success**: Continue the publishing action



```
execute_workflow    Finished executing workflow for trigger `event-publish` (180). Outcome: success
```

- **failure** | `blocked`: Stop publishing and log the reason



```
execute_workflow    Execution stopped.
                    Node `stop-execution` (8) from Workflow `Workflow for trigger event-publish` (180) returned the following error: Execution stopped
```

Sources of Workflow modules (0)

Currently 36 built-in modules.

- **Trigger** module (11): built-in **only**
    - Get in touch if you want more
- **Logic** module (10): built-in & **custom**
- **Action** module (15): built-in & **custom**

Sources of Workflow modules (1)

- Built-in **default** modules
  - Part of the MISP codebase
  - Get in touch if you want us to increase the selection (or merge PR!)

Tag operation v0.2
Add or remove tags on Event or Attributes.
Send Mail v0.1
Allow to send a Mail to a list or recipients
Webhook v0.2
Allow to perform custom callbacks to the provided URL
Stop execution v0.1
Essentially stops the execution for blocking workflows. Do nothing for non-blocking ones
Push to ZMQ v0.1
Push to the ZMQ channel
MS Teams Webhook v0.2
Perform callbacks to the MS Teams webhook provided by the "Incoming Webhook" connector
Enrich Event v0.1
Enrich all Attributes contained in the Event with the provided module.
ORVAL
Quadrupel
TRAPPIST
Trappistes
BIRRA
BEER
CERVEZA
TRAPPIST
Tripel

Sources of Workflow modules (2)

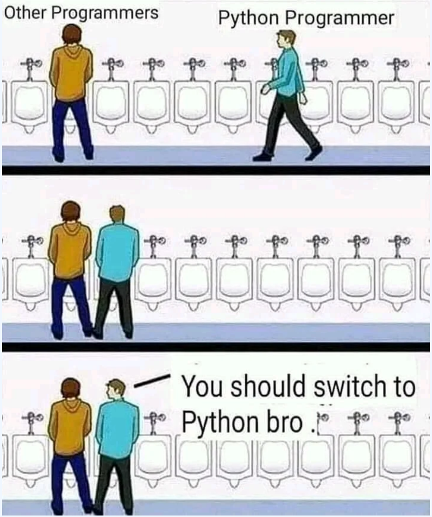
User-defined **custom** modules

- Written in PHP
- Extend existing modules
- MISP code reuse

Sources of Workflow modules (3)

Modules from the misp-module **enrichment service**

- Written in Python
- Can use any python libraries
- Plug & Play



Getting started with workflows (5)

Let’s build a workflow!

🛠️

Creating a workflow with the editor

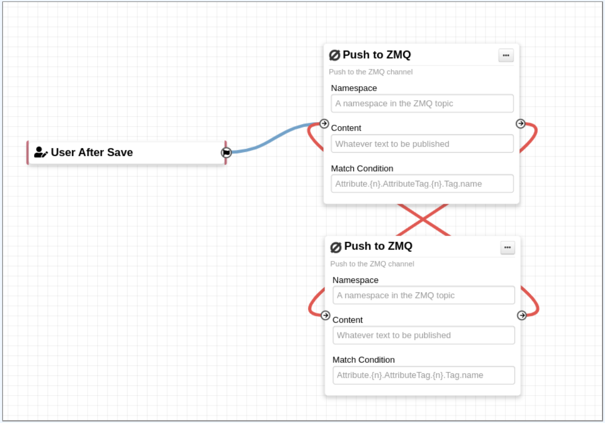
1. Prevent event publication if **tlp:red** tag
2. Send a mail to `admin@admin.test` about potential data leak
3. Otherwise, send a notification on **Mattermost**, **MS Teams**, **Telegram**, ...

