## Contents

1 Versions 3

2 Contents 5
   2.1 Installation ................................................................. 5
   2.2 Getting Started ............................................................... 6
   2.3 Overview ................................................................. 8
   2.4 Examples ............................................................... 13
   2.5 APIs or bindings? ......................................................... 14

3 API Reference 17
   3.1 API Reference ............................................................... 17
   3.2 API Coverage ............................................................ 103

4 FAQ 107

5 Contributing 109

6 Indices and tables 111

Python Module Index 113
Version: 1.2.0.8

The python-stix library provides an API for developing and consuming Structured Threat Information eXpression (STIX) content. Developers can leverage the API to develop applications that create, consume, translate, or otherwise process STIX content. This page should help new developers get started with using this library. For more information about STIX, please refer to the STIX website.

Note: These docs provide standard reference for this Python library. For documentation on idiomatic usage and common patterns, as well as various STIX-related information and utilities, please visit the STIXProject at GitHub.
Each version of python-stix is designed to work with a single version of the STIX Language. The table below shows the latest version the library for each version of STIX.

<table>
<thead>
<tr>
<th>STIX Version</th>
<th>python-stix Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2</td>
<td>1.2.0.8 (PyPI) (GitHub)</td>
</tr>
<tr>
<td>1.1.1</td>
<td>1.1.1.15 (PyPI) (GitHub)</td>
</tr>
<tr>
<td>1.1.0</td>
<td>1.1.0.6 (PyPI) (GitHub)</td>
</tr>
<tr>
<td>1.0.1</td>
<td>1.0.1.1 (PyPI) (GitHub)</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0.0a7 (PyPI) (GitHub)</td>
</tr>
</tbody>
</table>

Users and developers working with multiple versions of STIX content may want to take a look at stix-ramrod, which is a library designed to update STIX and CybOX content.

Check out the Working with python-stix section for examples on how to integrate stix-ramrod and python-stix.
2.1 Installation

The installation of python-stix can be accomplished through a few different workflows.

2.1.1 Recommended Installation

Use pypi and pip:

$ pip install stix

You might also want to consider using a virtualenv. Please refer to the pip installation instructions for details regarding the installation of pip.

2.1.2 Dependencies

The python-stix library relies on some non-standard Python libraries for the processing of STIX content. Revisions of python-stix may depend on particular versions of dependencies to function correctly. These versions are detailed within the distutils setup.py installation script.

The following libraries are required to use python-stix:

- lxml - A Pythonic binding for the C libraries libxml2 and libxslt.
- python-cybox - A library for consuming and producing CybOX content.
- python-dateutil - A library for parsing datetime information.

Each of these can be installed with pip or by manually downloading packages from PyPI. On Windows, you will probably have the most luck using pre-compiled binaries for lxml. On Ubuntu (12.04 or 14.04), you should make sure the following packages are installed before attempting to compile lxml from source:

- libxml2-dev
- libxslt1-dev
- zlib1g-dev
2.1.3 Manual Installation

If you are unable to use pip, you can also install python-stix with `setuptools`. If you don’t already have setuptools installed, please install it before continuing.

1. Download and install the dependencies above. Although setuptools will generally install dependencies automatically, installing the dependencies manually beforehand helps distinguish errors in dependency installation from errors in stix installation. Make sure you check to ensure the versions you install are compatible with the version of stix you plan to install.

2. Download the desired version of stix from PyPI or the GitHub releases page. The steps below assume you are using the 1.2.0.8 release.

3. Extract the downloaded file. This will leave you with a directory named stix-1.2.0.8.

   $ tar -zxf stix-1.2.0.8.tar.gz
   $ ls
   stix-1.2.0.8 stix-1.2.0.8.tar.gz

   OR

   $ unzip stix-1.2.0.8.zip
   $ ls
   stix-1.2.0.8 stix-1.2.0.8.zip

4. Run the installation script.

   $ cd stix-1.2.0.8
   $ python setup.py install

5. Test the installation.

   $ python
   Python 2.7.6 (default, Mar 22 2014, 22:59:56)
   [GCC 4.8.2] on linux2
   Type "help", "copyright", "credits" or "license" for more information.
   >>> import stix
   >>>

   If you don’t see an `ImportError`, the installation was successful.

2.1.4 Further Information

If you’re new to installing Python packages, you can learn more at the Python Packaging User Guide, specifically the Installing Python Packages section.

Version: 1.2.0.8

2.2 Getting Started

This page gives an introduction to `python-stix` and how to use it.
2.2.1 Prerequisites

The python-stix library provides an API for creating or processing STIX content. As such, it is a developer tool that can be leveraged by those who know Python 2.7/3.3+ and are familiar with object-oriented programming practices, Python package layouts, and are comfortable with the installation of Python libraries. To contribute code to the python-stix repository, users must be familiar with git and GitHub pull request methodologies. Understanding XML, XML Schema, and the STIX language is also incredibly helpful when using python-stix in an application.

2.2.2 Your First STIX Application

Once you have installed python-stix, you can begin writing Python applications that consume or create STIX content!

Note: The python-stix library provides bindings and APIs, both of which can be used to parse and write STIX XML files. For in-depth description of the APIs, bindings, and the differences between the two, please refer to APIs or bindings?

Creating a STIX Package

```python
from stix.core import STIXPackage
# Import the STIX Package API
from stix.report import Report
# Import the STIX Report API
from stix.report.header import Header
# Import the STIX Report Header API

stix_package = STIXPackage()
# Create an instance of STIXPackage
stix_report = Report()
# Create a Report instance
stix_report.header = Header()
# Create a header for the report
stix_report.header.description = "Getting Started!"
# Set the description
stix_package.add(stix_report)
# Add the report to our STIX Package

print(stix_package.to_xml())
# Print the XML for this STIX Package
```

Parsing STIX XML

```python
from stix.core import STIXPackage
# Import the STIX Package API

fn = 'stix_content.xml'
# The STIX content filename
stix_package = STIXPackage.from_xml(fn)
# Parse using the from_xml() method
```

2.2.3 Examples

The python-stix GitHub repository contains several example scripts that help illustrate the capabilities of the APIs. These examples can be found here. Accompanying walkthrough slides are available. These scripts are simple command line utilities that can be executed by passing the name of the script to a Python interpreter.

Example:

$ python ex_01.py
Note: You must install python-stix before running these example scripts.

Version: 1.2.0.8

2.3 Overview

This page provides a quick overview needed to understand the inner workings of the python-stix library. If you prefer a more hands-on approach, browse the Examples.

Version: 1.2.0.8

2.3.1 ID Namespaces

By default, python-stix sets the default ID namespace to http://example.com with an alias of example. This results in STIX id declarations that look like id="example:Package-2813128d-f45e-41f7-b10a-20a5656e3785".

To change this, use the mixbox.idgen.set_id_namespace() method which takes a dictionary as a parameter.

```python
from stix.core import STIXPackage
from mixbox.idgen import set_id_namespace
from mixbox.namespaces import Namespace

NAMESPACE = Namespace("http://MY-NAMESPACE.com", "myNS")
set_id_namespace(NAMESPACE)  # new ids will be prefixed by "myNS"

stix_package = STIXPackage()  # id will be created automatically
print stix_package.to_xml()
```

Which outputs:

```xml
<stix:STIX_Package
   xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
   xmlns:myNS="http://MY-NAMESPACE.com"
   xmlns:stix="http://stix.mitre.org/stix-1"
   xmlns:xs="http://www.w3.org/2001/XMLSchema"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:xlink="http://www.w3.org/1999/xlink"
   id="myNS:Package-b2039368-9476-4a5b-8c1d-0ef5d1b37e06" version="1.2"/>
```

Success! The xmlns:myNS="http://MY-NAMESPACE.com" matches our NAMESPACE dictionary and the id attribute includes the myNS namespace alias.

Working With CybOX

When setting the ID namespace in python-stix, the ID namespace will also be set in python-cybox.

Version: 1.2.0.8
2.3.2 Controlled Vocabularies

Many fields in STIX leverage the `stixCommon:ControlledVocabularyStringType`, which acts as a base type for controlled vocabulary implementations. The STIX language defines a set of default controlled vocabularies which are found in the `stix_default_vocabs.xsd` XML Schema file.

The `python-stix` library contains a `stix.common.vocabs` module, which defines the `VocabString` class implementation of the schema `ControlledVocabularyStringType` as well as `VocabString` implementations which correspond to default controlled vocabularies.

For example, the `stix_default_vocabs.xsd` schema defines a controlled vocabulary for STIX Package Intents: `PackageIntentVocab-1.0`. The `stix.common.vocabs` module contains an analogous `PackageIntent` class, which acts as a derivation of `VocabString`.

Each `VocabString` implementation contains:

- A static list of class-level term attributes, each beginning with `TERM_` (e.g., `TERM_INDICATORS`)
- A tuple containing all allowed vocabulary terms: `_ALLOWED_VALUES`, which is used for input validation. This is generated via the `vocabs.register_vocab()` class decorator.
- Methods found on `stix.Entity`, such as `to_xml()`, `to_dict()`, `from_dict()`, etc.

Interacting With VocabString Fields

The following sections define ways of interacting with VocabString fields.

Default Vocabulary Terms

The STIX Language often suggested a default controlled vocabulary type for a given controlled vocabulary field. Each controlled vocabulary contains an enumeration of allowed terms.

Each `VocabString` implementation found in the `stix.common.vocabs` module contains static class-level attributes for each vocabulary term. When setting controlled vocabulary field values, it is recommended that users take advantage of these class-level attributes.

The following demonstrates setting the `Package_Intent` field with a default vocabulary term. Note that the `STIXHeader.package_intents` property returns a list. As such, we use the `append()` method to add terms. Other STIX controlled vocabulary fields may only allow one value rather than a list of values.

```python
from stix.core import STIXHeader
from stix.common.vocabs import PackageIntent

header = STIXHeader()
header.package_intents.append(PackageIntent.TERM_INDICATORS)

print(header.to_xml())
```

Which outputs:

```xml
<stix:STIXHeaderType>
  <stix:Package_Intent xsi:type="stixVocabs:PackageIntentVocab-1.0">Indicators</stix:Package_Intent>
</stix:STIXHeaderType>
```
Non-Default Vocabulary Terms

Though it is suggested, STIX content authors are not required to use the default controlled vocabulary for a given field. As such, python-stix allows users to pass in non-default values for controlled vocabulary fields.

To set a controlled vocabulary to a non-default vocabulary term, pass a `VocabString` instance into a controlled vocabulary field.

A raw `VocabString` field will contain no `xsi:type` information or `_ALLOWED_VALUES` members, which removes the input and schema validation requirements.

```python
from stix.core import STIXHeader
from stix.common.vocabs import VocabString, PackageIntent

header = STIXHeader()
non_default_term = VocabString("NON-DEFAULT VOCABULARY TERM")
header.package_intents.append(non_default_term)

print(header.to_xml())
```

Which outputs:

```
<stix:STIXHeaderType>
  <stix:Package_Intent>NON-DEFAULT VOCABULARY TERM</stix:Package_Intent>
</stix:STIXHeaderType>
```

Notice that the `<stix:Package_Intent>` field does not have an `xsi:type` attribute. As such, this field can contain any string value and is not bound by a controlled vocabulary enumeration of terms.

Working With Custom Controlled Vocabularies

STIX allows content authors and developers to extend the `ControlledVocabularyStringType` schema type for the definition of new controlled vocabularies. The python-stix library allows developers to create and register Python types which mirror the custom XML Schema vocabulary types.

XSD Example  The following XML Schema example shows the definition of a a new custom controlled vocabulary schema type. Instances of this schema type could be used wherever a `ControlledVocabularyStringType` instance is expected (e.g., the STIX_Header/Packet_Intent field).

```xml
<xs:schema
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:customVocabs="http://customvocabs.com/vocabs-1"
    xmlns:stixVocabs="http://stix.mitre.org/default_vocabularies-1"
    xmlns:stixCommon="http://stix.mitre.org/common-1"
    targetNamespace="http://customvocabs.com/vocabs-1"
    elementFormDefault="qualified"
    version="1.2"
    xml:lang="English">
<xs:import
    namespace="http://stix.mitre.org/common-1"
    schemaLocation="http://stix.mitre.org/XMLSchema/common/1.2/stix_common.xsd"/>
<xs:complexType
    name="CustomVocab-1.0">
  <xs:simpleContent>
    <xs:restriction
        base="stixCommon:ControlledVocabularyStringType">
      <xs:simpleType>
        <xs:union
            memberTypes="customVocabs:CustomEnum-1.0"/>
      </xs:simpleType>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
```

Filename: customVocabs.xsd
**XML Instance Sample**  The following STIX XML instance document shows a potential use of this field. Note the xsi:type=customVocabs:CustomVocab-1.0 on the Package_Intent field.

Filename: customVocabs.xml

```xml
<stix:STIX_Package
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:stixExample="http://stix.mitre.org/example"
    xmlns:stix="http://stix.mitre.org/stix-1"
    xmlns:customVocabs="http://customvocabs.com/vocabs-1"
    xsi:schemaLocation="
        http://stix.mitre.org/stix-1 /path/to/stix_core.xsd
        http://customvocabs.com/vocabs-1 /path/to/customVocabs.xsd"
    id="stixExample:STIXPackage-33fe3b22-0201-47cf-85d0-97c02164528d"
    version="1.2">
    <stix:STIX_Header>
        <stix:Package_Intent xsi:type="customVocabs:CustomVocab-1.0">FOO</stix:Package_Intent>
    </stix:STIX_Header>
</stix:STIX_Package>
```

**Python Code**  To parse content which uses custom controlled vocabularies, Python developers don’t have to do anything special—just call `STIXPackage.from_xml()` on the input and all the namespaces, xsi:types, etc. are attached to each instance of `VocabString`. When serializing the document, the input namespaces and xsi:type attributes are retained!

However, to create new content which utilizes a schema defined and enforced custom controlled vocabulary, developers must create a `VocabString` implementation which mirrors the schema definition.

For our CustomVocab-1.0 schema type, the Python would look like this:

```python
from stix.common import vocabs

# Create a custom vocabulary type
@vocabs.register_vocab
class CustomVocab(vocabs.VocabString):
    _namespace = 'http://customvocabs.com/vocabs-1'
    _XSI_TYPE = 'customVocabs:CustomVocab-1.0'
    # Valid terms
    TERM_FOO = 'FOO'
    TERM_BAR = 'BAR'
```

As you can see, we can express a lot of the same information found in the XML Schema definition, but in Python!
• _namespace: The targetNamespace for our custom vocabulary
• _XSI_TYPE: The xsi:type attribute value to write out for instances of this vocabulary.
• TERM_FOO|BAR: Allowable terms for the vocabulary. These terms are collected for input validation.

**Note:** The @register_vocab class decorator registers the class and its xsi:type as a VocabString implementation so python-stix will know to build instances of CustomVocab when parsed content contains CustomVocab-1.0 content.

This also inspects the class attributes for any that begin with TERM_ and collects their values for the purpose of input validation.

**Warning:** Before python-stix 1.2.0.0, users registered custom VocabString implementations via the stix.common.vocabs.add_vocab() method. This method still exists but is considered DEPRECATED in favor of the stix.common.vocabs.register_vocab() class decorator.

```python
# builtin
from StringIO import StringIO

# python-stix modules
from stix.core import STIXPackage
from stix.common.vocabs import VocabString, register_vocab
from mixbox.namespaces import register_namespace, Namespace

XML = ""
""
<stix:STIX_Package
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:stix="http://stix.mitre.org/stix-1"
    xmlns:customVocabs="http://customvocabs.com/vocabs-1"
    xmlns:example="http://example.com/"
    xsi:schemaLocation="http://stix.mitre.org/stix-1 /path/to/stix_core.xsd"
    http://customvocabs.com/vocabs-1 /path/to/customVocabs.xsd"
    id="example:STIXPackage-33fe3b22-0201-47cf-85d0-97c02164528d"
    version="1.2">
    <stix:STIX_Header>
        <stix:Package_Intent xsi:type="customVocabs:CustomVocab-1.0">FOO</stix:Package_Intent>
    </stix:STIX_Header>
</stix:STIX_Package>
"""

# Create a VocabString class for our CustomVocab-1.0 vocabulary which
@register_vocab
class CustomVocab(VocabString):
    _namespace = 'http://customvocabs.com/vocabs-1'
    _XSI_TYPE = 'customVocabs:CustomVocab-1.0'
    TERM_FOO = 'FOO'
    TERM_BAR = 'BAR'

    register_namespace(Namespace(CustomVocab._namespace, "customVocabNS"))

# Parse the input document
sio = StringIO(XML)
package = STIXPackage.from_xml(sio)

# Retrieve the first (and only) Package_Intent entry
```
# Print information about the input Package_Intent
print('%s %s %s' % (type(package_intent), package_intent.xsi_type, package_intent))

# Add another Package Intent
bar = CustomVocab('BAR')
package.stix_header.add_package_intent(bar)

# This will include the 'BAR' CustomVocab entry
print(package.to_xml())

Version: 1.2.0.8

2.4 Examples

This page includes some basic examples of creating and parsing STIX content.

There are a couple things we do in these examples for purposes of demonstration that shouldn’t be done in production code:

- In some examples, we use `set_id_method(IDGenerator.METHOD_INT)` to make IDs for STIX constructs easier to read and cross-reference within the XML document. In production code, you should omit this statement, which causes random UUIDs to be created instead, or create explicit IDs yourself for STIX constructs.

See the STIX Idioms documentation for more great examples of how to use python-stix.

