

# Extending MISP with Python modules

MISP - Malware Information Sharing Platform & Threat Sharing



**CIRCL**

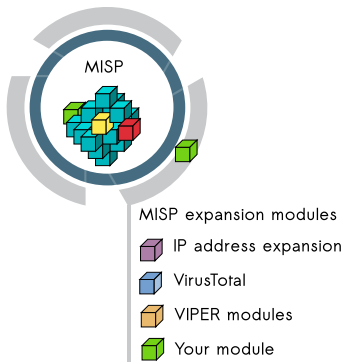
Computer Incident  
Response Center  
Luxembourg

Alexandre Dulaunoy -  
*TLP:WHITE*

March 24, 2016

# MISP modules - extending MISP with Python scripts

---



- Extending MISP with expansion modules with zero customization in MISP.
- A simple ReST API between the modules and MISP allowing auto-discovery of new modules with their features.
- Benefit from existing Python modules in Viper or any other tools.
- MISP modules functionality introduced in MISP 2.4.28.

## MISP modules - installation

---

- MISP modules can be run on the same system or on a remote server.
- Python 3 is required to run MISP modules.
  - `git clone git@github.com:MISP/misp-modules.git`
  - `cd misp-modules`
  - `pip3 install -r REQUIREMENTS`
  - `cd bin`
  - `python3 misp-modules.py`

## MISP modules - Simple REST API mechanism

---

- <http://127.0.0.1:6666/modules> - introspection interface to get all modules available
  - returns a JSON with a description of each module
- <http://127.0.0.1:6666/query> - interface to query a specific module
  - to send a JSON to query the module
- MISP autodiscovers the available modules and the MISP site administrator can enable modules as they wish.
- If a configuration is required for a module, MISP adds automatically the option in the server settings.

# Finding available MISP modules

---

- `curl -s http://127.0.0.1:6666/modules`

```
1      {
2      "type": "expansion",
3      "name": "dns",
4      "meta": {
5          "module-type": [
6              "expansion",
7              "hover"
8          ],
9          "description": "Simple DNS expansion
10             service to resolve IP address from
11             MISP attributes",
12          "author": "Alexandre Dulaunoy",
13          "version": "0.1"
14      },
15      "mispattributes": {
16          "output": [
17              "ip-src",
18              "ip-dst"
19          ],
20          "input": [
21              "hostname",
22              "domain"
23          ]
24      }
25  }
```

## Querying a module

---

- `curl -s http://127.0.0.1:6666/query -H "Content-Type: application/json" -data @body.json -X POST`

body.json

1

```
{"module": "dns", "hostname": "www.circl.lu"}
```

- and the response of the dns module:

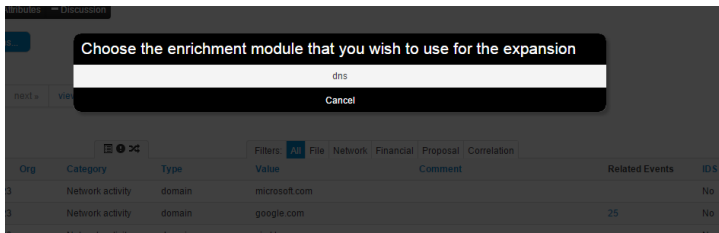
1

```
{"results": [{"values": ["149.13.33.14"],  
"types": ["ip-src", "ip-dst"]}]}
```

2

# MISP modules - How it's integrated in the UI?

Filters: All	File	Network	Financial	Proposal	Correlation				
Value	Comment	Related Events	IDS	Distribution	Actions				
microsoft.com			No	Inherit	* 🗑️				
google.com		25	No	Inherit	* 🗑️				
circl.lu			No	Inherit	* 🗑️				



## Enrichment Results

Below you can see the attributes that are to be created. Make sure that the categories and the types are correct, often several options will be offered based on an inconclusive automatic resolution.

Value	Category	Type	IDS <input type="checkbox"/>	Comment	Actions
23.100.122.175	Network activity	ip-src	<input type="checkbox"/>	Imported via the freetext import. ✕	

→

# MISP modules - configuration in the UI

## Server settings

Overview MISP settings (18) GnuPG settings (3) Proxy settings (5) Security settings (2) Misc settings (1) Plugin settings (22) Diagnostics Workers

### Enrichment

Priority	Setting	Value	Description
Critical	Plugin.Enrichment_services_enable	true	Enable/disable the enrichment services
Recommended	Plugin.Enrichment_services_url	http://127.0.0.1	The url used to access the enrichment services
Recommended	Plugin.Enrichment_services_port	6666	The port used to access the enrichment services
Recommended	Plugin.Enrichment_cve_enabled	false	Enable or disable the cve enrichment
Recommended	Plugin.Enrichment_dns_enabled	true	Enable or disable the dns enrichment
Recommended	Plugin.Enrichment_sourcecache_enabled	false	Enable or disable the sourcecache enrichment
Recommended	Plugin.Enrichment_sourcecache_archivepath		Set this required module setting
Recommended	Plugin.Enrichment_passivetotal_enabled	true	Enable or disable the passivetotal enrichment
Recommended	Plugin.Enrichment_passivetotal_username	alexandre.dulaunoy@circl.lu	Set this required module setting
Recommended	Plugin.Enrichment_passivetotal_password		Set this required module setting



# Creating your module

---

```
import json
import dns.resolver

misperrors = {'error': 'Error'}
mispattributes = {'input': ['hostname', 'domain'], 'output': ['ip-src', 'ip-dst']}
moduleinfo = {'version': '0.1', 'author': 'Alexandre-Dulaunoy',
              'description': 'Simple_DNS_expansion_service_to_resolve_IP_address_from_MISP_attributes', 'module-type': ['expansion', 'hover']}

def handler(q=False):
    if q is False:
        return False
    request = json.loads(q)
    if request.get('hostname'):
        toquery = request['hostname']
    elif request.get('domain'):
        toquery = request['domain']
    else:
        return False
    r = dns.resolver.Resolver()
    r.timeout = 2
    r.lifetime = 2
    r.nameservers = ['8.8.8.8']
    try:
        answer = r.query(toquery, 'A')
    except dns.resolver.NXDOMAIN:
        misperrors['error'] = "NXDOMAIN"
        return misperrors
    except dns.exception.Timeout:
        misperrors['error'] = "Timeout"
        return misperrors
    except:
        misperrors['error'] = "DNS_resolving_error"
        return misperrors
    r = {'results': [{'types': mispattributes['output'], 'values': [str(answer[0])]}]}
    return r

def introspection():
    return mispattributes

def version():
    return moduleinfo
```

## Testing your module

---

- Copy your module `dns.py` in `modules/expansion/`
- Restart the server `misp-modules.py`

```
[adulau:~/git/misp-modules/bin]$ python3 misp-modules.py
2016-03-20 19:25:43,748 - misp-modules - INFO - MISP modules passivetotal imported
2016-03-20 19:25:43,787 - misp-modules - INFO - MISP modules sourcecache imported
2016-03-20 19:25:43,789 - misp-modules - INFO - MISP modules cve imported
2016-03-20 19:25:43,790 - misp-modules - INFO - MISP modules dns imported
2016-03-20 19:25:43,797 - misp-modules - INFO - MISP modules server started on TCP port 6666
```

- Check if your module is present in the introspection
- `curl -s http://127.0.0.1:6666/modules`
- If yes, test it directly with MISP or via `curl`

## Q&A

---



- <https://github.com/MISP/misp-modules>
- <https://github.com/MISP/>
- We welcome new modules and pull requests.
- MISP modules can be designed as standalone application.