The MANTIS Framework
Cyber-Threat Intelligence Mgmt. for CERTs

B. Grobauer, S. Berger, J. Göbel, T. Schreck, J. Wallinger | Siemens CERT
Note

- MANTIS is available as Open Source under GPL v2+ from https://github.com/siemens/django-mantis

- At time of this presentation (2014-06-24), the publicly available revision is MANTIS 0.2.0

- The examples shown in this talk are based on MANTIS 0.3.0

- MANTIS 0.3.0 will be released within the next few weeks:
  - either follow the repository on github
  - or subscribe to the MANTIS mailing list by sending a mail to mantis-ti-discussion-join@lists.trusted-introducer.org.
In a big corporation, there are many sources of cyber threat intelligence ....

- CISO
- SOC(s)
- Local Security Teams
- CERT
- CISO Community
- IT Provider
- Scan Service
- IT Provider
- Commercial TI Provider(s)
- External Partners
- External Incident Reporters
- External Partners
- External Incident Reporters
- Commercial TI Provider(s)
- External Partners
- External Incident Reporters
… and we need the full picture!

- CISO
- SOC(s)
- CERT
- Commercial TI Provider(s)
- External Partners
- External Incident Reporters
- Local Security Teams
- CISO Community
- Scan Service
- IT Provider
- Commercial TI Provider(s)
- External Partners
- External Incident Reporters
With MANTIS, we are working towards a tool that provides us with this full picture!
Today, there are several open-source tools that cover aspects of cyber threat intelligence management … … what are their distinguishing features?

MISP

MANTIS

(upcoming fall 2014 as Open Source)
What are the distinguishing features of a cyber threat intelligence management solution?

**Internal data model**

- XML (maybe)
- ?
- XML (maybe)

**Supported functionality**
*THE* basic design decision when implementing a solution for managing cyber threat intelligence: The internal data model

- **Genesis**
  What does your data model look like?
  - Home-brew
  - Somehow derived from a standard

- **Distance**
  How close is your data model to the (main) exchange standard(s) you are going to utilize?

- **Flexibility**
  - If the exchange standard allows very flexible usage: does your model, too, or do you narrow things down?
  - Can your model cope with moderate revision changes?
Genesis of the internal data model:
Arguments against home-brew models

- Homebrew means re-doing work others have already done (and that probably much more thoroughly than you have time for)

- Homebrew necessarily increases “distance” (see next slide)
Implications of „distance“ between the exchange standard and your data model:
Import and Export

- The further removed your internal data model is, the more you have to work for import and export.
- The real problem is the import: what to do with information that cannot be mapped into your internal data model?
  - reject and don’t import at all?
  - import partially (as far as it fits your data model?)
Flexibility: two extremes

Extremely flexible
- Just dump each file into an XML database (assuming that your main standard is in XML) …

Rather inflexible
- Create a database model for a given revision of some part of the standard
Implications of flexibility: Processing

- Flexibility eases import, but makes processing more complicated, since you cannot assume that things always look the same:
  - **automated mechanisms** must be able to deal with different representations of data … and in all likelihood will fail in some cases
  - **visualization/presentation** to the user becomes more complicated; your users will require a higher level of expertise regarding the data format
  - **export** becomes a challenge: you have imported data in revisions X, Y, and Z of a given standard; to what revision can you export?
Our choices for the MANTIS data model

- **Genesis:** “stand on the shoulders of giants” – the data model mirrors the threat intelligence exchanges standards that are relevant to us

- **Distance:** exchange standards and data model are very close (for details see next few slides)

- **Flexibility:**
  - regarding import: the Mantis importer is very forgiving and will import,
    - e.g., different revisions of STIX/CybOX in a sensible way with relatively little effort in adapting the importer to revision changes
  - XML that does quite conform to a standard’s XML schema
  - regarding the challenges wrt. processing and export: much of this is still future work … but following the “crawl, walk, run” approach: we are already able to crawl …
Why do we need maximum tolerance for exchange data formats and their revisions?