2.4.1 Creating a STIX Package

```python
from stix.core import STIXPackage
from stix.report import Report
from stix.report.header import Header
from stix.utils import IDGenerator, set_id_method

set_id_method(IDGenerator.METHOD_INT) # For testing and demonstration only!

stix_package = STIXPackage()
stix_report = Report()
stix_report.header = Header()
stix_report.header.description = "Getting Started!"
stix_package.add(stix_report)

print(stix_package.to_xml())
```

Which outputs:

```xml
<stix:STIX_Package
    xmlns:cybox="http://cybox.mitre.org/cybox-2"
    xmlns:cyboxCommon="http://cybox.mitre.org/common-2"
    xmlns:cyboxVocabs="http://cybox.mitre.org/default_vocabularies-2"
    xmlns:example="http://example.com"
    xmlns:report="http://stix.mitre.org/Report-1"
    xmlns:stix="http://stix.mitre.org/stix-1"
    xmlns:stixCommon="http://stix.mitre.org/common-1"
    xmlns:stixVocabs="http://stix.mitre.org/default_vocabularies-1"
```
2.4.2 Controlled Vocabularies: VocabString

This section has moved! Head over to Controlled Vocabularies for the documentation.

2.4.3 ID Namespaces

This section has moved! Head over to ID Namespaces for the documentation.

Version: 1.2.0.8

2.5 APIs or bindings?

This page describes both the APIs and the bindings provided by the python-stix library.

2.5.1 Overview

The python-stix library provides APIs and utilities that aid in the creation, consumption, and processing of Structured Threat Information eXpression (STIX) content. The APIs that drive much of the functionality of python-stix sit on top of a binding layer that acts as a direct connection between Python and the STIX XML. Because both the APIs and the bindings allow for the creation and development of STIX content, developers that are new to python-stix may not understand the differences between the two. This document aims to identify the purpose and uses of the APIs and bindings.

2.5.2 Bindings

The python-stix library leverages machine generated XML-to-Python bindings for the creation and processing of STIX content. These bindings are created using the generateDS utility and can be found under stix.bindings within the package hierarchy.

The STIX bindings allow for a direct, complete mapping between Python classes and STIX XML Schema data structures. That being said, it is possible (though not advised) to use only the STIX bindings to create STIX documents. However, because the code is generated from XML Schema without contextual knowledge of relationships or broader organizational/developmental schemes, it is often a cumbersome and laborious task to create even the simplest of STIX documents.

Developers within the python-stix team felt that the binding code did not lend itself to rapid development or natural navigation of data, and so it was decided that a higher-level API should be created.
2.5.3 APIs

The python-stix APIs are classes and utilities that leverage the STIX bindings for the creation and processing of STIX content. The APIs are designed to behave more naturally when working with STIX content, allowing developers to conceptualize and interact with STIX documents as pure Python objects and not XML Schema objects.

The APIs provide validation of inputs, multiple input and output formats, more Pythonic access of data structure internals and interaction with classes, and better interpretation of a developers intent through datatype coercion and implicit instantiation.

Note: The python-stix APIs are under constant development. Our goal is to provide full API coverage of the STIX data structures, but not all structures are exposed via the APIs yet. Please refer to the API Reference for API coverage details.

2.5.4 Brevity Wins

The two code examples show the difference in creating and printing a simple STIX document consisting of only a STIX Package and a STIX Header with a description and produced time using the python-stix and python-cybox bindings. Both examples will produce the same STIX XML!

API Example

```python
from datetime import datetime
from stix.core import STIXPackage, STIXHeader
from stix.common import InformationSource
from cybox.common import Time

# Create the STIX Package and STIX Header objects
stix_package = STIXPackage()
stix_header = STIXHeader()

# Set the description
stix_header.description = 'APIs vs. Bindings Wiki Example'

# Set the produced time to now
stix_header.information_source = InformationSource()
stix_header.information_source.time = Time()
stix_header.information_source.time.produced_time = datetime.now()

# Build document
stix_package.stix_header = stix_header

# Print the document to stdout
print(stix_package.to_xml())
```

Binding Example

```python
import sys
from datetime import datetime

import stix.bindings.stix_core as stix_core_binding
import stix.bindings.stix_common as stix_common_binding
import cybox.bindings.cybox_common as cybox_common_binding

# Create the STIX Package and STIX Header objects
stix_package = stix_core_binding.STIXType()
```

2.5. APIs or bindings?
stix_header = stix_core_binding.STIXHeaderType()

# Set the description
stix_header_description = stix_common_binding.StructuredTextType()
stix_header_description.set_valueOf_('APIs vs. Bindings Wiki Example')

# Set the produced time to now
stix_header_time = cybox_common_binding.TimeType()
stix_header_time.set_Produced_Time(datetime.now())

# Bind the time to the STIX Header’s Information Source element
stix_header_info_source = stix_common_binding.InformationSourceType()
stix_header_info_source.set_Time(stix_header_time)

# Build the document
stix_header.set_Description(stix_header_description)
stix_header.set_Information_Source(stix_header_info_source)
stix_package.set_STIX_Header(stix_header)

# Print the document to stdout
stix_package.export(sys.stdout, 0, stix_core_binding.DEFAULT_XML_NS_MAP)

2.5.5 Feedback

If there is a problem with the APIs or bindings, or if there is functionality missing from the APIs that forces the use of the bindings, let us know in the python-stix issue tracker
3.1 API Reference

The python-stix APIs are the recommended tools for reading, writing, and manipulating STIX XML documents.

Note: The python-stix APIs are currently under development. As such, API coverage of STIX data constructs is incomplete; please bear with us as we work toward complete coverage. This documentation also serves to outline current API coverage.

3.1.1 STIX

Modules located in the base stix package

Version: 1.2.0.8

**stix.base Module**

**Classes**

class stix.base.Entity

Base class for all classes in the STIX API.

    find(id_)

    Searches the children of an Entity implementation for an object with an id_ property that matches id_.

    to_xml(include_namespaces=True, include_schemalocs=False, ns_dict=None, schemaloc_dict=None, pretty=True, auto_namespace=True, encoding='utf-8')

    Serializes a Entity instance to an XML string.

    The default character encoding is utf-8 and can be set via the encoding parameter. If encoding is None, a string (unicode in Python 2, str in Python 3) is returned.

    Parameters

        • auto_namespace – Automatically discover and export XML namespaces for a STIX Entity instance.
• **include_namespaces** – Export namespace definitions in the output XML. Default is True.

• **include_schemalocs** – Export xsi:schemaLocation attribute in the output document. This will attempt to associate namespaces declared in the STIX document with schema locations. If a namespace cannot be resolved to a schemaLocation, a Python warning will be raised. Schemalocations will only be exported if include_namespaces is also True.

• **ns_dict** – Dictionary of XML definitions (namespace is key, alias is value) to include in the exported document. This must be passed in if auto_namespace is False.

• **schemaloc_dict** – Dictionary of XML namespace: schema location mappings to include in the exported document. These will only be included if auto_namespace is False.

• **pretty** – Pretty-print the XML.

• **encoding** – The output character encoding. Default is utf-8. If encoding is set to None, a string (unicode in Python 2, str in Python 3) is returned.

Returns  An XML string for this Entity instance. Default character encoding is utf-8.

class stix.base.EntityList(*args)
    Bases: mixbox.entities.EntityList, stix.base.Entity

Version: 1.2.0.8

**stix.data_marking** Module

**Classes**

class stix.data_marking.Marking(markings=None)
    Bases: stix.base.EntityList

class stix.data_marking.MarkingSpecification(controlled_structure=None, marking_structures=None)
    Bases: stix.base.Entity

class stix.data_marking.MarkingStructure
    Bases: stix.base.Entity

**Functions**

stix.data_marking.add_extension(cls)
    Registers a stix.Entity class as an implementation of an xml type.

    Classes must have an _XSI_TYPE class attributes to be registered. The value of this attribute must be a valid xsi:type.

    **Note:** This was designed for internal use.

**3.1.2 STIX Campaign**

Modules located in the stix.campaign package
Version: 1.2.0.8

**stix.campaign Module**

**Overview**

The `stix.campaign` module implements *Campaign*. Campaigns are instances of ThreatActors pursuing an intent, as observed through sets of Incidents and/or TTP, potentially across organizations.

**Documentation Resources**

- Campaign Data Model

**Classes**

```python
class stix.campaign.Campaign(id_=None, idref=None, timestamp=None, title=None, description=None, short_description=None):
    Bases: stix.base.BaseCoreComponent

    Implementation of the STIX Campaign.

    Parameters
    ----------
    id (optional) -- An identifier. If None, a value will be generated via mixbox.idgen.create_id(). If set, this will unset the idref property.
    idref (optional) -- An identifier reference. If set this will unset the id_ property.
    timestamp (optional) -- A timestamp value. Can be an instance of datetime.datetime or str.
    description -- A description of the purpose or intent of this object.
    short_description -- A short description of the intent or purpose of this object.
    title -- The title of this object.

    add_activity(value)
    Add an Activity object to the activity collection.

    add_description(description)
    Add a description to the descriptions collection.
    This is the same as calling “foo.descriptions.add(bar)”.

    add_short_description(description)
    Add a description to the short_descriptions collection.
    This is the same as calling “foo.short_descriptions.add(bar)”.

    description
    A single description about the contents or purpose of this object.
    Default Value: None

    Note: If this object has more than one description set, this will return the description with the lowest ordinality value.
```
Returns An instance of `StructuredText`

`find(id_)`
Searches the children of an `Entity` implementation for an object with an `id_` property that matches `id_`.

`short_description`
A single short description about the contents or purpose of this object.
Default Value: None

**Note:** If this object has more than one short description set, this will return the description with the lowest ordinality value.

Returns An instance of `StructuredText`

`to_dict()`
Convert to a `dict`
Subclasses can override this function.

Returns Python dict with keys set from this Entity.

`to_json()`
Export an object as a JSON String.

`to_obj(ns_info=None)`
Convert to a GenerateDS binding object.
Subclasses can override this function.

Returns An instance of this Entity’s `_binding_class` with properties set from this Entity.

class `stix.campaign.AssociatedCampaigns`(scope=\`None\`, *\`args\*)
Bases: `stix.common.related.GenericRelationshipList`

class `stix.campaign.Attribution`(scope=\`None\`, *\`args\*)
Bases: `stix.common.related.GenericRelationshipList`

class `stix.campaign.Names`(*\`args\*)
Bases: `stix.base.EntityList`

class `stix.campaign.RelatedIncidents`(scope=\`None\`, *\`args\*)
Bases: `stix.common.related.GenericRelationshipList`

class `stix.campaign.RelatedIndicators`(scope=\`None\`, *\`args\*)
Bases: `stix.common.related.GenericRelationshipList`

class `stix.campaign.RelatedTTPs`(scope=\`None\`, *\`args\*)
Bases: `stix.common.related.GenericRelationshipList`

### 3.1.3 STIX Common

Modules located in the `stix.common` package

Version: 1.2.0.8
**stix.common Module**

**Classes**

```python
class stix.common.EncodedCDATA (value=None, encoded=None)
    Bases: stix.base.Entity
```

**Version:** 1.2.0.8

---

**stix.common.activity Module**

**Classes**

```python
class stix.common.activity.Activity
    Bases: stix.base.Entity
```

```python
    add_description (description)
        Adds a description to the descriptions collection.
        This is the same as calling “foo.descriptions.add(bar)”.

    description
        A single description about the contents or purpose of this object.
        Default Value: None
```

**Note:** If this object has more than one description set, this will return the description with the lowest ordinality value.

**Returns** An instance of `StructuredText`

---

**stix.common.confidence Module**

**Classes**

```python
class stix.common.confidence.Confidence (value=None, timestamp=None, description=None, source=None)
    Bases: stix.base.Entity
```

```python
    add_description (description)
        Adds a description to the descriptions collection.
        This is the same as calling “foo.descriptions.add(bar)”.

    description
        A single description about the contents or purpose of this object.
        Default Value: None
```

**Note:** If this object has more than one description set, this will return the description with the lowest ordinality value.
Returns  An instance of StructuredText

Version: 1.2.0.8

**stix.common.datetimewithprecision Module**

**Classes**

class  stix.common.datetimewithprecision.DateTimeWithPrecision(value=None, precision='second')

Bases: stix.base.Entity

**Constants**

stix.common.datetimewithprecision.DATE_PRECISION_VALUES = ('year', 'month', 'day')
tuple() -> empty tuple
tuple( iterable) -> tuple initialized from iterable's items
If the argument is a tuple, the return value is the same object.

stix.common.datetimewithprecision.TIME_PRECISION_VALUES = ('hour', 'minute', 'second')
tuple() -> empty tuple
tuple( iterable) -> tuple initialized from iterable's items
If the argument is a tuple, the return value is the same object.

stix.common.datetimewithprecision.DATETIME_PRECISION_VALUES = ('year', 'month', 'day', 'hour', 'minute', 'second')
tuple() -> empty tuple
tuple( iterable) -> tuple initialized from iterable's items
If the argument is a tuple, the return value is the same object.

Version: 1.2.0.8

**stix.common.identity Module**

**Classes**

class  stix.common.identity.Identity(id_=None, idref=None, name=None, related_identities=None)

Bases: stix.base.Entity

class  stix.common.identity.RelatedIdentities(*args)

Bases: stix.base.EntityList

**Functions**

stix.common.identity.add_extension(cls)

Registers a stix.Entity class as an implementation of an xml type.

Classes must have an _XSI_TYPE class attributes to be registered. The value of this attribute must be a valid xsi:type.

**Note:** This was designed for internal use.

Version: 1.2.0.8
stix.common.information_source Module

Classes

class stix.common.information_source.InformationSource (description=None, identity=None, time=None, tools=None, contributing_sources=None, references=None)

Bases: stix.base.Entity

add_description (description)
   Adds a description to the descriptions collection.
   This is the same as calling “foo.descriptions.add(bar)”.

description
   A single description about the contents or purpose of this object.
   Default Value: None

   Note: If this object has more than one description set, this will return the description with the lowest ordinality value.

   Returns: An instance of StructuredText

class stix.common.information_source.ContributingSources (*args)

Bases: stix.base.EntityList

Version: 1.2.0.8

stix.common.kill_chains Module

Classes

class stix.common.kill_chains.KillChain (id_=None, name=None, definer=None, reference=None)

Bases: stix.base.Entity

class stix.common.kill_chains.KillChains (*args)

Bases: stix.base.EntityList

class stix.common.kill_chains.KillChainPhase (phase_id=None, name=None, ordinality=None)

Bases: stix.common.kill_chains.KillChainPhase

class stix.common.kill_chains.KillChainPhaseReference (phase_id=None, name=None, ordinality=None, kill_chain_id=None, kill_chain_name=None)

Bases: stix.common.kill_chains.KillChainPhase

class stix.common.kill_chains.KillChainPhasesReference (*args)

Bases: stix.base.EntityList

3.1. API Reference 23
Lockheed Martin Kill Chain

There is a shortcut for adding kill chain phases from the Lockheed Martin Cyber Kill Chain to indicators:

```python
from stix.common.kill_chains.lmco import PHASE_RECONNAISSANCE
from stix.indicator import Indicator
i = Indicator()
i.add_kil_chain_phase(PHASE_RECONNAISSANCE)
print i.to_xml(include_namespaces=False)
```

```xml
<indicator:Indicator id="example:indicator-2bb1c0ea-7dd8-40fb-af64-7199f00719c1" timestamp="2015-03-17T19:14:22.797675+00:00" xsi:type='indicator:IndicatorType'>
  <indicator:Kill_Chain_Phases>
    <stixCommon:Kill_Chain_Phase phase_id="stix:TTP-af1016d6-a744-4ed7-ac91-00fe2272185a"/>
  </indicator:Kill_Chain_Phases>
</indicator:Indicator>
```

Version: 1.2.0.8

**stix.common.related Module**

**Classes**

```python
class stix.common.related.GenericRelationship(confidence=None, 
  information_source=None, relationship=None)
Bases: stix.base.Entity

class stix.common.related.GenericRelationshipList(scope=None, *args)
Bases: stix.base.EntityList
Base class for concrete GenericRelationshipList types.

**Note:** Subclasses must supply exactly one multiple TypedField.
```

```python
class stix.common.related.RelatedPackageRef(idref=None, 
  timestamp=None, confidence=None, information_source=None, 
  relationship=None)
Bases: stix.common.related.GenericRelationship

class stix.common.related.RelatedPackageRefs(*args)
Bases: stix.base.EntityList

class stix.common.related._BaseRelated(item=None, 
  confidence=None, information_source=None, relationship=None)
Bases: stix.common.related.GenericRelationship
A base class for related types.
This class is not a real STIX type and should not be directly instantiated.

**Note:** Subclasses must supply a TypedField named item!
```

```python
class stix.common.related.RelatedCampaign(item=None, 
  confidence=None, information_source=None, relationship=None)
Bases: stix.common.related._BaseRelated
```
class stix.common.related.RelatedCOA(item=None, confidence=None, information_source=None, relationship=None)
    Bases: stix.common.related._BaseRelated

class stix.common.related.RelatedExploitTarget(item=None, confidence=None, information_source=None, relationship=None)
    Bases: stix.common.related._BaseRelated

class stix.common.related.RelatedIdentity(item=None, confidence=None, information_source=None, relationship=None)
    Bases: stix.common.related._BaseRelated

class stix.common.related.RelatedIncident(item=None, confidence=None, information_source=None, relationship=None)
    Bases: stix.common.related._BaseRelated

class stix.common.related.RelatedIndicator(item=None, confidence=None, information_source=None, relationship=None)
    Bases: stix.common.related._BaseRelated

class stix.common.related.RelatedObservable(item=None, confidence=None, information_source=None, relationship=None)
    Bases: stix.common.related._BaseRelated

class stix.common.related.RelatedThreatActor(item=None, confidence=None, information_source=None, relationship=None)
    Bases: stix.common.related._BaseRelated

class stix.common.related.RelatedTTP(item=None, confidence=None, information_source=None, relationship=None)
    Bases: stix.common.related._BaseRelated

class stix.common.related.RelatedReports(scope=None, *args)
    Bases: stix.common.related.GenericRelationshipList

class stix.common.related.RelatedReport(item=None, confidence=None, information_source=None, relationship=None)
    Bases: stix.common.related._BaseRelated

Version: 1.2.0.8

stix.common.statement Module

Classes

class stix.common.statement.Statement(value=None, timestamp=None, description=None, source=None)
    Bases: stix.base.Entity

    add_description(description)
        Adds a description to the descriptions collection.
        This is the same as calling “foo.descriptions.add(bar)”.

    description
        A single description about the contents or purpose of this object.

