Extending MISP with Python modules

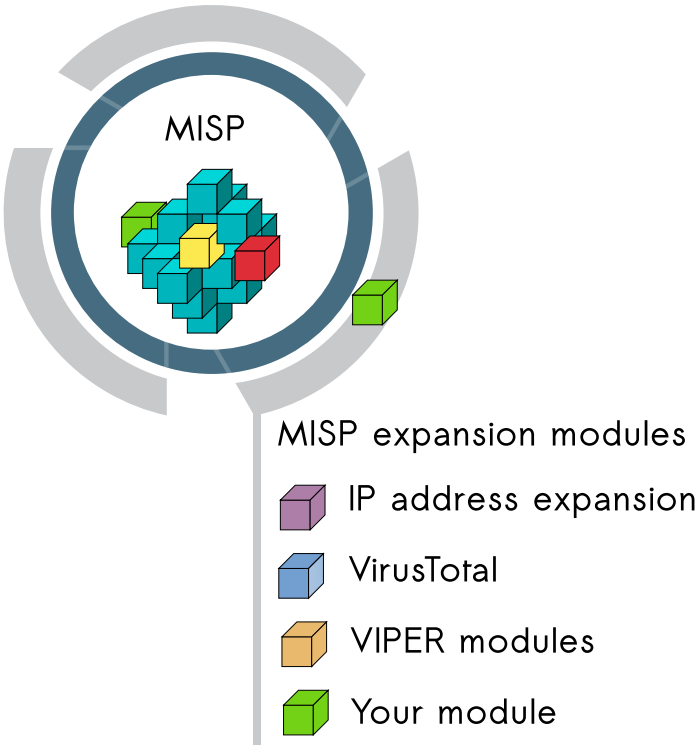
MISP - Malware Information Sharing Platform & Threat Sharing

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 functionnality 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


```
{
"type": "expansion",
"name": "dns",
"meta": {
  "module-type": [
    "expansion",
    "hover"
  ],
  "description": "Simple DNS expansion
     service to resolve IP address from
     MISP attributes",
  "author": "Alexandre Dulaunoy",
  "version": "0.1"
},
"mispattributes": {
  "output": [
    "ip-src",
    "ip-dst"
  ],
  "input": [
    "hostname",
    "domain"
  ]
}
```

Querying a module

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

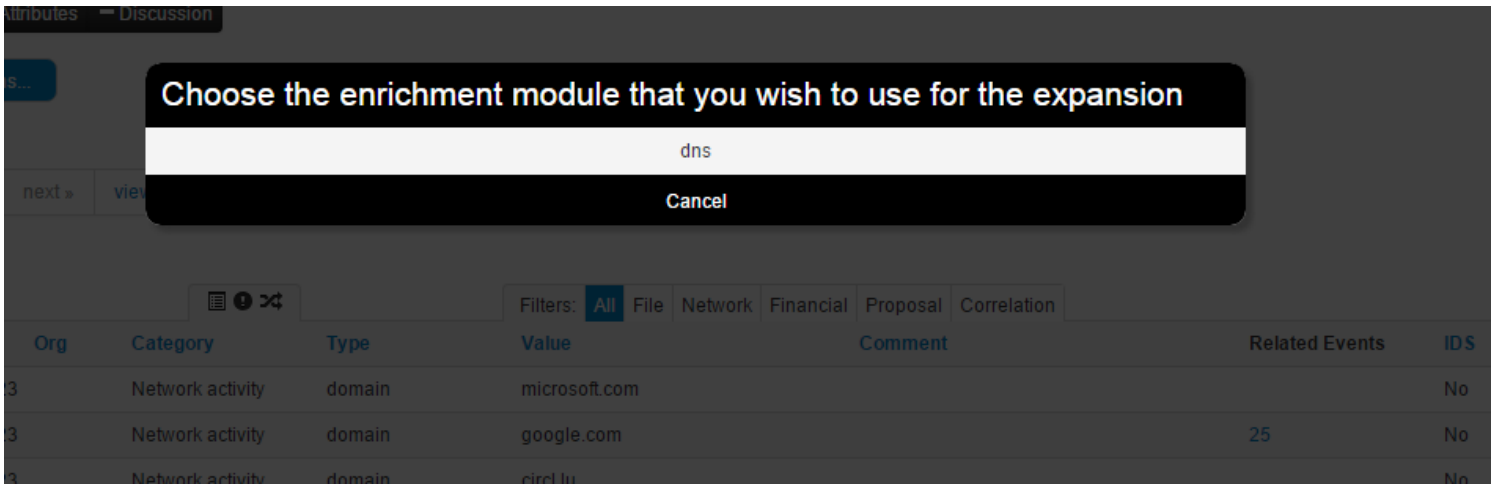
body.json

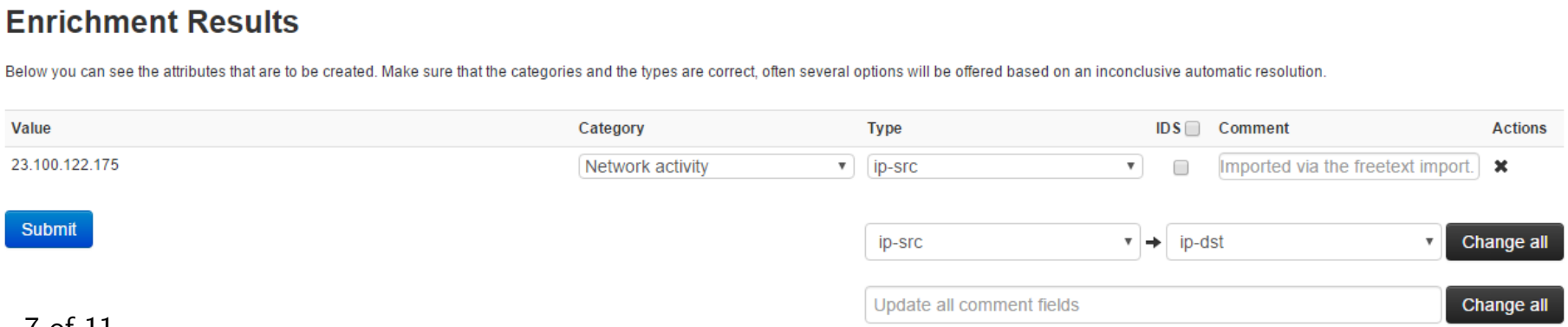
```
{"module": "dns", "hostname": "www.circl.lu"}
```

- and the response of the dns module:

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

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





MISP modules - configuration in the UI

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.