- At the moment, we cannot do without OpenIOC, so a STIX/CybOX-exclusive solution will not work. And it looks like we will also start importing the MISP data format …
  ➔ need to be able to import several standards

- I bet you that two years, after STIX 3.0 has been released, there will still be persons or tools that keep sending you STIX 1.0.1 …
  ➔ need to be able to import different revisions
MANTIS’s data model: pretty flexible, but a trying to do a bit more than just dumping XMLs or JSONs

- If you look at STIX and CybOX, you see that XML’s hierarchical structure is used for two different purposes:
  - modelling of containment relations between different objects
  - description of facts

[Diagram showing MANTIS's data model with Observable, Event, Action, File, Filename, Filepath, Hashes, Type (MD5), Value (6E48C3...)]

This, MANTIS preserves
This, MANTIS flattens into a list of “fact term”-value pairs ... and deduplicates these facts
Example: A CybOX Observable XML Source

```xml
<cybox:Observable id="example:Observable-a727a717-1852-4c79-9a16-2f3a8b4632c2"/>
    <cybox:Event id="example:Event-44578866-b0c5-4551-84dd-0f1f02f8210f">
        <cybox:Action id="example:Action-a18a058c-effa-4060-b8be-25e1b1ade75f" action_status="Success" context="Host" timestamp="2013-04-08T09:22:00.0Z">
            <cybox:Type xsi:type="cyboxVocabs:ActionTypeVocab-1.0">Create</cybox:Type>
            <cybox:Name xsi:type="cyboxVocabs:ActionNameVocab-1.0">Create File</cybox:Name>
            <cybox:Associated_Objects>
                <cybox:Associated_Object id="example:Object-5ec92e95-a31f-470b-97c4-aa9046189fbb"/>
                    <cybox:Properties xsi:type="FileObj:FileObjectType">
                        <FileObj:File_Name>fooobar.dll</FileObj:File_Name>
                        <FileObj:File_Path>C:\Windows\system32\</FileObj:File_Path>
                        <FileObj:Hashes>
                            <cyboxCommon:Hash>
                                <cyboxCommon:Type>MD5</cyboxCommon:Type>
                                <cyboxCommon:Simple_Hash_Value datatype="hexBinary">6E48C34B742A931EC2CE90ABD7DAC6A</cyboxCommon:Simple_Hash_Value>
                            </cyboxCommon:Hash>
                        </FileObj:Hashes>
                    </cybox:Properties>
                    <cybox:Association_Type xsi:type="cyboxVocabs:ActionObjectAssociationTypeVocab-1.0">Affected</cybox:Association_Type>
                </cybox:Associated_Object>
            </cybox:Associated_Objects>
        </cybox:Action>
    </cybox:Actions>
</cybox:Event>
</cybox:Observable>
```
Example: Importing a CybOX 2.0 Observable

XML Source: Focusing on objects and facts

```xml
<cybox:Observable id="example:Observable-a727a717-1852-4c79-9a16-2f3a8b4632c2">
  <cybox:Event id="example:Event-44578866-b0c5-4551-84dd-0f1f02f8210f">
    <cybox:Actions>
      <cybox:Action id="example:Action-a18a058c-effa-4060-b8be-25e1b1ade75f" action_status="Success" context="Host" timestamp="2013-04-08T09:22:00.0Z">
        <cybox:Type xsi:type="cyboxVocab:ActionTypeVocab-1.0">Create</cybox:Type>
        <cybox:Name xsi:type="cyboxVocab:ActionNameVocab-1.0">Create File</cybox:Name>
        <cybox:Associated_Objects>
          <cybox:Associated_Object id="example:Object-5ec92e95-a31f-470b-97c4-aa9046189fbb">
            <cybox:Properties xsi:type="FileObj FileObjectType">
              <FileObj:File_Name>foobar.dll</FileObj:File_Name>
              <FileObj:File_Path>C:\Windows\system32</FileObj:File_Path>
              <FileObj:Hashes>
                <cyboxCommon:Hash>
                  <cyboxCommon:Type>MD5</cyboxCommon:Type>
                  <cyboxCommon:Simple_Hash_Value datatype="hexBinary">6E48C348D742A931EC2CE90ABD7DAC6A</cyboxCommon:Simple_Hash_Value>
                </cyboxCommon:Hash>
              </FileObj:Hashes>
            </cybox:Properties>
          </cybox:Associated_Object>
        </cybox:Associated_Objects>
      </cybox:Action>
    </cybox:Actions>
  </cybox:Event>
</cybox:Observable>
```

An action that creates a file with certain file name, file path and hash
Example: A CybOX Observable XML Source

Defining object boundaries

In the XML, an identifier is provided for each structure that naturally gives rise to an information object of its own.
Example: A CybOX Observable XML Source

Extracting „flat“ facts from hierarchical XML

The facts we are really interested into about the observed file are:

- Properties/File_Name = foobar.dll
- Properties/File_Path = C:\Windows\system32
- Properties/Hashes/Hash/Type = MD5
- Properties/Hashes/Hash/Simple_Hash_Value = 6E48C34D74A931EC2CE90ABD7DAC6A
Example: Importing a CybOX 2.0 Observable

Resulting Structure

Info Object: **Event**: Action(s): Create File (6 facts) ...