        Default Value: None
Note: If this object has more than one description set, this will return the description with the lowest ordinality value.

Returns An instance of StructuredText

Version: 1.2.0.8

stix.common.structured_text Module

Classes

class stix.common.structured_text.StructuredText(value=None, ordinality=None)
    Bases: stix.base.Entity
    Used for storing descriptive text elements.

    id
      An id for the text element, typically used for controlled structure xpath selectors.

    value
      The text value of this object.

    structuring_format
      The format of the text. For example, html5.

    __str__()
      Returns a UTF-8 encoded string representation of the value.

    __unicode__()
      Returns a unicode string representation of the value.

    to_dict()
      Converts this object into a dictionary representation.

    Note: If no properties or attributes are set other than value, this will return a string.

to_obj(ns_info=None)
    Convert to a GenerateDS binding object.
    Subclasses can override this function.

    Returns An instance of this Entity’s _binding_class with properties set from this Entity.

class stix.common.structured_text.StructuredTextList(*args)
    Bases: stix.base.TypedCollection, _abcoll.Sequence
    A sequence type used to store StructureText objects.

    Parameters *args – A variable-length argument list which can contain single StructuredText objects or sequences of objects.

    __delitem__(key)
      Removes the item with a given ordinality.

    Parameters key – An ordinality value.

    Raises KeyError – If the key does not match the ordinality for any object in the collection.
__getitem__(key)
    Returns the StructuredText object with a matching ordinality.

    Parameters key – An ordinality value.

    Raises KeyError – If key does not match the ordinality of any StructuredText object.

__iter__()
    Returns an iterator for the collection sorted by ordinality.

add(value)
    Adds the StructuredText value to the collection.

    If value is not a StructuredText object, an attempt will be made to convert it to one.

Note: If value does not have an ordinality set, one will be assigned. If value has an ordinality which matches one already in the collection, value will replace the existing item.

    Parameters value – A StructuredText object.

insert(value)
    Inserts value into the collection.

    If value has an ordinality which conflicts with an existing value, the existing value (and any contiguous values) will have their ordinality values incremented by one.

next_ordinality
    Returns the “+1” of the highest ordinality in the collection.

remove(value)
    Removes the value from the collection.

reset()
    Assigns sequential ordinality values to each of the sorted StructuredText objects, starting with 1 and ending at len(self).

sorted
    Returns a copy of the collection of internal StructuredText objects, sorted by their ordinality.

to_dict()
    Returns a list of dictionary representations of the contained objects.

    An attempt is made to flatten out the returned list when there is only one item in the collection. This is to support backwards compatibility with previous versions of python-stix.

    •If the list repr has more than one item, return the list.

    •If there is only one item, inspect it.

        –If the item is not a dictionary, return it.

        –If its ordinality key has a corresponding value of 1, remove it from the dictionary since it’s assumed if there is only one item.

        –After removing ordinality, if the only key left is value, just return the value of value (a string).

    to_obj(ns_info=None)
    Returns a binding object list for the StructuredTextList.

    If the list has a length of 1, and its member has an ordinality of 1, the ordinality will be unset.
**update(iterable)**

Adds each item of iterable to the collection.

**Parameters**

**iterable** – An iterable collection of *StructuredText* objects to add to this collection.

**Note:** Any existing objects with conflicting ordinality values will be overwritten.

Version: 1.2.0.8

---

**stix.common.tools Module**

**Classes**

**class stix.common.tools.ToolInformation(title=None, short_description=None, tool_name=None, tool_vendor=None)**

   **Bases:** stix.base.Entity, cybox.common.tools.ToolInformation

   **add_short_description(description)**

   Adds a description to the short_descriptions collection.
   
   This is the same as calling “foo.short_descriptions.add(bar)”.

   **short_description**

   A single short description about the contents or purpose of this object.
   
   Default Value: None

   **Note:** If this object has more than one short description set, this will return the short description with the lowest ordinality value.

   **Returns** An instance of *StructuredText*

Version: 1.2.0.8

---

**stix.common.vocabs Module**

**Classes**

**class stix.common.vocabs.Asset_type_1_0(value=None)**

   **Bases:** stix.common.vocabs.VocabString

   **TERM_ACCESS_READER** = ‘Access reader’

   **TERM_ADMINISTRATOR** = ‘Administrator’

   **TERM_ATM** = ‘ATM’

   **TERM_AUDITOR** = ‘Auditor’

   **TERM_AUTH_TOKEN** = ‘Auth token’

   **TERM_BACKUP** = ‘Backup’

   **TERM_BROADBAND** = ‘Broadband’
TERM_CALL_CENTER = ‘Call center’
TERM_CAMERA = ‘Camera’
TERM_CASHIER = ‘Cashier’
TERM_CUSTOMER = ‘Customer’
TERM_DATABASE = ‘Database’
TERM_DCS = ‘DCS’
TERM_DESKTOP = ‘Desktop’
TERM_DEVELOPER = ‘Developer’
TERM_DHCP = ‘DHCP’
TERM_DIRECTORY = ‘Directory’
TERM_DISK_DRIVE = ‘Disk drive’
TERM_DISK_MEDIA = ‘Disk media’
TERM_DNS = ‘DNS’
TERM/Documents = ‘Documents’
TERM_ENDUSER = ‘End-user’
TERM_EXECUTIVE = ‘Executive’
TERM_FILE = ‘File’
TERM_FINANCE = ‘Finance’
TERM_FIREWALL = ‘Firewall’
TERM_FLASH_DRIVE = ‘Flash drive’
TERM_FORMER_EMPLOYEE = ‘Former employee’
TERM_GAS_TERMINAL = ‘Gas terminal’
TERM_GUARD = ‘Guard’
TERM_HELPDESK = ‘Helpdesk’
TERM_HSM = ‘HSM’
TERM_HUMAN_RESOURCES = ‘Human resources’
TERM_IDS = ‘IDS’
TERM_KIOSK = ‘Kiosk’
TERM_LAN = ‘LAN’
TERM_LAPTOP = ‘Laptop’
TERM_LOG = ‘Log’
TERM_MAIL = ‘Mail’
TERM_MAINFRAME = ‘Mainframe’
TERM_MAINTENANCE = ‘Maintenance’
TERM_MANAGER = ‘Manager’
TERM_MEDIA = ‘Media’
TERM_MOBILE_PHONE = 'Mobile phone'
TERM_NETWORK = 'Network'
TERM_PARTNER = 'Partner'
TERM_PAYMENT_CARD = 'Payment card'
TERM_PAYMENT_SWITCH = 'Payment switch'
TERM_PBX = 'PBX'
TERM_PED_PAD = 'PED pad'
TERM_PERIPHERAL = 'Peripheral'
TERM_PERSON = 'Person'
TERM_PLC = 'PLC'
TERM_POS_CONTROLLER = 'POS controller'
TERM_POS_TERMINAL = 'POS terminal'
TERM_PRINT = 'Print'
TERM_PRIVATE_WAN = 'Private WAN'
TERM_PROXY = 'Proxy'
TERM_PUBLIC_WAN = 'Public WAN'
TERM_REMOTE_ACCESS = 'Remote access'
TERM_ROUTER_OR_SWITCH = 'Router or switch'
TERM_RTU = 'RTU'
TERM_SAN = 'SAN'
TERM_SCADA = 'SCADA'
TERM_SERVER = 'Server'
TERM_SMART_CARD = 'Smart card'
TERM_TABLET = 'Tablet'
TERM_TAPES = 'Tapes'
TERM_TELEPHONE = 'Telephone'
TERM UNKNOWN = 'Unknown'
TERM_USER DEVICE = 'User Device'
TERM_VOIP_ADAPTER = 'VoIP adapter'
TERM_VOIP_PHONE = 'VoIP phone'
TERM_WEB_APPLICATION = 'Web application'
TERM_WLAN = 'WLAN'

class stix.common.vocabs.AttackerInfrastructureType_1_0(value=None)
Bases: stix.common.vocabs.VocabString
TERM ANONYMIZATION = 'Anonymization'
TERM ANONYMIZATION_PROXY = 'Anonymization - Proxy'
TERM_ANONYMIZATION_TOR_NETWORK = 'Anonymization - TOR Network'
TERM_ANONYMIZATION_VPN = 'Anonymization - VPN'
TERM_COMMUNICATIONS = 'Communications'
TERM_COMMUNICATIONS_BLOGS = 'Communications - Blogs'
TERM_COMMUNICATIONS_FORUMS = 'Communications - Forums'
TERM_COMMUNICATIONSINTERNETRELAYCHAT = 'Communications - Internet Relay Chat'
TERM_COMMUNICATIONSMICROBLOGS = 'Communications - Micro-Blogs'
TERM_COMMUNICATIONSMOBILECOMMUNICATIONS = 'Communications - Mobile Communications'
TERM_COMMUNICATIONSSOCIALNETWORKS = 'Communications - Social Networks'
TERM_COMMUNICATIONSUSERGENERATEDCONTENTWEBSITES = 'Communications - User-Generated Content Websites'
TERM_DOMAIN_REGISTRATION = 'Domain Registration'
TERM_DOMAIN_REGISTRATIONDYNAMICDNSSERVICES = 'Domain Registration - Dynamic DNS Services'
TERM_DOMAIN_REGISTRATIONLEGITIMATEDOMAINREGISTRATIONSERVICES = 'Domain Registration - Legitimate Domain Registration Services'
TERM_DOMAIN_REGISTRATIONMALICIOUSDOMAINREGISTRARS = 'Domain Registration - Malicious Domain Registrars'
TERM_DOMAIN_REGISTRATIONTOPLEVELDOMAINREGISTRARS = 'Domain Registration - Top-Level Domain Registrars'
TERM_ELECTRONIC_PAYMENT_METHODS = 'Electronic Payment Methods'
TERM_HOSTING = 'Hosting'
TERM_HOSTING_BULLETPROOF_OR_ROGUE_HOSTING = 'Hosting - Bulletproof / Rogue Hosting'
TERM_HOSTING_CLOUD_HOSTING = 'Hosting - Cloud Hosting'
TERM_HOSTING_COMPROMISED_SERVER = 'Hosting - Compromised Server'
TERM_HOSTING_FAST_FLUX_BOTNET_HOSTING = 'Hosting - Fast Flux Botnet Hosting'
TERM_HOSTING_LEGITIMATE_HOSTING = 'Hosting - Legitimate Hosting'
class stix.common.vocabs.AttackerToolType_1_0 (value=None)
Bases: cybox.common.vocabs.VocabString
TERM_APPLICATION_SCANNER = 'Application Scanner'
TERM_MALWARE = 'Malware'
TERM_PASSWORD_CRACKING = 'Password Cracking'
TERM_PENETRATION_TESTING = 'Penetration Testing'
TERM_PORT_SCANNER = 'Port Scanner'
TERM_TRAFFIC_SCANNER = 'Traffic Scanner'
TERM_VULNERABILITY_SCANNER = 'Vulnerability Scanner'
class stix.common.vocabs.AvailabilityLossType_1_0 (value=None)
Bases: stix.common.vocabs.VocabString
TERM_ACCELERATION = 'Acceleration'
TERM_DEGREDATION = 'Degredation'
TERM_DESTRUCTION = 'Destruction'
TERM INTERRUPTION = 'Interruption'
class `stix.common.vocabs.AvailabilityLossType_1_1_1` (value=None)
Bases: `stix.common.vocabs.VocabString`

TERM_ACCELERATION = ‘Acceleration’
TERM_DEGRADATION = ‘Degradation’
TERM_DESTRUCTION = ‘Destruction’
TERM INTERRUPTION = ‘Interruption’
TERM LOSS = ‘Loss’
TERM OBSCURATION = ‘Obscuration’
TERM UNKNOWN = ‘Unknown’

class `stix.common.vocabs.COASTage_1_0` (value=None)
Bases: `stix.common.vocabs.VocabString`

TERM REMEDY = ‘Remedy’
TERM RESPONSE = ‘Response’

class `stix.common.vocabs.CampaignStatus_1_0` (value=None)
Bases: `stix.common.vocabs.VocabString`

TERM FUTURE = ‘Future’
TERM HISTORIC = ‘Historic’
TERM ONGOING = ‘Ongoing’

class `stix.common.vocabs.CourseOfActionType_1_0` (value=None)
Bases: `stix.common.vocabs.VocabString`

TERM DIPLOMATIC ACTIONS = ‘Diplomatic Actions’
TERM ERADICATION = ‘Eradication’
TERM HARDENING = ‘Hardening’
TERM INTERNAL_BLOCKING = ‘Internal Blocking’
TERM LOGICAL_ACCESS_RESTRICTIONS = ‘Logical Access Restrictions’
TERM MONITORING = ‘Monitoring’
TERM OTHER = ‘Other’
TERM PATCHING = ‘Patching’
TERM PERIMETER_BLOCKING = ‘Perimeter Blocking’
TERM PHYSICAL_ACCESS_RESTRICTIONS = ‘Physical Access Restrictions’
TERM POLICY_ACTIONS = ‘Policy Actions’
TERM PUBLIC_DISCLOSURE = ‘Public Disclosure’
TERM REBUILDING = ‘Rebuilding’
TERM REDIRECTION = ‘Redirection’
TERM_REDIRECTION_HONEY_POT = ‘Redirection (Honey Pot)’
TERM_TRAINING = ‘Training’

class stix.common.vocabs.DiscoveryMethod_1_0 (value=None)
Bases: stix.common.vocabs.VocabString
TERM_AGENT_DISCLOSURE = ‘Agent Disclosure’
TERM_ANTIVIRUS = ‘Antivirus’
TERM_AUDIT = ‘Audit’
TERM_CUSTOMER = ‘Customer’
TERM_FINANCIAL_AUDIT = ‘Financial Audit’
TERM_FRAUD_DETECTION = ‘Fraud Detection’
TERM_HIPS = ‘HIPS’
TERM_INCIDENT_RESPONSE = ‘Incident Response’
TERM_IT_AUDIT = ‘IT Audit’
TERM_LAW_ENFORCEMENT = ‘Law Enforcement’
TERM_LOG_REVIEW = ‘Log Review’
TERM_MONITORING_SERVICE = ‘Monitoring Service’
TERM_NIDS = ‘NIDS’
TERM_SECURITY_ALARM = ‘Security Alarm’
TERM_UNKNOWN = ‘Unknown’
TERM_UNRELATED_PARTY = ‘Unrelated Party’
TERM_USER = ‘User’

class stix.common.vocabs.DiscoveryMethod_2_0 (value=None)
Bases: stix.common.vocabs.VocabString
TERM_AGENT_DISCLOSURE = ‘Agent Disclosure’
TERM_ANTIVIRUS = ‘Antivirus’
TERM_AUDIT = ‘Audit’
TERM_CUSTOMER = ‘Customer’
TERM_EXTERNAL_FRAUD_DETECTION = ‘External - Fraud Detection’
TERM_FINANCIAL_AUDIT = ‘Financial Audit’
TERM_INTERNAL_FRAUD_DETECTION = ‘Internal - Fraud Detection’
TERM_HIPS = ‘HIPS’
TERM_INCIDENT_RESPONSE = ‘Incident Response’
TERM_INTERNAL_FRAUD_DETECTION = ‘Internal - Fraud Detection’
TERM_IT_AUDIT = ‘IT Audit’
TERM_LAW_ENFORCEMENT = ‘Law Enforcement’
TERM_LOG_REVIEW = ‘Log Review’
TERM_MONITORING_SERVICE = ‘Monitoring Service’
TERM_NIDS = ‘NIDS’
TERM_SECURITY_ALARM = 'Security Alarm'
TERM_UNKNOWN = 'Unknown'
TERM_UNRELATED_PARTY = 'Unrelated Party'
TERM_USER = 'User'
class stix.common.vocabs.HighMediumLow_1_0 (value=None):
    Bases: stix.common.vocabs.VocabString
    TERM_HIGH = 'High'
    TERM_LOW = 'Low'
    TERM_MEDIUM = 'Medium'
    TERM_NONE = 'None'
    TERM_UNKNOWN = 'Unknown'
class stix.common.vocabs.ImpactQualification_1_0 (value=None):
    Bases: stix.common.vocabs.VocabString
    TERM_CATASTROPHIC = 'Catastrophic'
    TERM_DAMAGING = 'Damaging'
    TERM_DISTRACTING = 'Distracting'
    TERM_INSIGNIFICANT = 'Insignificant'
    TERM_PAINFUL = 'Painful'
    TERM_UNKNOWN = 'Unknown'
class stix.common.vocabs.ImpactRating_1_0 (value=None):
    Bases: stix.common.vocabs.VocabString
    TERM_MAJOR = 'Major'
    TERM_MINOR = 'Minor'
    TERM_MODERATE = 'Moderate'
    TERM_NONE = 'None'
    TERM_UNKNOWN = 'Unknown'
class stix.common.vocabs.IncidentCategory_1_0 (value=None):
    Bases: stix.common.vocabs.VocabString
    TERM_DENIAL_OF_SERVICE = 'Denial of Service'
    TERM_EXERCISEORNETWORK_DEFENSE_TESTING = 'Exercise/Network Defense Testing'
    TERM_IMPROPER_USAGE = 'Improper Usage'
    TERM_INVESTIGATION = 'Investigation'
    TERM_MALICIOUS_CODE = 'Malicious Code'
    TERM_SCANSORPROBESORATTEMPTED_ACCESS = 'Scans/Probes/Attempted Access'
    TERM_UNAUTHORIZED_ACCESS = 'Unauthorized Access'
class stix.common.vocabs.IncidentEffect_1_0 (value=None):
    Bases: stix.common.vocabs.VocabString
    TERM_BRAND_OR_IMAGE_DEGRADATION = 'Brand or Image Degradation'
TERM_DATA_BREACH_OR_COMPROMISE = 'Data Breach or Compromise'
TERM_DEGRADATION_OF_SERVICE = 'Degradation of Service'
TERM_DESTRUCTION = 'Destruction'
TERM_DISRUPTION_OF_SERVICE_OR_OPERATIONS = 'Disruption of Service / Operations'
TERM_FINANCIAL_LOSS = 'Financial Loss'
TERM_LOSS_OF_COMPETITIVE_advantage = 'Loss of Competitive Advantage'
TERM_LOSS_OF_COMPETITIVE_advantage_economic = 'Loss of Competitive Advantage - Economic'
TERM_LOSS_OF_COMPETITIVE_advantageMilitary = 'Loss of Competitive Advantage - Military'
TERM_LOSS_OF_COMPETITIVE_advantage_political = 'Loss of Competitive Advantage - Political'
TERM_LOSS_OF_CONFIDENTIAL_OR_PROPRIETARY_INFORMATION_OR_INTELLECTUAL_PROPERTY = 'Loss of Confidential / Proprietary Information or Intellectual Property'
TERM_REGULATORY_COMPLIANCE_OR_LEGAL_IMPACT = 'Regulatory, Compliance or Legal Impact'
TERM_UNINTENDED_ACCESS = 'Unintended Access'
TERM_USER_DATA_LOSS = 'User Data Loss'