Considerations when working with workflows

Objective: Overview of some common pitfalls

Working with the editor - Operations not allowed

Execution loop are not authorized



User After Save

Push to ZMQ

Push to the ZMQ channel

Namespace: A namespace in the ZMQ topic

Content: Whatever text to be published

Match Condition: Attribute.{n}.AttributeTag.{n}.Tag.name

Push to ZMQ

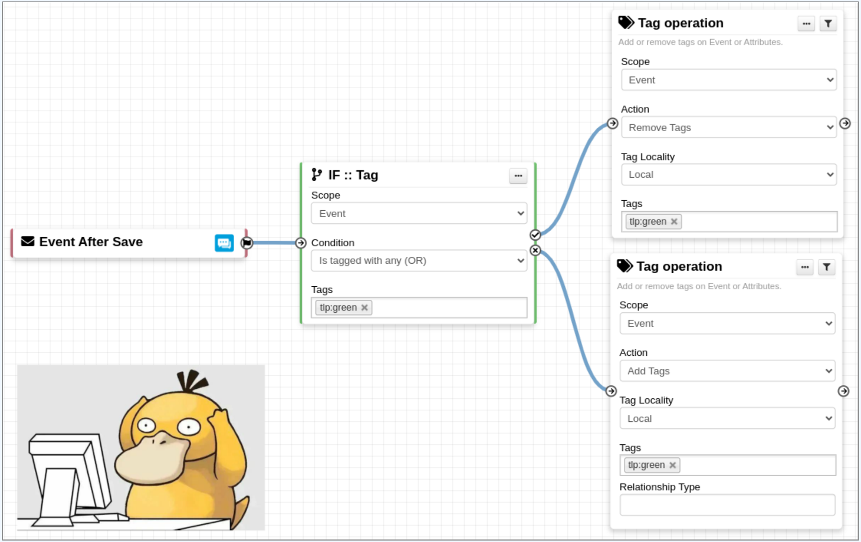
Push to the ZMQ channel

Namespace: A namespace in the ZMQ topic

Content: Whatever text to be published

Match Condition: Attribute.{n}.AttributeTag.{n}.Tag.name

Recursive workflows



Event After Save

IF :: Tag

Scope: Event

Condition: Is tagged with any (OR)

Tags: tlp:green ×

Tag operation

Add or remove tags on Event or Attributes.

Scope: Event

Action: Remove Tags

Tag Locality: Local

Tags: tlp:green ×

Tag operation

Add or remove tags on Event or Attributes.

Scope: Event

Action: Add Tags

Tag Locality: Local

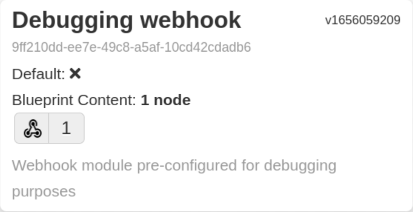
Tags: tlp:green ×

Relationship Type:

⚠Recursion: If an action re-run the workflow

Workflow blueprints

1. Blueprints allow to **re-use parts** of a workflow in another one
2. Blueprints can be saved, exported and **shared**



Debugging webhook v1656059209

9ff210dd-ee7e-49c8-a5af-10cd42cdadb6

Default: ✗

Blueprint Content: **1 node**

Webhook module pre-configured for debugging purposes

Blueprints sources:

1. Created or imported by users
2. From the `MISP/misp-workflow-blueprints` repository[1]

[1] https://github.com/MISP/misp-workflow-blueprints

Workflow blueprints

Currently, 4 blueprints available:

- Attach the `tlp:clear` tag on elements having the `tlp:white` tag
- Block actions if any attributes have the `PAP:RED` or `tlp:red` tag
- Disable `to_ids` flag for existing hash in *hashlookup*
- Set tag based on *BGP Ranking* maliciousness level

Future works

- More 🖥 modules
- More ⇛ modules
- More ⚑ triggers
- More documentation
- Recursion prevention system
- On-the-fly data override?

Final words

- Designed to **quickly** and **cheaply** integrate MISP in CTI pipelines
- **Beta** Feature unlikely to change. But still..
- Waiting for feedback!
  - New triggers?
  - New modules?
  - What's acheivable