<table>
<thead>
<tr>
<th>Identifying data</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Identifier</strong></td>
</tr>
<tr>
<td><strong>Timestamp</strong></td>
</tr>
<tr>
<td><strong>Type</strong></td>
</tr>
<tr>
<td><strong>Import Timestamp</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Facts</th>
</tr>
</thead>
</table>
| **Event** | Action(s): Create File (6 facts) ...

---

© Siemens AG 2014. All rights reserved
Example: Importing a CybOX 2.0 Observable
Resulting Structure

<table>
<thead>
<tr>
<th>Identifier</th>
<th><a href="http://example.com:Event-44578866-b0c5-4551-84dd-0f1f02f8210f">http://example.com:Event-44578866-b0c5-4551-84dd-0f1f02f8210f</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>cybox.mitre.org:Event 2 (<a href="http://cybox.mitre.org/cybox">http://cybox.mitre.org/cybox</a>)</td>
</tr>
<tr>
<td>Timestamp</td>
<td>2014-06-19T00:07:21.713707+02:00</td>
</tr>
<tr>
<td>Import Timestamp</td>
<td>2014-06-19T00:07:21.713707+02:00</td>
</tr>
</tbody>
</table>

Facts

<table>
<thead>
<tr>
<th>Actions</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Create File (6 facts)</td>
</tr>
</tbody>
</table>

- @action_status: Success
- @context: Host
- @timestamp: 2013-04-08T09:22:00.0Z
Example: Importing a CybOX 2.0 Observable

Resulting Structure

- Cybox 2.0 Observable
  - Info Object
    - Fact 1
    - Fact n
- Cybox 2.0 Event
  - Info Object
    - Fact 1
    - Fact n
- Cybox 2.0 Action
  - Info Object
    - Fact 1
    - Fact n
- Cybox 2.0 File Object
  - Info Object
    - Fact 1
    - Fact n

Facts

- @action_status: Success
- @context: Host
- @timestamp: 2013-04-08T09:22:00.0Z
- Type: Create
- Name: Create File
- Associated_Objects: 
  - Associated_Object
  - foobar dll (5 facts)
Example: Importing a CybOX 2.0 Observable

Resulting Structure

Presentation to user mirrors hierarchical composition of facts, but underlying data model contains flattened „fact terms“
(Cyber) Threat Intelligence Tooling: A reference frame regarding functionality
Siemens CERT‘s MANTIS Framework

- MANTIS is based on Django, the Python-based web application framework.

- The current version of MANTIS contains import modules for STIX/CybOX, OpenIOC, and IODEF, but the architecture is of MANTIS is generic and provides for easy generation of additional import modules for other standards.
(Cyber) Threat Intelligence Tooling: A reference frame regarding functionality

How to import threat intelligence data that is expressed in relevant standards into your system?
MANTIS
TI-Standard Import

- We have talked about how STIX/CybOX XML is imported into MANTIS
- The MANTIS framework provides a generic importer class that has been customized to import
  - CybOX/STIX
  - OpenIOC indicators
  - IODEF
- Importer function can be triggered
  - programatically (using Celery for task management)
  - via commandline for scripting
  - via GUI ‘XML Import’ dialogue
  - via authoring GUI (import OpenIOC into STIX Test Mechanism)
MANTIS
TI-Standard Import
XML-Import via GUI