class stix.common.vocabs.IncidentStatus_1_0 (value=None)
Bases: stix.common.vocabs.VocabString
TERM_CLOSED = 'Closed'
TERM_CONTAINMENT_ACHIEVED = 'Containment Achieved'
TERM_DELETED = 'Deleted'
TERM_INCIDENT_REPORTED = 'Incident Reported'
TERM_NEW = 'New'
TERM_OPEN = 'Open'
TERM_REJECTED = 'Rejected'
TERM_RESTORED_ACHIEVED = 'Restoration Achieved'
TERM_STALLED = 'Stalled'

class stix.common.vocabs.IndicatorType_1_0 (value=None)
Bases: stix.common.vocabs.VocabString
TERM_ANONYMIZATION = 'Anonymization'
TERM_C2 = 'C2'
TERM_DOMAIN_WATCHLIST = 'Domain Watchlist'
TERM_EXFILTRATION = 'Exfiltration'
TERM_FILE_HASH_WATCHLIST = 'File Hash Watchlist'
TERM_HOST_CHARACTERISTICS = 'Host Characteristics'
TERM_IP_WATCHLIST = 'IP Watchlist'
TERM_MALICIOUS_EMAIL = 'Malicious E-mail'
TERM_MALWARE_ARTIFACTS = 'Malware Artifacts'
TERM_URL_WATCHLIST = 'URL Watchlist'
class stix.common.vocabs.IndicatorType_1_1(value=None)
  Bases: stix.common.vocabs.VocabString
  TERM_ANONYMIZATION = 'Anonymization'
  TERM_C2 = 'C2'
  TERM_COMPROMISED_PKI_CERTIFICATE = 'Compromised PKI Certificate'
  TERM_DOMAIN_WATCHLIST = 'Domain Watchlist'
  TERM_EXFILTRATION = 'Exfiltration'
  TERM_FILE_HASH_WATCHLIST = 'File Hash Watchlist'
  TERM_HOST_CHARACTERISTICS = 'Host Characteristics'
  TERM IMEI_WATCHLIST = 'IMEI Watchlist'
  TERM_IMSI_WATCHLIST = 'IMSI Watchlist'
  TERM_IP_WATCHLIST = 'IP Watchlist'
  TERM_LOGIN_NAME = 'Login Name'
  TERM_MALICIOUS_EMAIL = 'Malicious Email'
  TERM_MALWARE_ARTIFACTS = 'Malware Artifacts'
  TERM_URL_WATCHLIST = 'URL Watchlist'

class stix.common.vocabs.InformationSourceRole_1_0(value=None)
  Bases: stix.common.vocabs.VocabString
  TERM_AGGREGATOR = 'Aggregator'
  TERM_CONTENT_ENHANCERORREFINER = 'Content Enhancer/Refiner'
  TERM_INITIAL_AUTHOR = 'Initial Author'
  TERM_TRANSFORMERORTRANSLATOR = 'Transformer/Translator'

class stix.common.vocabs.InformationType_1_0(value=None)
  Bases: stix.common.vocabs.VocabString
  TERM_AUTHENTICATION_COOKIES = 'Authentication Cookies'
  TERM_INFORMATION_ASSETS = 'Information Assets'
  TERM_INFORMATION_ASSETS_CORPORATE_EMPLOYEE_INFORMATION = 'Information Assets - Corporate Employee Information'
  TERM_INFORMATION_ASSETS_CUSTOMER_PII = 'Information Assets - Customer PII'
  TERM_INFORMATION_ASSETS_EMAIL_LISTS_OR_ARCHIVES = 'Information Assets - Email Lists / Archives'
  TERM_INFORMATION_ASSETS_FINANCIAL_DATA = 'Information Assets - Financial Data'
  TERM_INFORMATION_ASSETS_INTELLECTUAL_PROPERTY = 'Information Assets - Intellectual Property'
  TERM_INFORMATION_ASSETS_MOBILE_PHONE_CONTACTS = 'Information Assets - Mobile Phone Contacts'
  TERM_INFORMATION_ASSETS_USER_CREDENTIALS = 'Information Assets - User Credentials'

class stix.common.vocabs.IntendedEffect_1_0(value=None)
  Bases: stix.common.vocabs.VocabString
  TERM_ACCOUNT_TAKEOVER = 'Account Takeover'
  TERM_ADVANTAGE = 'Advantage'
TERM_ADVANTAGE_ECONOMIC = ‘Advantage - Economic’
TERM_ADVANTAGE_MILITARY = ‘Advantage - Military’
TERM_ADVANTAGE_POLITICAL = ‘Advantage - Political’
TERM_BRAND_DAMAGE = ‘Brand Damage’
TERM_COMPETITIVE_ADVANTAGE = ‘Competitive Advantage’
TERM_DEGRADATION_OF_SERVICE = ‘Degradation of Service’
TERM_DENIAL_AND_DECEPTION = ‘Denial and Deception’
TERM_DESTRUCTION = ‘Destruction’
TERM_DISRUPTION = ‘Disruption’
TERM_EMBARRASSMENT = ‘Embarrassment’
TERM_EXPOSURE = ‘Exposure’
TERM_EXTORTION = ‘Extortion’
TERM_FRAUD = ‘Fraud’
TERM_HARASSMENT = ‘Harassment’
TERM_ICS_CONTROL = ‘ICS Control’
TERM_THEFT = ‘Theft’
TERM_THEFT_CREDENTIAL_THEFT = ‘Theft - Credential Theft’
TERM_THEFT_IDENTITY_THEFT = ‘Theft - Identity Theft’
TERM_THEFT_INTELLECTUAL_PROPERTY = ‘Theft - Intellectual Property’
TERM_THEFT_THEFT_OF_PROPRIETARY_INFORMATION = ‘Theft - Theft of Proprietary Information’
TERM_TRAFFIC_DIVERSION = ‘Traffic Diversion’
TERM_UNAUTHORIZED_ACCESS = ‘Unauthorized Access’

class stix.common.vocabs.LocationClass_1_0(value=None)
    Bases: stix.common.vocabs.VocabString
    TERM_COLOCATED = ‘Co-Located’
    TERM_EXTERNALLYLOCATED = ‘Externally-Located’
    TERM_INTERNALLYLOCATED = ‘Internally-Located’
    TERM_MOBILE = ‘Mobile’
    TERM_UNKNOWN = ‘Unknown’

class stix.common.vocabs.LossDuration_1_0(value=None)
    Bases: stix.common.vocabs.VocabString
    TERM_DAYS = ‘Days’
    TERM_HOURS = ‘Hours’
    TERM_MINUTES = ‘Minutes’
    TERM_PERMANENT = ‘Permanent’
    TERM_SECONDS = ‘Seconds’
    TERM_UNKNOWN = ‘Unknown’
TERM_WEEKS = 'Weeks'
class stix.common.vocabs.LossProperty_1_0(value=None):
    Bases: stix.common.vocabs.VocabString
    TERM_ACCOUNTABILITY = 'Accountability'
    TERM_AVAILABILITY = 'Availability'
    TERM_CONFIDENTIALITY = 'Confidentiality'
    TERM_INTEGRITY = 'Integrity'
    TERM_NONREPUTATION = 'Non-Repudiation'
class stix.common.vocabs.MalwareType_1_0(value=None):
    Bases: stix.common.vocabs.VocabString
    TERM_ADWARE = 'Adware'
    TERM_AUTOMATED_TRANSFER_SCRIPTS = 'Automated Transfer Scripts'
    TERM_BOT = 'Bot'
    TERM_BOT_CREDENTIAL_THEFT = 'Bot - Credential Theft'
    TERM_BOT_DOS = 'Bot - DDoS'
    TERM_BOT_LOADER = 'Bot - Loader'
    TERM_BOT_SPAM = 'Bot - Spam'
    TERM_DIALER = 'Dialer'
    TERM_DOS_OR_DDOS = 'DoS / DDoS'
    TERM_DOS_OR_DDOS_PARTICIPATORY = 'DoS / DDoS - Participatory'
    TERM_DOS_OR_DDOS_SCRIPT = 'DoS / DDoS - Script'
    TERM_DOS_OR_DDOS_STRESS_TESTTOOLS = 'DoS / DDoS - Stress Test Tools'
    TERM_EXPLOIT_KITS = 'Exploit Kits'
    TERM_POS_OR_ATM_MALWARE = 'POS / ATM Malware'
    TERM_RANSOMWARE = 'Ransomware'
    TERM_REMOTE_ACCESS_TROJAN = 'Remote Access Trojan'
    TERM_ROGUE_ANTIVIRUS = 'Rogue Antivirus'
    TERM_ROOTKIT = 'Rootkit'
class stix.common.vocabs.ManagementClass_1_0(value=None):
    Bases: stix.common.vocabs.VocabString
    TERM_COMANAGEMENT = 'Co-Management'
    TERM_EXTERNALLYMANAGED = 'Externally-Managed'
    TERM_INTERNALLYMANAGED = 'Internally-Managed'
    TERM_UNKNOWN = 'Unknown'
class stix.common.vocabs.Motivation_1_0(value=None):
    Bases: stix.common.vocabs.VocabString
    TERM_EGO = 'Ego'
TERM_FINANCIAL_OR_ECONOMIC = 'Financial or Economic'
TERM_IDEOLOGICAL = 'Ideological'
TERM_IDEOLOGICAL_ANTICORRUPTION = 'Ideological - Anti-Corruption'
TERM_IDEOLOGICAL_ANTISTABLISHMENT = 'Ideological - Anti-Establishment'
TERM_IDEOLOGICAL_ENVIRONMENTAL = 'Ideological - Environmental'
TERM_IDEOLOGICAL_ETHNIC_NATIONALIST = 'Ideological - Ethnic / Nationalist'
TERM_IDEOLOGICAL_HUMAN_RIGHTS = 'Ideological - Human Rights'
TERM_IDEOLOGICAL_INFORMATION_FREEDOM = 'Ideological - Information Freedom'
TERM_IDEOLOGICAL_RELIGIOUS = 'Ideological - Religious'
TERM_IDEOLOGICAL_SECURITY_AWARENESS = 'Ideological - Security Awareness'
TERM_MILITARY = 'Military'
TERM_OPPORTUNISTIC = 'Opportunistic'
TERM_POLICITAL = 'Policital'
class stix.common.vocabs.Motivation_1_0_1(value=None)
    Bases: stix.common.vocabs.VocabString
TERM_EGO = 'Ego'
TERM_FINANCIAL_OR_ECONOMIC = 'Financial or Economic'
TERM_IDEOLOGICAL = 'Ideological'
TERM_IDEOLOGICAL_ANTICORRUPTION = 'Ideological - Anti-Corruption'
TERM_IDEOLOGICAL_ANTISTABLISHMENT = 'Ideological - Anti-Establishment'
TERM_IDEOLOGICAL_ENVIRONMENTAL = 'Ideological - Environmental'
TERM_IDEOLOGICAL_ETHNIC_NATIONALIST = 'Ideological - Ethnic / Nationalist'
TERM_IDEOLOGICAL_HUMAN_RIGHTS = 'Ideological - Human Rights'
TERM_IDEOLOGICAL_INFORMATION_FREEDOM = 'Ideological - Information Freedom'
TERM_IDEOLOGICAL_RELIGIOUS = 'Ideological - Religious'
TERM_MILITARY = 'Military'
TERM_OPPORTUNISTIC = 'Opportunistic'
TERM_POLICITAL = 'Policital'
class stix.common.vocabs.Motivation_1_1(value=None)
    Bases: stix.common.vocabs.VocabString
TERM_EGO = 'Ego'
TERM_FINANCIAL_OR_ECONOMIC = 'Financial or Economic'
TERM_IDEOLOGICAL = 'Ideological'
TERM_IDEOLOGICAL_ANTICORRUPTION = 'Ideological - Anti-Corruption'
TERM_IDEOLOGICAL_ANTISTABLISHMENT = 'Ideological - Anti-Establishment'
TERM_IDEOLOGICAL_ENVIRONMENTAL = 'Ideological - Environmental'
TERM_IDEOLOGICAL_ETHNIC_NATIONALIST = 'Ideological - Ethnic / Nationalist'
TERM_IDEOLOGICAL_HUMAN_RIGHTS = 'Ideological - Human Rights'
TERM_IDEOLOGICAL_INFORMATION_FREEDOM = 'Ideological - Information Freedom'
TERM_IDEOLOGICAL_RELIGIOUS = 'Ideological - Religious'
TERM_MILITARY = 'Military'
TERM_OPPORTUNISTIC = 'Opportunistic'
TERM_POLICITAL = 'Policital'
TERM_IDEOLOGICAL_ETHNIC_NATIONALIST = ‘Ideological - Ethnic / Nationalist’
TERM_IDEOLOGICAL_HUMAN_RIGHTS = ‘Ideological - Human Rights’
TERM_IDEOLOGICAL_INFORMATION_FREEDOM = ‘Ideological - Information Freedom’
TERM_IDEOLOGICAL_RELIGIOUS = ‘Ideological - Religious’
TERM_IDEOLOGICAL_SECURITY_AWARENESS = ‘Ideological - Security Awareness’
TERM_MILITARY = ‘Military’
TERM_OPPORTUNISTIC = ‘Opportunistic’
TERM_POLITICAL = ‘Political’

class stix.common.vocabs.OwnershipClass_1_0 (value=None)
    Bases: stix.common.vocabs.VocabString
    TERM_CUSTOMEROWNED = ‘Customer-Owned’
    TERM_EMPLOYEEOWNED = ‘Employee-Owned’
    TERM_INTERNALLYOWNED = ‘Internally-Owned’
    TERM_PARTNEROWNED = ‘Partner-Owned’
    TERM_UNKNOWN = ‘Unknown’

class stix.common.vocabs.PackageIntent_1_0 (value=None)
    Bases: stix.common.vocabs.VocabString
    TERM_ATTACK_PATTERN_CHARACTERIZATION = ‘Attack Pattern Characterization’
    TERM_CAMPAIGN_CHARACTERIZATION = ‘Campaign Characterization’
    TERM_COLLECTIVE_THREAT_INTELLIGENCE = ‘Collective Threat Intelligence’
    TERM_COURSES_OF_ACTION = ‘Courses of Action’
    TERM_EXPLOIT_CHARACTERIZATION = ‘Exploit Characterization’
    TERM_INCIDENT = ‘Incident’
    TERM_INDICATORS = ‘Indicators’
    TERM_INDICATORS_ENDPOINT_CHARACTERISTICS = ‘Indicators - Endpoint Characteristics’
    TERM_INDICATORS_MALWARE_ARTIFACTS = ‘Indicators - Malware Artifacts’
    TERM_INDICATORS_NETWORK_ACTIVITY = ‘Indicators - Network Activity’
    TERM_INDICATORS_PHISHING = ‘Indicators - Phishing’
    TERM_INDICATORS_WATCHLIST = ‘Indicators - Watchlist’
    TERM_MALWARE_CHARACTERIZATION = ‘Malware Characterization’
    TERM_MALWARE_SAMPLES = ‘Malware Samples’
    TERM_OBSERVATIONS = ‘Observations’
    TERM_OBSERVATIONS_EMAIL = ‘Observations - Email’
    TERM_THREAT_ACTOR_CHARACTERIZATION = ‘Threat Actor Characterization’
    TERM_THREAT_REPORT = ‘Threat Report’
    TERM_TTP_INFRASTRUCTURE = ‘TTP - Infrastructure’
    TERM_TTP_TOOLS = ‘TTP - Tools’
class stix.common.vocabs.PlanningAndOperationalSupport_1_0(value=None)
    Bases: stix.common.vocabs.VocabString

