Contributing to the MISP Project

Become part of the community to design, develop and improve information sharing

Team CIRCL

http://www.misp-project.org/
Twitter: @MISPProject

MISP Training @ Luxembourg December Edition 2019
20191203
The MISP project has a Contributor Covenant Code of Conduct¹.

The goal of the code of conduct is to foster an open, fun and welcoming environment.

Another important aspect of the MISP projects is to welcome different areas of expertise in information sharing and analysis. The diversity of the MISP community is important to make the project useful for everyone.

¹https://github.com/MISP/MISP/code_of_conduct.md
The most common way to contribute to the MISP project is to report a bug, issues or suggesting features.

Each project (MISP core, misp-modules, misp-book, misp-taxonomies, misp-galaxy, misp-object or PyMISP) has their own issue management.

Don’t forget that you can cross-reference issues from other sub-projects.

If you know an answer or could help on a specific issue, we welcome all contributions including useful comments to reach a resolution.
Reporting security vulnerabilities

If you find security vulnerabilities (even minor ones) in MISP project, send an encrypted email (info@circl.lu) with the details and especially how to reproduce the issues. Avoid to share publicly the vulnerability before a fix is available in MISP. PGP key fingerprint: CA57 2205 C002 4E06 BA70 BE89 EAAD CFFC 22BD 4CD5.

We usually fix reported and confirmed security vulnerabilities in less than 48 hours.

We will request a CVE number if the reporters didn’t ask for one (don’t forget to mention how you want to be credited).
The majority of the repositories within the MISP GitHub organisation includes automatic integration with TravisCI.

If you contribute and make a pull-request, **verify if your changes affect the result of the tests.**

Automatic integration is not perfect including Travis but it’s a quick win to catch new bugs or major issues in contribution.

When you do a pull-request, TravisCI is automatically called².

- If this fails, no worries, **review the output at Travis** (it’s not always you).

We are working on additional automatic tests including unit testing for the MISP core software (contributors are welcome).

²https://travis-ci.org/MISP
All JSON format (galaxy, taxonomies, objects or warning-lists) are described in a JSON Schema\(^3\).

The TravisCI tests are including JSON validation (via jq) and validated with the associated JSON schema.

How to contribute a JSON library (objects, taxonomies, galaxy or warning-list):

▶ If you update a JSON library, don’t forget to run `jq_all_the_things.sh`. It’s fast and easy. If it fails, review your JSON.
▶ Commit your code and make a pull-request.

Documentations (in PDF and HTML format) for the libraries are automatically generated from the JSON via asciidoctor\(^4\).

---

\(^3\) `schema_name.json`

\(^4\) Example [https://github.com/MISP/misp-galaxy/blob/master/tools/ascidoc_galaxy.py](https://github.com/MISP/misp-galaxy/blob/master/tools/ascidoc_galaxy.py)
In addition to the automatic generation of documentations from JSON files, we maintain misp-book\(^5\) which is a generic documentation for MISP including usage, API documentation, best practices and specific configuration settings.

The book is generated in HTML, PDF, epub and mobi using GitBook\(^6\) which is a framework to write documentation in MarkDown format.

TravisCI is included in misp-book and the book generation is tested at each commit.

The MISP book is regularly published on misp-project.org and circl.lu website.

Contributors are welcome especially for new topics\(^7\) and also fixing our broken english.

\(^5\)https://github.com/MISP/misp-book
\(^6\)https://github.com/GitbookIO
\(^7\)Topics of interest are analysts best-practices,
If you want to contribute to our IETF Internet-Draft for the MISP standard, misp-rfc is the repository where to contribute.

**Update only the markdown file**, the XML and ASCII for the IETF I-D are automatically generated.

If a major release or updates happen in the format, we will publish the I-D to the IETF.

The process is always MISP implementation → IETF I-D updates.

---

---

\(^8\)https://github.com/MISP/misp-rfc

\(^9\)https://datatracker.ietf.org/doc/search/?name=misp&activedrafts=on&rfcs=on