Name: Import of XML via GUI

Xml:

```
<stix:STIX_Package
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:stix="http://stix.mitre.org/stix-1"
 xmlns:indicator="http://stix.mitre.org/Indicator-2"
 xmlns:cybox="http://cybox.mitre.org/cybox-2"
 xmlns:DomainNameObject="http://cybox.mitre.org/objects#DomainNameObject-1"
 xmlns:Vocab="http://cybox.mitre.org/default_vocabularies-2"
 xmlns:Vocab2="http://stix.mitre.org/default_vocabularies-1"
 xmlns:example="http://example.com"
 xsi:schemaLocation="http://stix.mitre.org/stix-1 ../stix_core.xsd
 http://stix.mitre.org/Indicator-2 ../indicator.xsd
 http://cybox.mitre.org/default_vocabularies-2 ../cybox/cybox_default_vocabularies.xsd
 http://cybox.mitre.org/default_vocabularies-1 ../cybox_default_vocabularies.xsd
 http://cybox.mitre.org/objects#DomainNameObject-1 ../cybox/objects/Domain_Name_Object.xsd"
 id="example:STIXPackage-f61cd874-494d-4194-a3e6-6b487dbb6d6e"
 timestamp="2014-05-08T09:00:00.000000Z"
 version="1.1.1"
>
 <stix:STIX_Header>
   <stix:Title>Example watchlist that contains domain information.</stix:Title>
   <stix:Package_Intent xsi:type="stixVocab:PackageIntentVocab-1.0">Indicators - Watchlist</stix:Package_Intent>
 </stix:STIX_Header>
 <stix:Indicators>
   <stix:Indicator xsi:type="indicator:IndicatorType" id="example:Indicator-2e20c5b2-56fa-46cd-9662-bf199c69d2c9" timestamp="2014-05-

ATTENTION: Make sure that the identifier namespaces occurring in the XML are contained in your allowed namespaces (see display on right-hand side)!!! Otherwise, the created objects will be moved into a temporary namespace!!!
```

Import
MANTIS
Import of OpenIOC as part of STIX Authoring

Drag & Drop
How to organize interfaces and transport between sharing partners
For organizing interfaces for import from external sharing partners, we plan to leverage MANTIS.

- Luckily, MITRE’s TAXII proof-of-concept implementation YETI is also running on top of Django.

- Imports registered by YETI can be made to trigger an import task in MANTIS (using Celery for asynchronous processing).
(Cyber )Threat Intelligence Tooling:
A reference frame regarding functionality

How to make sense of the collected TI data?
MANTIS
Filtering by Object Property

Filter Parameters

InfoObject Type: 
Object Type matches: 
InfoObject Family: 
Name contains: 
ID Namespace: 
ID contains: 
Object Creation Timestamp: Any date
Import Timestamp: Any date
Marking ID contains: 

Submit Query  Save Search

Filter Parameters

InfoObject Type: stix.mitre.org:STIXPackage
Object Type matches: 
InfoObject Family: 
Name contains: 
ID Namespace: 
ID contains: 
Object Creation Timestamp: 
Import Timestamp: 
Marking ID contains: 

Submit Query  Save Search
### Filter Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fact term (w/o attribute) matches:</td>
<td>Hashes/Hash/SimpleHashValue</td>
</tr>
<tr>
<td>Attribute matches:</td>
<td></td>
</tr>
<tr>
<td>Value contains:</td>
<td></td>
</tr>
<tr>
<td>Object name contains:</td>
<td></td>
</tr>
<tr>
<td>Object Timestamp:</td>
<td>Any date</td>
</tr>
<tr>
<td>Import Timestamp:</td>
<td>Any date</td>
</tr>
<tr>
<td>ID Namespace:</td>
<td>---------</td>
</tr>
<tr>
<td>InfoObject Type:</td>
<td>---------</td>
</tr>
<tr>
<td>Object Type name contains:</td>
<td></td>
</tr>
<tr>
<td>Marking ID contains:</td>
<td></td>
</tr>
</tbody>
</table>

---

**Submit Query**  **Save Search**
### Displaying Information Objects

#### Identifying data

<table>
<thead>
<tr>
<th>Identifier</th>
<th><a href="http://www.mandiant.comttp-33159b98-3264-4e10-a968-d67975b6272f">http://www.mandiant.comttp-33159b98-3264-4e10-a968-d67975b6272f</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>stix.mitre.org:TTP 1 (<a href="http://stix.mitre.org/TTP">http://stix.mitre.org/TTP</a>)</td>
</tr>
<tr>
<td>Timestamp</td>
<td>2013-02-19T01:00:02+01:00</td>
</tr>
<tr>
<td>Import Timestamp</td>
<td>2014-06-18T13:29:09.473642+02:00</td>
</tr>
</tbody>
</table>