    TERM_DATA_EXPLOITATION = 'Data Exploitation'
    TERM_DATA_EXPLOITATION_ANALYTIC_SUPPORT = 'Data Exploitation - Analytic Support'
    TERM_DATA_EXPLOITATION_TRANSLATION_SUPPORT = 'Data Exploitation - Translation Support'
    TERM_FINANCIAL_RESOURCES = 'Financial Resources'
    TERM_FINANCIAL_RESOURCES_ACADEMIC = 'Financial Resources - Academic'
    TERM_FINANCIAL_RESOURCES_COMMERCIAL = 'Financial Resources - Commercial'
    TERM_FINANCIAL_RESOURCES_GOVERNMENT = 'Financial Resources - Government'
    TERM_FINANCIAL_RESOURCES_HACKTIVIST_OR_GRASSROOT = 'Financial Resources - Hacktivist or Grassroot'
    TERM_FINANCIAL_RESOURCES_NONATTRIBUTABLE_FINANCE = 'Financial Resources - Non-Attributable Finance'
    TERM_PLANNING = 'Planning'
    TERM_PLANNING_OPEN_SOURCE_INTELLIGENCE_OSINT_GETHERING = 'Planning - Open-Source Intelligence (OSINT) Gethering'
    TERM_PLANNING_OPERATIONAL_COVER_PLAN = 'Planning - Operational Cover Plan'
    TERM_PLANNING_PRE_OPERATIONAL_SURVEILLANCE_AND_RECONNAISSANCE = 'Planning - Pre-Operational Surveillance and Reconnaissance'
    TERM_PLANNING_TARGET_SELECTION = 'Planning - Target Selection'
    TERM_SKILL_DEVELOPMENT_RECRUITMENT = 'Skill Development / Recruitment'
    TERM_SKILL_DEVELOPMENT_RECRUITMENT_CONTRACTING_AND_HIRING = 'Skill Development / Recruitment - Contracting and Hiring'
    TERM_SKILL_DEVELOPMENT_RECRUITMENT_DOCUMENT_EXPLOITATION_DOCEX_TRAINING = 'Skill Development / Recruitment - Document Exploitation (DOCEX) Training'
    TERM_SKILL_DEVELOPMENT_RECRUITMENT_INTERNAL_TRAINING = 'Skill Development / Recruitment - Internal Training'
    TERM_SKILL_DEVELOPMENT_RECRUITMENT_MILITARY_PROGRAMS = 'Skill Development / Recruitment - Military Programs'
    TERM_SKILL_DEVELOPMENT_RECRUITMENT_SECURITY_HACKER_CONFERENCES = 'Skill Development / Recruitment - Security / Hacker Conferences'
    TERM_SKILL_DEVELOPMENT_RECRUITMENT_UNDERGROUND_FORUMS = 'Skill Development / Recruitment - Underground Forums'
    TERM_SKILL_DEVELOPMENT_RECRUITMENT_UNIVERSITY_PROGRAMS = 'Skill Development / Recruitment - University Programs'

class stix.common.vocabs.PlanningAndOperationalSupport_1_0_1(value=None)
    Bases: stix.common.vocabs.VocabString

    TERM_DATA_EXPLOITATION = 'Data Exploitation'
    TERM_DATA_EXPLOITATION_ANALYTIC_SUPPORT = 'Data Exploitation - Analytic Support'
    TERM_DATA_EXPLOITATION_TRANSLATION_SUPPORT = 'Data Exploitation - Translation Support'
    TERM_FINANCIAL_RESOURCES = 'Financial Resources'
    TERM_FINANCIAL_RESOURCES_ACADEMIC = 'Financial Resources - Academic'
    TERM_FINANCIAL_RESOURCES_COMMERCIAL = 'Financial Resources - Commercial'
    TERM_FINANCIAL_RESOURCES_GOVERNMENT = 'Financial Resources - Government'
    TERM_FINANCIAL_RESOURCES_HACKTIVIST_OR_GRASSROOT = 'Financial Resources - Hacktivist or Grassroot'
    TERM_FINANCIAL_RESOURCES_NONATTRIBUTABLE_FINANCE = 'Financial Resources - Non-Attributable Finance'
    TERM_PLANNING = 'Planning'
    TERM_PLANNING_OPEN_SOURCE_INTELLIGENCE_OSINT_GETHERING = 'Planning - Open-Source Intelligence (OSINT) Gethering'
TERM_PLANNING_OPERATIONAL_COVER_PLAN = 'Planning - Operational Cover Plan'
TERM_PLANNING_PREOPERATIONAL_SURVEILLANCE_AND_RECONNAISSANCE = 'Planning - Pre-Operational Surveillance and Reconnaissance'
TERM_PLANNING_TARGET_SELECTION = 'Planning - Target Selection'
TERM_Skill_Development_or_Recruitment = 'Skill Development / Recruitment'
TERM_Skill_Development_or_Recruitment_CONTRACTING_AND_HIRING = 'Skill Development / Recruitment - Contracting and Hiring'
TERM_Skill_Development_or_Recruitment_DOCUMENT_EXPLOITATION_DOCEX_TRAINING = 'Skill Development / Recruitment - Document Exploitation (DOCEX) Training'
TERM_Skill_Development_or_Recruitment_Internal_Training = 'Skill Development / Recruitment - Internal Training'
TERM_Skill_Development_or_Recruitment_Military_Programs = 'Skill Development / Recruitment - Military Programs'
TERM_Skill_Development_or_Recruitment_SECURITY_OR_HACKER_CONFERENCES = 'Skill Development / Recruitment - Security / Hacker Conferences'
TERM_Skill_Development_or_Recruitment_UNDERGROUND_FORUMS = 'Skill Development / Recruitment - Underground Forums'
TERM_Skill_Development_or_Recruitment_UNIVERSITY_PROGRMS = 'Skill Development / Recruitment - University Programs'

class stix.common.vocabs.ReportIntent_1_0 (value=None)
Bases: stix.common.vocabs.VocabString
TERM_ATTACK_PATTERN_CHARACTERIZATION = 'Attack Pattern Characterization'
TERM_CAMPAIGN_CHARACTERIZATION = 'Campaign Characterization'
TERM_COLLECTIVE_THREAT_INTELLIGENCE = 'Collective Threat Intelligence'
TERM_Courses_of_Action = 'Courses of Action'
TERM_EXPLOIT_CHARACTERIZATION = 'Exploit Characterization'
TERM_INCIDENT = 'Incident'
TERM_INDICATORS = 'Indicators'
TERM_INDICATORS_ENDPOINT_CHARACTERISTICS = 'Indicators - Endpoint Characteristics'
TERM_INDICATORS_MALWARE_ARTIFACTS = 'Indicators - Malware Artifacts'
TERM_INDICATORS_NETWORK_ACTIVITY = 'Indicators - Network Activity'
TERM_INDICATORS_PHISHING = 'Indicators - Phishing'
TERM_INDICATORS_WATCHLIST = 'Indicators - Watchlist'
TERM_MALWARE_CHARACTERIZATION = 'Malware Characterization'
TERM_MALWARE_SAMPLES = 'Malware Samples'
TERM_OBSERVATIONS = 'Observations'
TERM_OBSERVATIONS_EMAIL = 'Observations - Email'
TERM_THREAT_ACTOR_CHARACTERIZATION = 'Threat Actor Characterization'
TERM_THREAT_REPORT = 'Threat Report'
TERM_TTP_INFRASTRUCTURE = 'TTP - Infrastructure'
TERM_TTP_TOOLS = 'TTP - Tools'

class stix.common.vocabs.SecurityCompromise_1_0 (value=None)
Bases: stix.common.vocabs.VocabString
TERM_NO = 'No'
TERM_SUSPECTED = 'Suspected'
TERM_UNKNOWN = ‘Unknown’
TERM_YES = ‘Yes’
class stix.common.vocabs.SystemType_1_0(value=None)
    Bases: stix.common.vocabs.VocabString
    TERM_ENTERPRISE_SYSTEMS = ‘Enterprise Systems’
    TERM_ENTERPRISE_SYSTEMS_APPLICATION_LAYER = ‘Enterprise Systems - Application Layer’
    TERM_ENTERPRISE_SYSTEMS_DATABASE_LAYER = ‘Enterprise Systems - Database Layer’
    TERM_ENTERPRISE_SYSTEMS_ENTERPRISE_TECHNOLOGIES_AND_SUPPORT_INFRASTRUCTURE = ‘Enterprise Systems - Enterprise Technologies and Support Infrastructure’
    TERM_ENTERPRISE_SYSTEMS_NETWORKING_DEVICES = ‘Enterprise Systems - Networking Devices’
    TERM_ENTERPRISE_SYSTEMS_NETWORK_SYSTEMS = ‘Enterprise Systems - Network Systems’
    TERM_ENTERPRISE_SYSTEMS_VOIP = ‘Enterprise Systems - VoIP’
    TERM_ENTERPRISE_SYSTEMS_WEB_LAYER = ‘Enterprise Systems - Web Layer’
    TERM_INDUSTRIAL_CONTROL_SYSTEMS = ‘Industrial Control Systems’
    TERM_INDUSTRIAL_CONTROL_SYSTEMS_EQUIPMENT_UNDER_CONTROL = ‘Industrial Control Systems - Equipment Under Control’
    TERM_INDUSTRIAL_CONTROL_SYSTEMS_OPERATIONS_MANAGEMENT = ‘Industrial Control Systems - Operations Management’
    TERM_INDUSTRIAL_CONTROL_SYSTEMS_SAFETY_PROTECTION_AND_LOCAL_CONTROL = ‘Industrial Control Systems - Safety, Protection and Local Control’
    TERM_INDUSTRIAL_CONTROL_SYSTEMS_SUPERVISORY_CONTROL = ‘Industrial Control Systems - Supervisory Control’
    TERM_MOBILE_SYSTEMS = ‘Mobile Systems’
    TERM_MOBILE_SYSTEMS_MOBILE_DEVICES = ‘Mobile Systems - Mobile Devices’
    TERM_MOBILE_SYSTEMS_MOBILE_OPERATING SYSTEMS = ‘Mobile Systems - Mobile Operating Systems’
    TERM_MOBILE_SYSTEMS_NEAR_FIELD_COMMUNICATIONS = ‘Mobile Systems - Near Field Communications’
    TERM_THIRDPARTY_SERVICES = ‘Third-Party Services’
    TERM_THIRDPARTY_SERVICES_APPLICATION_STORES = ‘Third-Party Services - Application Stores’
    TERM_THIRDPARTY_SERVICES_CLOUD SERVICES = ‘Third-Party Services - Cloud Services’
    TERM_THIRDPARTY_SERVICES_SECURITY_VENDORS = ‘Third-Party Services - Security Vendors’
    TERM_THIRDPARTY_SERVICES_SOCIAL_MEDIA = ‘Third-Party Services - Social Media’
    TERM_THIRDPARTY_SERVICES_SOFTWARE_UPDATE = ‘Third-Party Services - Software Update’
    TERM_USERS = ‘Users’
    TERM_USERS_APPLICATION_AND_SOFTWARE = ‘Users - Application And Software’
    TERM_USERS_REMOVABLE_MEDIA = ‘Users - Removable Media’
    TERM_USERS_WORKSTATION = ‘Users - Workstation’
class stix.common.vocabs.ThreatActorSophistication_1_0(value=None)
    Bases: stix.common.vocabs.VocabString
    TERM_ASPIRANT = ‘Aspirant’
    TERM_EXPERT = ‘Expert’
    TERM_INNOVATOR = ‘Innovator’
    TERM_NOVICE = ‘Novice’
TERM_PRACTITIONER = ‘Practitioner’

class stix.common.vocabs.ThreatActorType_1_0 (value=None)
    Bases: stix.common.vocabs.VocabString
    TERM_CYBER_ESPIONAGE_OPERATIONS = ‘Cyber Espionage Operations’
    TERM_DISGRUNTLED_CUSTOMER_OR_USER = ‘Disgruntled Customer / User’
    TERM_ECRIME_ACTOR_CREDENTIAL_THEFT_BOTNET_OPERATOR = ‘eCrime Actor - Credential Theft Botnet Operator’
    TERM_ECRIME_ACTOR_CREDENTIAL_THEFT_BOTNET_SERVICE = ‘eCrime Actor - Credential Theft Botnet Service’
    TERM_ECRIME_ACTOR_MALWARE_DEVELOPER = ‘eCrime Actor - Malware Developer’
    TERM_ECRIME_ACTOR_MONEY_LAUNDERING_NETWORK = ‘eCrime Actor - Money Laundering Network’
    TERM_ECRIME_ACTOR_ORGANIZED_CRIME_ACTOR = ‘eCrime Actor - Organized Crime Actor’
    TERM_ECRIME_ACTOR_SPAM_SERVICE = ‘eCrime Actor - Spam Service’
    TERM_ECRIME_ACTOR_TRAFFIC_SERVICE = ‘eCrime Actor - Traffic Service’
    TERM_ECRIME_ACTOR_UNDERGROUND_CALL_SERVICE = ‘eCrime Actor - Underground Call Service’
    TERM_HACKER = ‘Hacker’
    TERM_HACKER_BLACK_HAT = ‘Hacker - Black hat’
    TERM_HACKER_GRAY_HAT = ‘Hacker - Gray hat’
    TERM_HACKER_WHITE_HAT = ‘Hacker - White hat’
    TERM_HACKTIVIST = ‘Hacktivist’
    TERM_INSIDER_THREAT = ‘Insider Threat’
    TERM_STATE_ACTOR_OR_AGENCY = ‘State Actor / Agency’

class stix.common.vocabs.Versioning_1_0 (value=None)
    Bases: stix.common.vocabs.VocabString
    TERM_REVOKE = ‘Revokes’
    TERM_UPDATES_REVIS = ‘Updates - Revises’
    TERM_UPDATE_CORRECTS = ‘Updates - Corrects’

class stix.common.vocabs.VocabString (value=None)
    Bases: stix.base.Entity

is_plain ()
    Whether the VocabString can be represented as a single value.

stix.common.vocabs.AssetType
    alias of AssetType_1_0

stix.common.vocabs.AttackerInfrastructureType
    alias of AttackerInfrastructureType_1_0

stix.common.vocabs.AttackerToolType
    alias of AttackerToolType_1_0

stix.common.vocabs.AvailabilityLossType
    alias of AvailabilityLossType_1_1_1

stix.common.vocabs.CampaignStatus
    alias of CampaignStatus_1_0
stix.common.vocabs.COASTage
  alias of COASTage_1.0

stix.common.vocabs.CourseOfActionType
  alias of CourseOfActionType_1.0

stix.common.vocabs.DiscoveryMethod
  alias of DiscoveryMethod_2.0

stix.common.vocabs.HighMediumLow
  alias of HighMediumLow_1.0

stix.common.vocabs.ImpactQualification
  alias of ImpactQualification_1.0

stix.common.vocabs.ImpactRating
  alias of ImpactRating_1.0

stix.common.vocabs.IncidentCategory
  alias of IncidentCategory_1.0

stix.common.vocabs.IncidentEffect
  alias of IncidentEffect_1.0

stix.common.vocabs.IncidentStatus
  alias of IncidentStatus_1.0

stix.common.vocabs.IndicatorType
  alias of IndicatorType_1.1

stix.common.vocabs.InformationSourceRole
  alias of InformationSourceRole_1.0

stix.common.vocabs.InformationType
  alias of InformationType_1.0

stix.common.vocabs.IntendedEffect
  alias of IntendedEffect_1.0

stix.common.vocabs.LocationClass
  alias of LocationClass_1.0

stix.common.vocabs.LossDuration
  alias of LossDuration_1.0

stix.common.vocabs.LossProperty
  alias of LossProperty_1.0

stix.common.vocabs.MalwareType
  alias of MalwareType_1.0

stix.common.vocabs.ManagementClass
  alias of ManagementClass_1.0

stix.common.vocabs.Motivation
  alias of Motivation_1.1

stix.common.vocabsOwnershipClass
  alias of OwnershipClass_1.0

stix.common.vocabs.PackageIntent
  alias of PackageIntent_1.0
Functions

stix.common.vocabs.add_vocab(cls)
Registers a VocabString subclass.

Note: The register_vocab() class decorator has replaced this method.

stix.common.vocabs.register_vocab(cls)
Class decorator that registers a VocabString subclass.
Also, calculate all the permitted values for class being decorated by adding an _ALLOWED_VALUES tuple of all the values of class members beginning with TERM_.

3.1.4 STIX Core

Modules located in the stix.core package

Version: 1.2.0.8

stix.core.stix_header Module

Classes

class stix.core.stix_header.STIXHeader (package_intents=None, description=None, handling=None, information_source=None, title=None, short_description=None)

Bases: stix.base.Entity
The STIX Package Header.

Parameters

• handling – The data marking section of the Header.
• information_source – The InformationSource section of the Header.
• package_intents – DEPRECATED. A collection of VocabString defining the intent of the parent STIXPackage.
• description – DEPRECATED. A description of the intent or purpose of the parent STIXPackage.
• **short_description** – **DEPRECATED.** A short description of the intent or purpose of the parent STIXPackage.

• **title** – **DEPRECATED.** The title of the STIXPackage.

**profiles**
A collection of STIX Profiles the parent STIXPackage conforms to.

**title**
DEPRECATED. The title of the parent STIXPackage.

**add_description** *(description)*
DEPRECATED. Adds a description to the descriptions collection.

This is the same as calling “foo.descriptions.add(bar)”.

**add_package_intent** *(package_intent)*
DEPRECATED. Adds VocabString object to the package_intents collection.

If the input is not an instance of VocabString, an effort will be made to convert it into an instance of PackageIntent.

**add_profile** *(profile)*
Adds a profile to the STIX Header. A Profile is represented by a string URI.

**add_short_description** *(description)*
DEPRECATED. Adds a description to the short_descriptions collection.

This is the same as calling “foo.short_descriptions.add(bar)”.

**description**
DEPRECATED. A single description about the contents or purpose of this object.

Default Value: None

**Note:** If this object has more than one description set, this will return the description with the lowest ordinality value.

**Returns** An instance of StructuredText

**short_description**
DEPRECATED. A single short description about the contents or purpose of this object.

Default Value: None

**Note:** If this object has more than one short description set, this will return the description with the lowest ordinality value.

**Returns** An instance of StructuredText

**Version:** 1.2.0.8

**stix.core.stix_package Module**
Overview

The `stix.core.stix_package` module implements `STIXPackage`.

STIXType defines a bundle of information characterized in the Structured Threat Information eXpression (STIX) language.

Documentation Resources

- STIX Package Data Model

Classes

```python
class stix.core.stix_package.STIXPackage(id_=None, idref=None, timestamp=None, stix_header=None, courses_of_action=None, exploit_targets=None, indicators=None, observables=None, incidents=None, threat_actors=None, ttps=None, campaigns=None, related_packages=None, reports=None)
```

Bases: `stix.base.Entity`

A STIX Package object.

Parameters

- `id` *(optional)*  – An identifier. If None, a value will be generated via `mixbox.idgen.create_id()`. If set, this will unset the `idref` property.
- `idref`  – DEPRECATED  An identifier reference. If set this will unset the `id_` property.
- `timestamp`  – DEPRECATED  A timestamp value. Can be an instance of `datetime.datetime` or `str`.
- `header`  – A Report `Header` object.
- `campaigns`  – A collection of `Campaign` objects.
- `course_of_action`  – A collection of `CourseOfAction` objects.
- `exploit_targets`  – A collection of `ExploitTarget` objects.
- `incidents`  – A collection of `Incident` objects.
- `indicators`  – A collection of `Indicator` objects.
- `threat_actors`  – A collection of `ThreatActor` objects.
- `ttps`  – A collection of `TTP` objects.
- `related_packages`  – DEPRECATED. A collection of `RelatedPackage` objects.
- `reports`  – A collection of `Report` objects.

```
add(entity)
Add `entity` to a top-level collection. For example, if `entity` is an `Indicator` object, the `entity` will be added to the `indicators` top-level collection.

add_campaign(campaign)
Add a `Campaign` object to the `campaigns` collection.

add_course_of_action(course_of_action)
Add a `CourseOfAction` object to the `courses_of_action` collection.
```
add_exploit_target (exploit_target)
   Adds an ExploitTarget object to the exploit_targets collection.

add_incident (incident)
   Adds an Incident object to the incidents collection.

add_indicator (indicator)
   Adds an Indicator object to the indicators collection.

add.observable (observable)
   Adds an Observable object to the observables collection.
   If observable is not an Observable instance, an effort will be made to convert it to one.

add_related_package (related_package)
   Adds a RelatedPackage object to the related_packages collection.

add_report (report)
   Adds a Report object to the reports collection.

add_threat_actor (threat_actor)
   Adds an ThreatActor object to the threat_actors collection.

add_ttp (ttp)
   Adds an TTP object to the ttps collection.

find (id_)
   Searches the children of a Entity implementation for an object with an id_ property that matches id_.

classmethod from_xml (xml_file, encoding=None)
   Parses the xml_file file-like object and returns a STIXPackage instance.

   Parameters
   
   • xml_file – A file, file-like object, etree.Element, or etree._ElementTree instance.
   • encoding – The character encoding of the xml_file input. If None, an attempt will be made to determine the input character encoding. Default is None.

   Returns An instance of STIXPackage.

to_dict ()
   Convert to a dict
   Subclasses can override this function.

   Returns Python dict with keys set from this Entity.

to_dict ()
   Convert to a dict
   Subclasses can override this function.

   Returns Python dict with keys set from this Entity.

to_json()
   Export an object as a JSON String.

to_obj (ns_info=None)
   Convert to a GenerateDS binding object.
   Subclasses can override this function.

   Returns An instance of this Entity's _binding_class with properties set from this Entity.
to_obj (ns_info=None)
Convert to a GenerateDS binding object.
Subclasses can override this function.

Returns An instance of this Entity's _binding_class with properties set from this Entity.

to_xml (include_namespaces=True, include_schemalocs=False, ns_dict=None, schemaloc_dict=None,
pretty=True, auto_namespace=True, encoding='utf-8')
Serializes a Entity instance to an XML string.

The default character encoding is utf-8 and can be set via the encoding parameter. If encoding is None,
a string (unicode in Python 2, str in Python 3) is returned.

Parameters

- auto_namespace – Automatically discover and export XML namespaces for a STIX Entity instance.

- include_namespaces – Export namespace definitions in the output XML. Default is True.

- include_schemalocs – Export xsi:schemaLocation attribute in the output document. This will attempt to associate namespaces declared in the STIX document with schema locations. If a namespace cannot be resolved to a schemaLocation, a Python warning will be raised. Schemalocations will only be exported if include_namespaces is also True.

- ns_dict – Dictionary of XML definitions (namespace is key, alias is value) to include in the exported document. This must be passed in if auto_namespace is False.

- schemaloc_dict – Dictionary of XML namespace: schema location mappings to include in the exported document. These will only be included if auto_namespace is False.

- pretty – Pretty-print the XML.

- encoding – The output character encoding. Default is utf-8. If encoding is set to None, a string (unicode in Python 2, str in Python 3) is returned.

Returns An XML string for this Entity instance. Default character encoding is utf-8.
**stix.coa Module**

**Overview**

The `stix.coa` module implements `CourseOfAction`.

CoursesOfAction are specific measures to be taken to address threat whether they are corrective or preventative to address ExploitTargets, or responsive to counter or mitigate the potential impacts of Incidents.

**Documentation Resources**

- Course Of Action Data Model

**Classes**


Bases: `stix.base.BaseCoreComponent`

Implementation of the STIX Course of Action.

**Parameters**

- **id (optional)** – An identifier. If None, a value will be generated via `mixbox.idgen.create_id()`. If set, this will unset the idref property.
- **idref (optional)** – An identifier reference. If set this will unset the id_ property.
- **timestamp (optional)** – A timestamp value. Can be an instance of `datetime.datetime` or `str`.
- **description** – A description of the purpose or intent of this object.
- **short_description** – A short description of the intent or purpose of this object.
- **title** – The title of this object.

**add_description**(description)

Adds a description to the descriptions collection.

This is the same as calling “foo.descriptions.add(bar)”.

**add_short_description**(description)

Adds a description to the short_descriptions collection.

This is the same as calling “foo.short_descriptions.add(bar)”.

**description**

A single description about the contents or purpose of this object.

Default Value: None

**Note:** If this object has more than one description set, this will return the description with the lowest ordinality value.

**Returns**

An instance of `StructuredText`

**find**(id_)

Searches the children of a `Entity` implementation for an object with an `id_` property that matches `id_`.
**short_description**
A single short description about the contents or purpose of this object.

Default Value: None

**Note:** If this object has more than one short description set, this will return the description with the lowest ordinality value.

**Returns** An instance of **StructuredText**

**to_dict()**
Convert to a dict

Subclasses can override this function.

**Returns** Python dict with keys set from this Entity.

**to_json()**
Export an object as a JSON String.

**to_obj**(ns_info=None)
Convert to a GenerateDS binding object.

Subclasses can override this function.

**Returns** An instance of this Entity’s _binding_class with properties set from this Entity.

class stix.coa.RelatedCOAs(scope=None, *args)
    Bases: stix.common.related.GenericRelationshipList

Version: 1.2.0.8

**stix.coa.objective Module**

**Classes**

class stix.coa.objective.Objective(description=None, short_description=None)
    Bases: stix.base.Entity

    **add_description**(description)
    Adds a description to the descriptions collection.

    This is the same as calling “foo.descriptions.add(bar)”.

    **add_short_description**(description)
    Adds a description to the short_descriptions collection.

    This is the same as calling “foo.short_descriptions.add(bar)”.

    **description**
    A single description about the contents or purpose of this object.

    Default Value: None

    **Note:** If this object has more than one description set, this will return the description with the lowest ordinality value.
Returns An instance of `StructuredText`

**short_description**
A single short description about the contents or purpose of this object.

Default Value: None

Note: If this object has more than one short description set, this will return the description with the lowest ordinality value.

Returns An instance of `StructuredText`

### 3.1.6 STIX Exploit Target

Modules located in the stix.exploit_target package

**Version**: 1.2.0.8

**stix.exploit_target Module**

**Overview**

The `stix.exploit_target` module implements `ExploitTarget`.

This denotes the specific vulnerability, weakness, or software configuration that creates a security risk.

**Documentation Resources**

- Exploit Target Data Model
- Exploit Target Idioms

**Classes**

```python
class stix.exploit_target.ExploitTarget (id_=None, idref=None, timestamp=None, title=None, description=None, short_description=None)
```

Bases: `stix.base.BaseCoreComponent`

Implementation of STIX Exploit Target.

**Parameters**

- `id` *(optional)* – An identifier. If None, a value will be generated via `mixbox.idgen.create_id()`. If set, this will unset the `idref` property.
- `idref` *(optional)* – An identifier reference. If set this will unset the `id_` property.
- `title` *(optional)* – A string title.
- `timestamp` *(optional)* – A timestamp value. Can be an instance of `datetime.datetime` or `str`.
- `description` *(optional)* – A string description.
- `short_description` *(optional)* – A string short description.
add_configuration(value)
    Adds a configuration to the configurations list property.

    **Note:** If `None` is passed in no value is added

    **Parameters** value – A configuration value.
    **Raises** ValueError – If the value param is of type `Configuration`

add_description(description)
    Adds a description to the descriptions collection.
    This is the same as calling “foo.descriptions.add(bar)”.

add_short_description(description)
    Adds a description to the short_descriptions collection.
    This is the same as calling “foo.short_descriptions.add(bar)”.

add_vulnerability(value)
    Adds a vulnerability to the vulnerabilities list property.

    **Note:** If `None` is passed in no value is added

    **Parameters** value – A `Vulnerability` object
    **Raises** ValueError – if the value param is of type `Vulnerability`

add_weakness(value)
    Adds a weakness to the weaknesses list property.

    **Note:** If `None` is passed in no value is added

    **Parameters** value – A `Weakness` object.
    **Raises:** ValueError if the value param is of type `Weakness`

description
    A single description about the contents or purpose of this object.
    **Default Value:** None

    **Note:** If this object has more than one description set, this will return the description with the lowest ordinality value.

    **Returns** An instance of `StructuredText`

find(id_)
    Searches the children of a `Entity` implementation for an object with an id_ property that matches id_.

**short_description**
A single short description about the contents or purpose of this object.

Default Value: None

**Note:** If this object has more than one short description set, this will return the description with the lowest ordinality value.

**Returns** An instance of *StructuredText*

**to_dict()**
Convert to a dict
Subclasses can override this function.

**Returns** Python dict with keys set from this Entity.

**to_json()**
Export an object as a JSON String.

**to_obj**(ns_info=None)
Convert to a GenerateDS binding object.
Subclasses can override this function.

**Returns** An instance of this Entity’s _binding_class with properties set from this Entity.

**class** stix.exploit_target.PotentialCOAs(coas=None, scope=None)
Bases: stix.common.related.GenericRelationshipList
A list of Potential_COA objects, defaults to empty array

**class** stix.exploit_target.RelatedExploitTargets(related_exploit_targets=None, scope=None)
Bases: stix.common.related.GenericRelationshipList
A list of RelatedExploitTargets objects, defaults to empty array

**Version:** 1.2.0.8

**stix.exploit_target.configuration Module**

**Overview**

The *stix.exploit_target.configuration* module captures the software configuration that causes a vulnerability in a system.

**Classes**

**class** stix.exploit_target.configuration.Configuration(description=None, short_description=None, cce_id=None)
Bases: stix.base.Entity
Implementation of STIX Configuration.

**Parameters**
• **cce_id** *(optional)* – Common Configuration Enumeration value as a string
• **description** *(optional)* – A string description.
• **short_description** *(optional)* – A string short description.

**add_description**( *description*)
Adds a description to the descriptions collection.
This is the same as calling “foo.descriptions.add(bar)”.

**add_short_description**( *description*)
Adds a description to the short_descriptions collection.
This is the same as calling “foo.short_descriptions.add(bar)”.

**description**
A single description about the contents or purpose of this object.
Default Value: None

**Note:** If this object has more than one description set, this will return the description with the lowest ordinality value.

**Returns**  An instance of *StructuredText*

**short_description**
A single short description about the contents or purpose of this object.
Default Value: None

**Note:** If this object has more than one short description set, this will return the description with the lowest ordinality value.

**Returns**  An instance of *StructuredText*

**Version:** 1.2.0.8

**stix.exploit_target.vulnerability Module**

**Overview**

The *stix.exploit_target.vulnerability* module captures the software version and specific bug that causes an exploitable condition.

**Classes**

**class** *stix.exploit_target.vulnerability.Vulnerability**( title=None,  
description=None,  
short_description=None)

**Bases:** *stix.base.Entity*

Implementation of STIX Vulnerability.

**Parameters**
• **title** *(optional)* – A string title.
• **description** *(optional)* – A string description.
• **short_description** *(optional)* – A string short description.

**add_description** *(description)*

Adds a description to the **descriptions** collection.

This is the same as calling “foo.descriptions.add(bar)”.

**add_short_description** *(description)*

Adds a description to the **short_descriptions** collection.

This is the same as calling “foo.short_descriptions.add(bar)”.

**description**

A single description about the contents or purpose of this object.

Default Value: None

**Note:** If this object has more than one description set, this will return the description with the lowest ordinality value.

**Returns** An instance of *StructuredText*

**short_description**

A single short description about the contents or purpose of this object.

Default Value: None

**Note:** If this object has more than one short description set, this will return the description with the lowest ordinality value.

**Returns** An instance of *StructuredText*

class **stix.exploit_target.vulnerability.CVSSVector**

Bases: **stix.base.Entity**

Common Vulnerability Scoring System object, representing its component measures

class **stix.exploit_target.vulnerability.AffectedSoftware** *(scope=None, *args)*

Bases: **stix.common.related.GenericRelationshipList**

Version: 1.2.0.8

**stix.exploit_target.weakness Module**

**Overview**

The **stix.exploit_target.weakness** module captures a given software weakness as enumerated by CWE
Classes

```python
class stix.exploit_target.weakness.Weakness(description=None, cwe_id=None):
    bases: stix.base.Entity

    Implementation of STIX Weakness.

    Parameters
    ----------
    cwe_id (optional) : string
        Common Weakness Enumeration value as a string
    description (optional) : string
        A string description.

    add_description(description)

        Adds a description to the descriptions collection.
        This is the same as calling "foo.descriptions.add(bar)".

    description

        A single description about the contents or purpose of this object.
        Default Value: None

    Note: If this object has more than one description set, this will return the description with the lowest ordinality value.

    Returns
    -------
    An instance of StructuredText
```

3.1.7 STIX Extensions

Modules located in the stix.extensions package

Version: 1.2.0.8

```python
stix.extensions.identity.ciq_identity_3_0 Module

Classes

```python
class stix.extensions.identity.ciq_identity_3_0.CIQIdentity3_0Instance(roles=None, specifcation=None):
    bases: stix.common.identity.Identity
```
```python
class stix.extensions.identity.ciq_identity_3_0.STIXCIQIdentity3_0 (party_name=None, languages=None, addresses=None, organisation_info=None, electronic_address_identifiers=None, free_text_lines=None, contact_numbers=None, nationalities=None)

Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.Address (free_text_address=None, country=None, administrative_area=None)

Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.AdministrativeArea (name_elements=None)

Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0._BaseNameElement (value=None)

Do not instantiate directly: use PersonNameElement or OrganisationNameElement

class stix.extensions.identity.ciq_identity_3_0.ContactNumber (contact_number_elements=None, communication_media_type=None)

Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.ContactNumberElement (value=None, type_=None)

Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.Country (name_elements=None)

Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.ElectronicAddressIdentifier (value=None, type_=None)

Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.FreeTextAddress (address_lines=None)

Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.FreeTextLine (value=None, type_=None)

Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.Language (value=None)

Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.NameElement (value=None, name_type=None, name_code=None, name_code_type=None)

Bases: stix.base.Entity
```

3.1. API Reference
class stix.extensions.identity.ciq_identity_3_0.NameLine(value=None, type_=None)
Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.OrganisationInfo(industry_type=None)
Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.OrganisationName(name_elements=None, subdivision_names=None, type_=None)
Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.OrganisationNameElement(value=None, element_type=None)
Bases: stix.extensions.identity.ciq_identity_3_0._BaseNameElement

class stix.extensions.identity.ciq_identity_3_0.PartyName(name_lines=None, person_names=None, organisation_names=None)
Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.PersonName(name_elements=None, type_=None)
Bases: stix.base.Entity

class stix.extensions.identity.ciq_identity_3_0.PersonNameElement(value=None, element_type=None)
Bases: stix.extensions.identity.ciq_identity_3_0._BaseNameElement

class stix.extensions.identity.ciq_identity_3_0.SubDivisionName(value=None, type_=None)
Bases: stix.base.Entity

Constants

stix.extensions.identity.ciq_identity_3_0.XML_NS_XPIL = ‘urn:oasis:names:tc:ciq:xpil:3’
str(object='') -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.identity.ciq_identity_3_0.XML_NS_XNL = ‘urn:oasis:names:tc:ciq:xnl:3’
str(object='') -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.identity.ciq_identity_3_0.XML_NS_XAL = ‘urn:oasis:names:tc:ciq:xal:3’
str(object='') -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.identity.ciq_identity_3_0.XML_NS_STIX_EXT = ‘http://stix.mitre.org/extensions/Identity#CIQIdentity3.0-1’
str(object='') -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

Version: 1.2.0.8
stix.extensions.malware.maec_4_1_malware Module

Classes

class stix.extensions.malware.maec_4_1_malware.MAECInstance(maec=None)
    Bases: stix.ttp.malware_instance.MalwareInstance

    The MAECInstance object provides an extension to the MalwareInstanceType which imports and leverages the
    MAEC 4.1 schema for structured characterization of Malware.

    This class extension is automatically registered by the MalwareInstanceFactory.

    **Warning:** Interacting with the maec field will fail if the maec library is not installed in your Python
    environment.

Version: 1.2.0.8

stix.extensions.marking.ais Module

STIX Extension for AIS Data Markings

Unlike the other marking extensions, the AIS marking extension is not loaded automatically, since AIS markings are
not a part of the published STIX 1.x specifications. They are included in python-stix because they’re common enough
that it is not worth creating a separate package.

If you are writing code that needs to parse AIS markings, make sure that your program imports this module before
beginning to parse any STIX documents:

```python
import stix.extensions.marking.ais
```

Classes

class stix.extensions.marking.ais.AISMarkingStructure(is_proprietary=None,
                                                     not_proprietary=None)
    Bases: stix.data_marking.MarkingStructure

Functions

stix.extensions.marking.ais.add_ais_marking(stix_package, proprietary, consent, color,
                                            **kwargs)

    This utility functions aids in the creation of an AIS marking and appends it to the provided STIX package.

    Parameters

    • **stix_package** – A stix.core.STIXPackage object.
    • **proprietary** – True if marking uses IsProprietary, False for NotProprietary.
    • **consent** – A string with one of the following values: “EVERYONE”, “NONE” or “USG”.
    • **color** – A string that corresponds to TLP values: “WHITE”, “GREEN” or “AMBER”.
    • **kwargs** – Six required keyword arguments that are used to create a CIQ identity ob-
        ject. These are: country_name_code, country_name_code_type, admin_area_name_code,
        admin_area_name_code_type, organisation_name, industry_type.
Raises ValueError – When keyword arguments are missing. User did not supply correct values for: proprietary, color and consent.

Note: The following line is required to register the AIS extension:

```python
>>> import stix.extensions.marking.ais
```

Any Markings under STIX Header will be removed. Please follow the guidelines for AIS.

The industry_type keyword argument accepts: a list of string based on defined sectors, a pipe-delimited string of sectors, or a single sector.

Examples

Applying AIS Markings

The STIX specification allows data markings to be applied to any combination of attributes and elements that can be described by XPath. That being said, the Automated Indicator Sharing (AIS) capability requires those markings controlled structure to select all nodes and attributes `//node() | //@*`. All required fields to create a valid AIS Markings are provided through the `add_ais_marking` function.