#### Facts

<table>
<thead>
<tr>
<th>@xsi:type</th>
<th>TTPType</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>HTRAN Malware C2</td>
</tr>
<tr>
<td>Behavior</td>
<td>Malware Malware Instance</td>
</tr>
<tr>
<td>Type</td>
<td>Relay</td>
</tr>
<tr>
<td>Name</td>
<td>HUC Packet Transmit Tool (HTRAN)</td>
</tr>
<tr>
<td>Description</td>
<td>When APT1 attackers are not using WFRC2, they require a</td>
</tr>
</tbody>
</table>

#### Resources

<table>
<thead>
<tr>
<th>Infrastructure</th>
<th>Type</th>
<th>Leveraged IP Blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Observable_</td>
<td>Observable</td>
<td>143.89.255.255 (condition InclusiveBetween) (4 facts)</td>
</tr>
<tr>
<td>Characterization</td>
<td>Object</td>
<td></td>
</tr>
<tr>
<td>Observable</td>
<td>Observable</td>
<td>143.89.255.255 (condition InclusiveBetween) (4 facts)</td>
</tr>
<tr>
<td></td>
<td>Object</td>
<td></td>
</tr>
</tbody>
</table>
MANTIS
Visualizing Object Relations
MANTIS
Searching Objects and Facts
What we can search for

Object Properties

Markings

Fact Terms and Fact Values
Filter Parameters

fact: [Properties/Value] regexp "business"
| object: identifier.namespace contains 'mandiant.com'
   && object_type.name contains 'URIObject'
| marked_by: (fact: [Marking_Structure/Statement] contains 'APT1')
## Saved searches for user John Doe

<table>
<thead>
<tr>
<th>Name</th>
<th>Link</th>
<th>Custom Query</th>
<th>Parameters</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter for STIX Packages</td>
<td>mantis/ViewInfo/Object</td>
<td>0</td>
<td>idobject_type:72</td>
<td>✅ ✓</td>
</tr>
<tr>
<td>Filter for OpenIOC Indicator</td>
<td>mantis/ViewInfo/Object</td>
<td>0</td>
<td>idobject_type:71</td>
<td>✅ ✓</td>
</tr>
<tr>
<td>Network Indicators of past two days</td>
<td>mantis/Search/CustomFactSearch</td>
<td>fact: fact_term regexp &quot;URI[Domain_Name]Properties[Value]&quot; &amp; attribute = &quot;condition&quot; &amp; @[condition] = &quot;Equals&quot;</td>
<td>paginate_by:50</td>
<td>✅ ✓</td>
</tr>
<tr>
<td>This is a temporary entry and won't be persisted unless you give it a name and press save</td>
<td>mantis/ViewInfo/Object</td>
<td></td>
<td>idobject_type:157</td>
<td>✅ ✓</td>
</tr>
</tbody>
</table>
(Cyber) Threat Intelligence Tooling: A reference frame regarding functionality

Support of user collaboration (intra- vs inter-instance)

Display & Visualize & Search & Analyse

Enrich & Process

Author & Edit

Policy Definition & Enforcement

TI-Standard Export (i.e., conversion from internal data model)

TI-Standard Transport

How to work with collected threat intelligence data (trigger activities, enrich with information, …)
MANTIS
Processing & Enrichment of Data

- MANTIS 0.3.0 does not offer standard methods for processing and enriching data

- In our internal instance customized for our use, we employ base classes offered by the MANTIS framework to implement
  - actions on objects
  - marking of objects with additional information

- First standard processing/enrichment method likely to be implemented by the next MANTIS release will be object tagging (i.e., marking of objects with relatively restricted markings)
(Cyber )Threat Intelligence Tooling:
A reference frame regarding functionality

Support of user collaboration
(intra- vs inter-instance)

Display & Visualize
& Search & Analyse

Enrich & Process

Author & Edit

How to author and edit threat intelligence data in the system
Interlude: The problem of authoring STIX and CybOX