```python
# python-stix imports
import stix
from stix.core import STIXPackage
from stix.extensions.marking.ais import add_ais_marking,
                                 COMMUNICATIONS_SECTOR,
                                 INFORMATION_TECHNOLOGY_SECTOR)
from stix.indicator import Indicator

# Create new STIX Package
stix_package = STIXPackage()

# Create new Indicator
indicator = Indicator(title='My Indicator Example',
                      description='Example using AIS')

# Add indicator to our STIX Package
stix_package.add_indicator(indicator)

# Create AIS Marking with CIQ Identity and attach it to STIX Header.
add_ais_marking(stix_package, False, 'EVERYONE', 'GREEN',
                country_name_code='US',
                country_name_code_type='ISO 3166-1 alpha-2',
                admin_area_name_code='US-VA',
                admin_area_name_code_type='ISO 3166-2',
                organisation_name='Example Corporation',
                industry_type=[INFORMATION_TECHNOLOGY_SECTOR, COMMUNICATIONS_SECTOR])

# Print the XML.
print stix_package.to_xml()

# Print the JSON.
print stix_package.to_json()
```

This corresponds to the XML result:
<stix:STIX_Package
  xmlns:AIS="http://www.us-cert.gov/STIXMarkingStructure#AISConsentMarking-2"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xpil="urn:oasis:names:tc:ciq:xpil:3"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xal="urn:oasis:names:tc:ciq:xal:3"
  xmlns:stix="http://stix.mitre.org/stix-1"
  xmlns:indicator="http://stix.mitre.org/Indicator-2"
  xmlns:marking="http://data-marking.mitre.org/Marking-1"
  xmlns:stixCommon="http://stix.mitre.org/common-1"
  xmlns:example="http://example.com"
  xmlns:stix-ciqidentity="http://stix.mitre.org/extensions/Identity#CIQIdentity3.0-1"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  id="example:Package-73ac199c-9dd8-4d8d-a37e-8ac40fc65ccf" version="1.2">
  <stix:STIX_Header>
    <stix:Handling>
      <marking:Marking>
        <marking:Controlled_Structure>
          <AIS:Not_Proprietary
            CISA_Proprietary="false">
            <AIS:AISConsent
              consent="EVERYONE"/>
            <AIS:TLPMarking
              color="GREEN"/>
          </AIS:Not_Proprietary>
        </marking:Controlled_Structure>
      </marking:Marking>
    </stix:Handling>
  </stix:STIX_Header>
  <stix:Indicators>
    <stix:Indicator
      id="example:indicator-eab71e49-e982-4874-a057-e75e51a76009"
      timestamp="2017-09-21T13:28:47.467000+00:00"
      xsi:type='indicator:IndicatorType'>
      <indicator:Title>My Indicator Example</indicator:Title>
      <indicator:Description>Example using AIS</indicator:Description>
    </stix:Indicator>
  </stix:Indicators>
</stix:STIX_Package>
The following corresponds to the JSON result:

```json
{
    "stix_header": {
        "handling": {
            "controlled_structure": "//node() | //@*",
            "information_source": {
                "identity": {
                    "xsi:type": "stix-ciqidentity:CIQIdentity3.0InstanceType",
                    "specification": {
                        "organisation_info": {
                            "industry_type": "Information Technology Sector|Communications Sector",
                            "party_name": {
                                "organisation_names": [
                                    {
                                        "name_elements": [
                                            {
                                                "value": "Example Corporation"
                                            }
                                        ]
                                    }
                                ],
                                "addresses": [
                                    {
                                        "country": {
                                            "name_elements": [n
                                            {
                                                "name_code_type": "ISO 3166-1 alpha-2",
                                                "name_code": "US"
                                            }
                                        ],
                                        "administrative_area": {
                                            "name_elements": [
                                                {
                                                    "name_code_type": "ISO 3166-2",
                                                    "name_code": "US-VA"
                                                }
                                            ]
                                        }
                                    }
                                ]
                            }
                        }
                    }
                },
                "marking_structures": [
                    {
                        "xsi:type": "AIS:AISMarkingStructure",
                        "not_proprietary": {
                            "tlp_marking": {
                                "color": "GREEN"
                            }
                        },
                        "ais_consent": {
                            "consent": "EVERYONE"
                        },
                        "cisa propriétaire": "false"
                    }
                ]
            }
        }
    }
}
```
Parsing AIS Markings

Using the same example used for Applying AIS Markings. This would be how a consumer of AIS would parse the data.

```python
# python-stix imports
import stix
from stix.core import STIXPackage
import stix.extensions.marking.ais  # Register the AIS markings

# Parse STIX Package
stix_package = STIXPackage.from_xml("stix_input.xml")
# stix_package = STIXPackage.from_json("stix_input.json")

# Print all indicators
for indicator in stix_package.indicators:
    print(indicator)

# Extract markings from STIX Header
markings = stix_package.stix_header.handling

# Print all markings contained in the STIX Header
for marking in markings:
    print(marking)
    print(marking.marking_structures)
    print("--------------MARKING CONTENT------------")
    ais_struct = marking.marking_structures[0]
    print("OBJ: %s" % ais_struct)
    print("NotProprietary OBJ: %s" % ais_struct.not_proprietary)
    print("CISA_Proprietary: %s" % ais_struct.not_proprietary.cisa_proprietary)
    print("Consent: %s" % ais_struct.not_proprietary.ais_consent.consent)
    print("TLP color: %s" % ais_struct.not_proprietary.tlp_marking.color)
    print("--------------INFORMATION SOURCE-------------")
    identity = marking.information_source.identity.specification
    print("OBJ: %s" % identity)
    print("Organization Name: %s" % identity.party_name.organisation_names[0].name_elements[0].value)
    print("Country: %s" % identity.addresses[0].country.name_elements[0].name_code)
    print("Country code type: %s" % identity.addresses[0].country.name_elements[0].name_code_type)
```
print("Administrative area: %s" % identity.addresses[0].administrative_area.name_elements[0].name_code)
print("Administrative area code type: %s" % identity.addresses[0].administrative_area.name_elements[0].name_code_type)
print("Industry Type: %s" % identity.organisation_info.industry_type)

>>> <stix.indicator.indicator.Indicator object at 0x...>
>>> <stix.data_marking.MarkingSpecification object at 0x...>
>>> [<stix.extensions.marking.ais.AISMarkingStructure object at 0x...>, ...]
>>> ----------MARKING CONTENT----------
>>> OBJ: <stix.extensions.marking.ais.AISMarkingStructure object at 0x...>
>>> NotProprietary OBJ: <stix.extensions.marking.ais.NotProprietary object at 0x...>
>>> CISA_Proprietary: False
>>> Consent: EVERYONE
>>> TLP color: GREEN
>>> ----------INFORMATION SOURCE----------
>>> OBJ: <stix.extensions.identity.ciq_identity_3_0.STIXCIQIdentity3_0 object at 0x...>
>>> Organization Name: Example Corporation
>>> Country: US
>>> Country code type: ISO 3166-1 alpha-2
>>> Administrative area: US-VA
>>> Administrative area code type: ISO 3166-2
>>> Industry Type: Information Technology Sector|Communications Sector

Constants

The following constants can be used for the industry_type keyword argument to add_ais_marking:

```
stix.extensions.marking.ais.CHEMICAL_SECTOR = 'Chemical Sector'
str(object='') -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.COMMERCIAL_FACILITIES_SECTOR = 'Commercial Facilities Sector'
str(object='') -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.COMMUNICATIONS_SECTOR = 'Communications Sector'
str(object='') -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.CRITICAL_MANUFACTURING_SECTOR = 'Critical Manufacturing Sector'
str(object='') -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.DAMS_SECTOR = 'Dams Sector'
str(object='') -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.DEFENSE_INDUSTRIAL_BASE_SECTOR = 'Defense Industrial Base Sector'
str(object='') -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.EMERGENCY_SERVICES_SECTOR = 'Emergency Services Sector'
str(object='') -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.
```
stix.extensions.marking.ais.ENERGY_SECTOR = ‘Energy Sector’
    str(object=’’) -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.FINANCIAL_SERVICES_SECTOR = ‘Financial Services Sector’
    str(object=’’) -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.FOOD_AND_AGRICULTURE_SECTOR = ‘Food and Agriculture Sector’
    str(object=’’) -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.GOVERNMENT_FACILITIES_SECTOR = ‘Government Facilities Sector’
    str(object=’’) -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.HEALTH_CARE_AND_PUBLIC_HEALTH_SECTOR = ‘Healthcare and Public Health Sector’
    str(object=’’) -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.INFORMATION_TECHNOLOGY_SECTOR = ‘Information Technology Sector’
    str(object=’’) -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.NUCLEAR_REACTORS_MATERIALS_AND_WASTE_SECTOR = ‘Nuclear Reactors, Materials, and Waste Sector’
    str(object=’’) -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.OTHER = ‘Other’
    str(object=’’) -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.TRANSPORTATION_SYSTEMS_SECTOR = ‘Transportation Systems Sector’
    str(object=’’) -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

stix.extensions.marking.ais.WATER_AND_WASTEWATER_SYSTEMS_SECTOR = ‘Water and Wastewater Systems Sector’
    str(object=’’) -> string
    Return a nice string representation of the object. If the argument is a string, the return value is the same object.

Version: 1.2.0.8

stix.extensions.marking.simple_marking Module

Classes

class stix.extensions.marking.simple_marking.SimpleMarkingStructure (statement=None)
    Bases: stix.data_marking.MarkingStructure

Version: 1.2.0.8
stix.extensions.marking.terms_of_use_marking Module

Classes

class stix.extensions.marking.terms_of_use_marking.TermsOfUseMarkingStructure (terms_of_use=None)
    Bases: stix.data_marking.MarkingStructure

Version: 1.2.0.8

stix.extensions.marking.tlp Module

Classes

class stix.extensions.marking.tlp.TLPMarkingStructure (color=None)
    Bases: stix.data_marking.MarkingStructure

Version: 1.2.0.8

stix.extensions.structured_coa.generic_structured_coa Module

Classes

class stix.extensions.structured_coa.generic_structured_coa.GenericStructuredCOA (id_=None,
idref=None)
    Bases: stix.coa.structured_coa._BaseStructuredCOA

    add_description (description)
    Adds a description to the descriptions collection.
    This is the same as calling "foo.descriptions.add(bar)".

description
    A single description about the contents or purpose of this object.
    Default Value: None

    Note: If this object has more than one description set, this will return the description with the lowest
    ordinality value.

    Returns An instance of StructuredText

Version: 1.2.0.8

stix.extensions.test_mechanism.generic_test_mechanism Module

Classes

class stix.extensions.test_mechanism.generic_test_mechanism.GenericTestMechanism (id_=None,
idref=None)
    Bases: stix.indicator.test_mechanism._BaseTestMechanism
**add_description** *(description)*

Adds a description to the `descriptions` collection.

This is the same as calling “foo.descriptions.add(bar)”.

**description**

A single description about the contents or purpose of this object.

Default Value: None

**Note:** If this object has more than one description set, this will return the description with the lowest ordinality value.

**Returns** An instance of `StructuredText`

Version: 1.2.0.8

```python
stix.extensions.test_mechanism.open_ioc_2010_test_mechanism Module

Classes

class stix.extensions.test_mechanism.open_ioc_2010_test_mechanism.OpenIOCTestMechanism(id_=None,
idref=None):
    Bases: stix.indicator.test_mechanism._BaseTestMechanism

Version: 1.2.0.8

stix.extensions.test_mechanism.snort_test_mechanism Module

Classes

class stix.extensions.test_mechanism.snort_test_mechanism.SnortTestMechanism(id_=None,
idref=None):
    Bases: stix.indicator.test_mechanism._BaseTestMechanism

Version: 1.2.0.8

stix.extensions.test_mechanism.yara_test_mechanism Module

Classes

class stix.extensions.test_mechanism.yara_test_mechanism.YaraTestMechanism(id_=None,
idref=None):
    Bases: stix.indicator.test_mechanism._BaseTestMechanism

Version: 1.2.0.8

3.1.8 STIX Incident

Modules located in the *stix.incident* package

Version: 1.2.0.8
The `stix.incident` module implements `Incident`.

Incidents are discrete instances of Indicators affecting an organization along with information discovered or decided during an incident response investigation.

Documentation Resources

- Incident Data Model

Classes

```python
class stix.incident.Incident (id_=None, idref=None, timestamp=None, title=None, description=None, short_description=None)
Bases: stix.base.BaseCoreComponent
```

Implementation of the STIX Incident.

Parameters

- `id` *(optional)* – An identifier. If None, a value will be generated via `mixbox.idgen.create_id()`. If set, this will unset the `idref` property.
- `idref` *(optional)* – An identifier reference. If set this will unset the `id_` property.
- `timestamp` *(optional)* – A timestamp value. Can be an instance of `datetime.datetime` or `str`.
- `description` – A description of the purpose or intent of this object.
- `short_description` – A short description of the intent or purpose of this object.
- `title` – The title of this object.

`add_affected_asset(v)`

Add a `AffectedAsset` object to the `affected_assets` collection.

`add_category(category)`

Add a `VocabString` object to the `categories` collection.

If `category` is a string, an attempt will be made to convert it into an instance of `IncidentCategory`.

`add_coa_requested(value)`

Add a `COARequested` object to the `coas_requested` collection.

`add_coa_taken(value)`

Add a `COATaken` object to the `coas_taken` collection.

`add_coordinator(value)`

Add a `InformationSource` object to the `coordinators` collection.

`add_description(description)`

Add a description to the `descriptions` collection.

This is the same as calling “foo.descriptions.add(bar)”.
add_discovery_method (value)
    Adds a VocabString object to the discovery_methods collection.
    If value is a string, an attempt will be made to convert it to an instance of DiscoveryMethod.

add_external_id (value)
    Adds an ExternalID object to the external_ids collection.

add_intended_effect (value)
    Adds a Statement object to the intended_effects collection.
    If value is a string, an attempt will be made to convert it into an instance of Statement.

add_related_indicator (value)
    Adds an Related Indicator to the related_indicators list property of this Incident.
    The indicator parameter must be an instance of RelatedIndicator or Indicator.
    If the indicator parameter is None, no item will be added to the related_indicators list property.
    Calling this method is the same as calling append() on the related_indicators property.
    See also:
    The RelatedIndicators documentation.

    Note: If the indicator parameter is not an instance of RelatedIndicator an attempt will be made to convert it to one.

Parameters indicator – An instance of Indicator or RelatedIndicator.
Raise ValueError – If the indicator parameter cannot be converted into an instance of RelatedIndicator

add_related_observable (value)
    Adds a Related Observable to the related_observables list property of this Incident.
    The observable parameter must be an instance of RelatedObservable or Observable.
    If the observable parameter is None, no item will be added to the related_observables list property.
    Calling this method is the same as calling append() on the related_observables property.
    See also:
    The RelatedObservables documentation.

    Note: If the observable parameter is not an instance of RelatedObservable an attempt will be made to convert it to one.

Parameters observable – An instance of Observable or RelatedObservable.
Raise ValueError – If the value parameter cannot be converted into an instance of RelatedObservable

add_responder (value)
    Adds an InformationSource object to the responders collection.
add_short_description(description)

Adds a description to the short_descriptions collection.
This is the same as calling “foo.short_descriptions.add(bar)”.

add_victim(victim)

Adds a IdentityType value to the victims collection.

description

A single description about the contents or purpose of this object.
Default Value: None

Note: If this object has more than one description set, this will return the description with the lowest
ordinality value.

Returns An instance of StructuredText

find(id_)

Searches the children of a Entity implementation for an object with an id_ property that matches id_.

short_description

A single short description about the contents or purpose of this object.
Default Value: None

Note: If this object has more than one short description set, this will return the description with the lowest
ordinality value.

Returns An instance of StructuredText

to_dict()

Convert to a dict

Subclasses can override this function.

Returns Python dict with keys set from this Entity.

to_json()

Export an object as a JSON String.

to_obj(ns_info=None)

Convert to a GenerateDS binding object.

Subclasses can override this function.

Returns An instance of this Entity’s _binding_class with properties set from this Entity.
class stix.incident.RelatedIncidents (scope=None, *args)
    Bases: stix.common.related.GenericRelationshipList

Version: 1.2.0.8

stix.incident.affected_asset Module

Classes

class stix.incident.affected_asset.AffectedAsset
    Bases: stix.base.Entity

    add_description (description)
        Adds a description to the descriptions collection.
        This is the same as calling “foo.descriptions.add(bar)”.

description
    A single description about the contents or purpose of this object.
    Default Value: None

    Note: If this object has more than one description set, this will return the description with the lowest ordinality value.