- STIX and CybOX are complex, …really, really complex
- The STIX/CybOX community is in the process of working out the intended usage of STIX/CybOX for standard use-cases (just last week, a discussion of how to communicate sightings of a given indicator got started on the mailing list)
- There will be organization/company-specific specializations of standard use-cases.
- Your tool needs a way to codify standard use cases such that the user can concentrate on entering the right data, while the tool takes care of generating STIX/CybOX that follows the intended usage for the particular use-case
MANTIS’s approach to authoring and editing threat intelligence

Authoring Interface for Use-Case „foo“

uses edit forms of

Standard Template for authoring CybOX object X in variant Y

uses transformers of

Objects originating from imported reports maintain a relationship with the defining JSON structure; the report can be modified by re-opening the JSON, editing it and carrying out another import: existing objects are then overwritten with the newly created version.
MANTIS
Authoring Campaign and Threat Actor

Campaign Information

Name: 
Title: First observed campaign against HR Department with aim of tax fraud carried out by Robin Hood
Description: This is the first time we observe a campaign by adversary Robin Hood against our HR department with the obvious aim of stealing data that can be used to file tax returns in the name of our employees.
Status: Ongoing
Activity timestamp from: 
Activity timestamp to: 
Confidence: High

Threat Actor Information

Identity name:
Identity aliases:
Title:
Description:
Confidence:

[six.mitre.org:ThreatActor]

- threat-actor-f1ce5ae-486b-486b-ac8-e7f75988ee9
  Comment View
- threat-actor-5ab429ac-5f2d-4a1b-b57d-cbf3c663b0
  Comment Group
- threat-actor-5b6b258d-4d4c-4b4a-9220-2b37b
  Communist Party of China
- threat-actor-bb623a-4b84-d280-9702-0220287b
  GSD 3rd Department
- threat-actor-9e624665-2708-436f-9b4e-2b988a63690b
  GSD 3rd Department / 2nd Bureau
- threat-actor-6df10344-0c22-467b-8d04-96c4d9bd1df
  People's Liberation Army
- threat-actor-2a6c05e-8b04-489e-a170-0e0d15a6c0b
  PLA General Staff
- threat-actor-d9e1d2e1-9d4f-414e-94f1-3c58364a422b
MANTIS
Authoring Indicator Information

Indicator Configuration

cert_my_organization:Indicator-4775a6c3-58cf-3dee-91f1-349568260f0f

Indicator producer:

Indicator title: Robin H00d's RAT

Indicator description: The RAT used by Robin H00d is disguised as file "

Indicator confidence: High
MANTIS
Authoring File Object via Drag & Drop

super_cool_game.exe
MANTIS
Authoring Generic URI Object

Generic URI
super.evil.com (4 facts)

Domain contacted by Robin Hood's RAT.
Observable Description

Type: Domain Name
Value: super.evil.com

File
super_cool_game.exe (58880 Bytes)

Observable Templates
- Address
- Artifact
- DNS Record
- Email Message (Default)
- File
- HTTP Session
- Port
- Generic URI
- Windows Service
# MANTIS

## Putting together the STIX Package

### STIX Package

<table>
<thead>
<tr>
<th>Package Contents</th>
<th>Package Meta</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Stix</strong></td>
<td><strong>Test STIX Package</strong></td>
</tr>
<tr>
<td></td>
<td><strong>This is a test of Mantis Authoring</strong></td>
</tr>
<tr>
<td></td>
<td><strong>White</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Save Draft</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Show JSON</strong></td>
</tr>
</tbody>
</table>

#### Observables

<table>
<thead>
<tr>
<th>File</th>
</tr>
</thead>
<tbody>
<tr>
<td>super_cool_game.exe (58080 Bytes)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Generic URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>super.evil.com (4 facts)</td>
</tr>
</tbody>
</table>
MANTIS

Viewing the resulting XML
(Cyber) Threat Intelligence Tooling:
A reference frame regarding functionality

To which extent can access to data & functionality be restricted (authorization concept)?

Support of user collaboration
(intra- vs inter-instance)

Display & Visualize

Enrich

Author

Access Restrictions
A user can be member of one or more groups (standard mechanism offered by Django)

By associating a group with identifier namespace information, it becomes an Authoring Group

Namespace information contains

- **default namespace**: objects created via authoring interface are created in identifier namespace as specified by *default namespace*

- **allowed namespace**: objects created by user e.g., via XML Import interface, may only carry identifiers with an allowed identifier namespace

A user can only access the author interface for reports created within an authoring group of which he is a member
MANTIS
Authoring Groups restrict authoring & editing

**Authoring Group: CERT Team**

**Default Namespace**
cert.my-organization.com

**Allowed Namespaces**
cert.my-organization.com

---

**Xml:**

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<stix:STIX_Package
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:stix="http://stix.mitre.org/stix-1"
 xmlns:indicator="http://stix.mitre.org/Indicator-2"
 xmlns:cybox="http://cybox.mitre.org/cybox-2"
 xmlns:DomainNameObj="http://cybox.mitre.org/objects#DomainNameObject"
 xmlns:cyboxVocabs="http://cybox.mitre.org/default_vocabularies-2"
 xmlns:cyboxVocab="http://cybox.mitre.org/default_vocabulary-1"
 xmlns:example="http://example.com"
 xsi:schemaLocation="http://stix.mitre.org/stix-1 ../stix_core.xsd"
 http://stix.mitre.org/Indicator-2 ../indicator.xsd"
 http://cybox.mitre.org/default_vocabularies-2 ../cybox/cybox_default_vocabularies.xsd"
 http://stix.mitre.org/default_vocabulary-1 ../stix_default_vocabulary.xsd"
 http://cybox.mitre.org/objects#DomainNameObject-1 ../cybox/objects/Domai
 id="example:STIXPackage-f61cd874-494d-4194-a3e6-6b467dbb6d6e"
 timestamp="2014-05-08T09:00:00.000000Z"
 version="1.1.1"/>
</stix:STIX_Header>

<stix:Title>Example watchlist that contains domain information.</stix:Title>
<stix:Package_Intent xsi:type="stixVocabs:PackageIntentVocab-1.0">Indicators - Watchlist</stix:Package_Intent>

<stix:Indicators>

<stix:Indicator xsi:type="indicator:IndicatorType" id="example:Indicator-2e20c5b2-55fa-46cd-9662-8f199c69d2c9" timestamp="2014-05-
```

**ATTENTION:** Make sure that the identifier namespaces occurring in the XML are contained in your allowed namespaces (see display on right-hand side)!!! Otherwise, the created objects will be moved into a temporary namespace!!!

---

Import
MANTIS
Authoring Groups restrict authoring & editing
(Cyber) Threat Intelligence Tooling: A reference frame regarding functionality

Support of user collaboration (intra- vs inter-instance)

How is collaboration of users on managing cyber-threat intelligence data supported?

Search & Analyse
Process
Edit

Access Restrictions

Export to Security Tooling (Log Analysis, SIEM, IDS, IPS, ...)

Synchronization with other instances

Export Policy Definition & Enforcement (i.e., conversion from internal data model)

TI-Standard Export (i.e., conversion from internal data model)

TI-Standard Transport

TI-Standard Import (i.e., conversion to internal data model)

Import of supporting data (OSINT etc.)

Arefact 2 TI-Import

Synchronization with other instances

TI-Standard Transport

TI-Standard Transport

Corporate Technology, RTC ITS CCS © Siemens AG 2014. All rights reserved
MANTIS
Users can cooperate on reports within authoring groups

- A user can access via the authoring interface authored within his authoring group(s)
- Reports may have an owner: as long as an owner holds the report, no other user may edit the report
- A report can be released by its owner; importing the report into MANTIS releases the report automatically
- Users can take ownership of a report currently owned by another user (to be used with care!)
### Drafts and Imports of Authoring Group CERT Team

<table>
<thead>
<tr>
<th>Owner</th>
<th>Name</th>
<th>Status</th>
<th>Timestamp</th>
<th>Take from owner</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Test STIX Package</td>
<td>Imported Q</td>
<td>June 18, 2014, 11:07 p.m.</td>
<td>Submit from owner</td>
</tr>
</tbody>
</table>

**Filter Parameters**

<table>
<thead>
<tr>
<th>Import Timestamp:</th>
<th>Any date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name contains:</td>
<td></td>
</tr>
<tr>
<td>Status:</td>
<td>Any status</td>
</tr>
<tr>
<td>User:</td>
<td></td>
</tr>
</tbody>
</table>

**Authoring Group: CERT Team**

- **Default Namespace**: cert.my-organization.com
- **Allowed Namespaces**: cert.my-organization.com

**Buttons**: Submit Query, Save Search
### History of 'Test STIX Package'

<table>
<thead>
<tr>
<th>Owner</th>
<th>Status</th>
<th>Kind</th>
<th>Timestamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>John Doe</td>
<td>Imported</td>
<td>JSON (Dingos Authoring)</td>
<td>June 18, 2014, 10:55 p.m.</td>
</tr>
<tr>
<td>John Doe</td>
<td>Draft</td>
<td>JSON (Dingos Authoring)</td>
<td>June 18, 2014, 10:55 p.m.</td>
</tr>
<tr>
<td>John Doe</td>
<td>Draft</td>
<td>JSON (Dingos Authoring)</td>
<td>June 18, 2014, 10:43 p.m.</td>
</tr>
<tr>
<td>John Doe</td>
<td>Draft</td>
<td>JSON (Dingos Authoring)</td>
<td>June 18, 2014, 10:42 p.m.</td>
</tr>
</tbody>
</table>
(Cyber )Threat Intelligence Tooling: A reference frame regarding functionality

How to feed security tooling (log analysis tools, SIEM, IDS, IPS, etc. with relevant threat intelligence)
MANTIS
Feeding security tools with threat intelligence via saved searches

- Custom searches
  - can be saved
  - can return results as CSV, simple JSON, etc.
- Concept for feeding security tools with simple indicator lists:
  - Generate a saved search that pulls relevant data out of the system
  - Allow tools access to saved searches via REST interface
MANTIS
Providing search results as csv
(Cyber )Threat Intelligence Tooling: A reference frame regarding functionality

How to get your information into a standard format fit for export?
MANTIS
Towards exporting data

- As discussed above when talking about the flexible data model: flexible import makes export challenging
  
  Possible concepts:
  
  - always export to a given revision of the standard; for data imported in older revisions, some data may be lost or misrepresented
  
  - always export to the revision that was imported (but for this we would need to have STIX/CybOX python bindings in different revisions in parallel
  
  ...?
  
- **But:** for our main use case of exporting self-authored data, we are all set for always exporting to the latest revision:
  
  - Upgrade transformers to new revision
  
  - Regenerate STIX/CybOX for all reports
Take away #1
Distinguishing features of Threat Intelligence Management Systems

- Genesis?
- Distance?
- Flexibility?
Caveat:
What MANTIS is and isn’t

- **MANTIS is** an *alpha/early beta implementation* of a framework for managing cyber threat intelligence expressed in standards such as STIX, CybOX, OpenIOC, IODEF, etc.
- Our aims of providing MANTIS as open source are:
  - To aide discussions about tooling for emerging standards such as STIX, CybOX et al.
  - To lower the entrance barrier for organizations and teams (esp. CERT teams) in using emerging standards for cyber-threat intelligence management and exchange.
  - To provide a platform on the basis of which research and community-driven development in the area of cyber-threat intelligence management can occur.

- **MANTIS isn’t** a finished tool or project: we like to think that it provides a solid basis on which cyber-threat intelligence management can be built up upon, but if you expect something that out of the box covers all aspects of cyber-threat intelligence management or are unable/unwilling to dive into Django and Python code and fix/modify according to your requirements, MANTIS isn’t for you. This may change sometime in the future when Mantis reaches version 1.0.0 … but currently, we are at 0.3.0…

- **MANTIS (currently) isn’t** a tool fit for importing *huge* datasets or huge numbers of datasets. This situation may change at some point of time with more stream-lined importers, but MANTIS is really not intended to deal with very big data the way log management solutions are.
Where to get MANTIS?

Access to the Mantis source code for installation:

- Either via git clone from the Mantis Github Repository (https://github.com/siemens/django-mantis.git) (recommended):
  ```
git clone https://github.com/siemens/django-mantis.git
  ```
- Or via download as zip package from https://github.com/siemens/django-mantis/archive/master.zip

There is a mailing list for discussions, questions, etc.:

- Subscribe to the mailing list by sending a mail to Mantis-ti-discussion-join@lists.trusted-introducer.org.
- The archives of the mailing list are available via Nabble (http://mantis-threat-intelligence-management-framework-discussion-list.57317.x6.nabble.com/)

Many thanks to the TF-CSIRT Trusted Introducer for their support in hosting the list!

All issues regarding Mantis and its components are tracked on the Mantis Issue Tracker (https://github.com/siemens/django-mantis/issues?state=open)