    Returns  An instance of StructuredText

class stix.incident.affected_asset.AssetType (value=None, count_affected=None)
    Bases: stix.common.vocabs.VocabString

    is_plain ()
        Override VocabString.is_plain()

Version: 1.2.0.8

stix.incident.coa Module

Classes

class stix.incident.coa.COADeclared (course_of_action=None)
    Bases: stix.base.Entity

class stix.incident.coa.COALevel (course_of_action=None)
    Bases: stix.base.Entity

class stix.incident.coa.COATrade (course_of_action=None)
    Bases: stix.base.Entity

class stix.incident.coa.COATime (start=None, end=None)
    Bases: stix.base.Entity

Version: 1.2.0.8

stix.incident.contributors Module
Classes

class stix.incident.contributors.Contributors(*args)
    Bases: stix.base.EntityList

Version: 1.2.0.8

stix.incident.direct_impact_summary Module

Classes

class stix.incident.direct_impact_summary.DirectImpactSummary
    Bases: stix.base.Entity

Version: 1.2.0.8

stix.incident.external_id Module

Classes

class stix.incident.external_id.ExternalID(value=None, source=None)
    Bases: stix.base.Entity

Version: 1.2.0.8

stix.incident.history Module

Classes

class stix.incident.history.History(*args)
    Bases: stix.base.EntityList

class stix.incident.history.HistoryItem
    Bases: stix.base.Entity

class stix.incident.history.JournalEntry(value=None)
    Bases: stix.base.Entity

Version: 1.2.0.8

stix.incident.impact_assessment Module

Classes

class stix.incident.impact_assessment.ImpactAssessment
    Bases: stix.base.Entity

Version: 1.2.0.8
stix.incident.indirect_impact_summary Module

Classes

class stix.incident.indirect_impact_summary.IndirectImpactSummary
    Bases: stix.base.Entity
Version: 1.2.0.8

stix.incident.loss_estimation Module

Classes

class stix.incident.loss_estimation.LossEstimation
    Bases: stix.base.Entity
Version: 1.2.0.8

stix.incident.property_affected Module

Classes

class stix.incident.property_affected.PropertyAffected
    description_of_effect
        A StructuredTextList object, containing descriptions about the purpose or intent of this object.
        Iterating over this object will yield its contents sorted by their ordinality value.
        Default Value: Empty StructuredTextList object.

    Returns  An instance of StructuredTextList

class stix.incident.property_affected.NonPublicDataCompromised (value=None, data_encrypted=None)
    Bases: stix.common.vocabs.VocabString
Version: 1.2.0.8

stix.incident.time Module

3.1. API Reference 75
Classes

```python
class stix.incident.time.Time (first_malicious_action=None, initial_compromise=None,
    first_data_exfiltration=None, incident_discovery=None,
    incident_opened=None, containment_achieved=None,
    restoration_achieved=None, incident_reported=None, incident_closed=None)
```

Bases: `stix.base.Entity`

Version: 1.2.0.8

**stix.incident.total_loss_estimation** Module

Classes

```python
class stix.incident.total_loss_estimation.TotalLossEstimation
```

Bases: `stix.base.Entity`

### 3.1.9 STIX Indicator

Modules located in the `stix.indicator` package

Version: 1.2.0.8

**stix.indicator.indicator** Module

Overview

The `stix.indicator.indicator` module implements `IndicatorType` STIX Language construct. The `IndicatorType` characterizes a cyber threat indicator made up of a pattern identifying certain observable conditions as well as contextual information about the patterns meaning, how and when it should be acted on, etc.

Documentation Resources

- Indicator Data Model
- Indicator Idioms

Classes

```python
class stix.indicator.indicator.Indicator (id_=None, idref=None, times tamp=None, title=None, description=None, short_description=None)
```

Bases: `stix.base.BaseCoreComponent`

Implementation of the STIX Indicator.

Parameters

- `id (optional)` – An identifier. If `None`, a value will be generated via `mixbox.idgen.create_id()`. If set, this will unset the `idref` property.
- `idref (optional)` – An identifier reference. If set this will unset the `id_` property.
• **title** *(optional)* – A string title.
• **timestamp** *(optional)* – A timestamp value. Can be an instance of `datetime.datetime` or `str`.
• **description** *(optional)* – A string description.
• **short_description** *(optional)* – A string short description.

**add_alternative_id**(value)
Adds an alternative id to the `alternative_id` list property.

**Note:** If `None` is passed in no value is added to the `alternative_id` list property.

**Parameters**
- **value** – An identifier value.

**add_description**(description)
Adds a description to the `descriptions` collection.

This is the same as calling “foo.descriptions.add(bar)”.

**add_indicated_ttp**(v)
Adds an Indicated TTP to the `indicated_ttps` list property of this `Indicator`.

The `v` parameter must be an instance of `stix.common.related.RelatedTTP` or `stix.ttp.TTP`.
If the `v` parameter is `None`, no item will be added to the `indicated_ttps` list property.

**Note:** If the `v` parameter is not an instance of `stix.common.related.RelatedTTP` an attempt will be made to convert it to one.

**Parameters**
- **v** – An instance of `stix.common.related.RelatedTTP` or `stix.ttp.TTP`

**Raises** `ValueError` – If the `v` parameter cannot be converted into an instance of `stix.common.related.RelatedTTP`

**add_indicator_type**(value)
Adds a value to the `indicator_types` list property.

The `value` parameter can be a `str` or an instance of `stix.common.vocabs.VocabString`.

**Note:** If the `value` parameter is a `str` instance, an attempt will be made to convert it into an instance of `stix.common.vocabs.IndicatorType`

**Parameters**
- **value** – An instance of `stix.common.vocabs.VocabString` or `str`

**Raise** `ValueError` – If the `value` param is a `str` instance that cannot be converted into an instance of `stix.common.vocabs.IndicatorType`

**add_kill_chain_phase**(value)
Add a new Kill Chain Phase reference to this Indicator.
Parameters **value** – a :class:`.KillChainPhase` or a :class:`str` representing the phase_id of. Note that you if you are defining a custom Kill Chain, you need to add it to the STIX package separately.

**add_object** *(object_)*

Adds a python-cybox Object instance to the observables list property.

This is the same as calling :meth:`Indicator.add_observable`

**Note:** If the :meth:`object` param is not an instance of :class:`cybox.core.Object` an attempt will be made to to convert it into one before wrapping it in a :class:`cybox.core.Observable` layer.

Parameters **object** – An instance of :class:`cybox.core.Object` or an object that can be converted into an instance of :class:`cybox.core.Observable`


**add_observable** *(observable)*

Adds an observable to the observable property of the :class:`Indicator`.

If the :meth:`observable` parameter is None, no item will be added to the observable property.

**Note:** The STIX Language dictates that an :class:`Indicator` can have only one Observable under it. Because of this, when a user adds another Observable a new, empty Observable will be crated and append the existing and new observable using the ObservableComposition property. To access the top level Observable can be achieved by the observable property. By default, the operator of the composition layer will be set to "OR". The operator value can be changed via the observable_composition_operator property.

Setting observable or observables with re-initialize the property and lose all Observable in the composition layer.

Parameters **observable** – An instance of :class:`cybox.core.Observable` or an object type that can be converted into one.


**add_related_campaign** *(value)*

Adds a Related Campaign to this Indicator.

The :meth:`value` parameter must be an instance of :class:`RelatedCampaignRef` or :class:`CampaignRef`.

If the :meth:`value` parameter is None, no item will be added to the related_campaigns collection.

Calling this method is the same as calling :meth:`append()` on the related_campaigns property.

**See also:**

The :class:`RelatedCampaignRef` documentation.

**Note:** If the :meth:`value` parameter is not an instance of :class:`RelatedCampaignRef` an attempt will be made to convert it to one.
Parameters `value` – An instance of `RelatedCampaignRef` or `Campaign`.

Raises `ValueError` – If the `value` parameter cannot be converted into an instance of `RelatedCampaignRef`.

### add_related_indicator( indicator )

Adds an Related Indicator to the `related_indicators` list property of this `Indicator`.

The `indicator` parameter must be an instance of `stix.common.related.RelatedIndicator` or `Indicator`.

If the `indicator` parameter is `None`, no item will be added to the `related_indicators` list property.

Calling this method is the same as calling `append()` on the `related_indicators` property.

See also:

The `RelatedIndicators` documentation.

### add_short_description( description )

Adds a description to the `short_descriptions` collection.

This is the same as calling “foo.short_descriptions.add(bar)”.

### add_test_mechanism( tm )

Adds an Test Mechanism to the `test_mechanisms` list property of this `Indicator`.

The `tm` parameter must be an instance of a `stix.indicator.test_mechanism._BaseTestMechanism` implementation.

If the `tm` parameter is `None`, no item will be added to the `test_mechanisms` list property.

See also:

Test Mechanism implementations are found under the `stix.extensions.test_mechanism` package.

Parameters `tm` – An instance of a `stix.indicator.test_mechanism._BaseTestMechanism` implementation.

Raises `ValueError` – If the `tm` parameter is not an instance of `stix.indicator.test_mechanism._BaseTestMechanism`.

### add_valid_time_position( value )

Adds an valid time position to the `valid_time_positions` property list.

If `value` is `None`, no item is added to the `value_time_positions` list.

Parameters `value` – An instance of `stix.indicator.valid_time.ValidTime`.

Raises `ValueError` – If the `value` argument is not an instance of `stix.indicator.valid_time.ValidTime`. 

3.1. API Reference 79
**description**
A single description about the contents or purpose of this object.

Default Value: None

**Note:** If this object has more than one description set, this will return the description with the lowest ordinality value.

**Returns** An instance of `StructuredText`

**find(id_)**
Searches the children of a `Entity` implementation for an object with an `id_` property that matches `id_`.

**get_produced_time()**
Gets the produced time for this `Indicator`.

This is the same as calling `produced_time = indicator.producer.time.produced_time`.

**Returns** None or an instance of `cybox.common.DateTimeWithPrecision`.

**get_received_time()**
Gets the received time for this `Indicator`.

This is the same as calling `received_time = indicator.producer.time.received_time`.

**Returns** None or an instance of `cybox.common.DateTimeWithPrecision`.

**observables**
A list of `cybox.core.Observable` instances. This can be set to a single object instance or a list of objects.

**Note:** If only one Observable is set, this property will return a list with the `observable` property.

If multiple `cybox.core.Observable` this property will return Observables under the `cybox.core.ObservableComposition`.

Access to the top level `cybox.core.Observable` is made via `observable` property.

Default Value: Empty list.

**Returns** A list of `cybox.core.Observable` instances.

**set_produced_time**(produced_time)
Sets the `produced_time` property of the `producer` property instance for `produced_time`.

This is the same as calling `indicator.producer.time.produced_time = produced_time`.

The `produced_time` parameter must be an instance of `str`, `datetime.datetime`, or `cybox.common.DateTimeWithPrecision`.

**Note:** If `produced_time` is a `str` or `datetime.datetime` instance an attempt will be made to convert it into an instance of `cybox.common.DateTimeWithPrecision`. 
Parameters **produced_time** – An instance of str, datetime.datetime, or cybox.common.DateTimeWithPrecision.

**set_producer_identity**(identity)
Sets the name of the producer of this indicator.

This is the same as calling indicator.producer.identity.name = identity.

If the producer property is None, it will be initialized to an instance of stix.common.information_source.InformationSource.

If the identity property of the producer instance is None, it will be initialized to an instance of stix.common.identity.Identity.

Note: if the identity parameter is not an instance stix.common.identity.Identity an attempt will be made to convert it to one.

Parameters **identity** – An instance of str or stix.common.identity.Identity.

**set_received_time**(received_time)
Sets the received time for this Indicator.

This is the same as calling indicator.producer.time.produced_time = produced_time.

The received_time parameter must be an instance of str, datetime.datetime, or cybox.common.DateTimeWithPrecision.

Parameters **received_time** – An instance of str, datetime.datetime, or cybox.common.DateTimeWithPrecision.

Note: If received_time is a str or datetime.datetime instance an attempt will be made to convert it into an instance of cybox.common.DateTimeWithPrecision.

**short_description**
A single short description about the contents or purpose of this object.

Default Value: None

Note: If this object has more than one short description set, this will return the description with the lowest ordinality value.

**Returns** An instance of StructuredText

**to_dict()**
Convert to a dict

Subclasses can override this function.

**Returns** Python dict with keys set from this Entity.

**to_json()**
Export an object as a JSON String.
to_obj(ns_info=None)
Convert to a GenerateDS binding object.

Subclasses can override this function.

Returns An instance of this Entity's _binding_class with properties set from this Entity.

class stix.indicator.indicator.CompositeIndicatorExpression(operator='OR', *args)
Bases: mixbox.entities.EntityList

Implementation of the STIX CompositeIndicatorExpressionType.

The CompositeIndicatorExpression class implements methods found on collections.MutableSequence and as such can be interacted with as a list (e.g., append()).

Note: The append() method can only accept instances of Indicator.

Examples

Add a Indicator instance to an instance of CompositeIndicatorExpression:

```python
>>> i = Indicator()
>>> comp = CompositeIndicatorExpression()
>>> comp.append(i)
```

Create a CompositeIndicatorExpression from a list of Indicator instances using *args argument list:

```python
>>> list_indicators = [Indicator() for i in xrange(10)]
>>> comp = CompositeIndicatorExpression(CompositeIndicatorExpression.OP_OR, *list_indicators)
>>> len(comp)
10
```

Parameters

- **operator**(str, optional) – The logical composition operator. Must be "AND" or "OR".
- ***args** – Variable length argument list of Indicator instances.

OP_AND

str

String "AND"

OP_OR

str

String "OR"

OPERATORS
tuple

Tuple of allowed operator values.

operator

str

The logical composition operator. Must be "AND" or "OR".
to_obj \( (\text{ns\_info} = \text{None}) \)

Convert to a GenerateDS binding object.

Subclasses can override this function.

**Returns**

An instance of this Entity’s _binding_class with properties set from this Entity.

```python
class stix.indicator.indicator.RelatedIndicators (related_indicators=None, scope=None)
Bases: stix.common.related.GenericRelationshipList
```

The RelatedIndicators class provides functionality for adding stix.common.related.RelatedIndicator instances to an Indicator instance.

The RelatedIndicators class implements methods found on collections.MutableSequence and as such can be interacted with as a list (e.g., append()).

The append() method can accept instances of stix.common.related.RelatedIndicator or Indicator as an argument.

**Examples**

Append an instance of Indicator to the Indicator.related_indicators property. The instance of Indicator will be wrapped in an instance of stix.common.related.RelatedIndicator:

```python
>>> related = Indicator()
>>> parent_indicator = Indicator()
>>> parent_indicator.related_indicators.append(related)
>>> print(type(indicator.related_indicators[0]))
<class 'stix.common.related.RelatedIndicator'>
```

Iterate over the related_indicators property of an Indicator instance and print the ids of each underlying Indicator instance:

```python
>>> for related in indicator.related_indicators:
...     print(related.item.id_)
```

**Parameters**

- **related_indicators** *(list, optional)* – A list of Indicator or stix.common.related.RelatedIndicator instances.

- **scope** *(str, optional)* – The scope of the items. Can be set to "inclusive" or "exclusive". See stix.common.related.GenericRelationshipList documentation for more information.

```python
scope
str
```

The scope of the items. Can be set to "inclusive" or "exclusive". See stix.common.related.GenericRelationshipList documentation for more information.

```python
to_obj \( (\text{ns\_info} = \text{None}) \)
```

Convert to a GenerateDS binding object.

Subclasses can override this function.
**class** stix.indicator.indicator.RelatedCampaignRefs((related_campaign_refs=None, scope=None))

Bases: stix.common.related.GenericRelationshipList

**to_obj**(ns_info=None)

Convert to a GenerateDS binding object.

Subclasses can override this function.

**Returns** An instance of this Entity’s _binding_class with properties set from this Entity.

---

**class** stix.indicator.indicator.SuggestedCOAs((suggested_coas=None, scope=None))

Bases: stix.common.related.GenericRelationshipList

The SuggestedCOAs class provides functionality for adding stix.common.related.RelatedCOA instances to an Indicator instance.

The SuggestedCOAs class implements methods found on collections.MutableSequence and as such can be interacted with as a list (e.g., append()).

The append() method can accept instances of stix.common.related.RelatedCOA or stix.coa.CourseOfAction as an argument.

**Note:** Calling append() with an instance of stix.coa.CourseOfAction will wrap that instance in a stix.common.related.RelatedCOA layer, with the item set to the stix.coa.CourseOfAction instance.

---

**Examples**

Append an instance of stix.coa.CourseOfAction to the Indicator.suggested_coas property. The instance of stix.coa.CourseOfAction will be wrapped in an instance of stix.common.related.RelatedCOA.

```python
>>> coa = CourseOfAction()
>>> indicator = Indicator()
>>> indicator.suggested_coas.append(coa)
>>> print(type(indicator.suggested_coas[0]))
<class 'stix.common.related.RelatedCOA'>
```

Iterate over the suggested_coas property of an Indicator instance and print the ids of each underlying stix.coa.CourseOfAction instance.

```python
>>> for related_coa in indicator.suggested_coas:
...    print(related_coa.item.id_)
```

**Parameters**

- **suggested_coas** *(list)* – A list of stix.coa.CourseOfAction or stix.common.related.RelatedCOA instances.

- **scope** *(str)* – The scope of the items. Can be set to "inclusive" or "exclusive". See stix.common.related.GenericRelationshipList documentation for more information.

---

---
The scope of the items. Can be set to "inclusive" or "exclusive". See
`stix.common.related.GenericRelationshipList` documentation for more information.

def to_obj(ns_info=None):
    Convert to a GenerateDS binding object.

    Subclasses can override this function.

    **Returns**  An instance of this Entity's _binding_class with properties set from this Entity.

class stix.indicator.indicator.IndicatorTypes(*args):
    Bases: stix.base.TypedList

    A `stix.common.vocabs.VocabString` collection which defaults to
    `stix.common.vocabs.IndicatorType`. This class implements methods found on
    `collections.MutableSequence` and as such can be interacted with like a list.

    **Note:** The `append()` method can accept `str` or `stix.common.vocabs.VocabString`
    instances. If a `str` instance is passed in, an attempt will be made to convert it to an instance of
    `stix.common.vocabs.IndicatorType`.

Examples

Add an instance of `stix.common.vocabs.IndicatorType`:

```python
>>> from stix.common.vocabs importIndicatorType
>>> itypes = IndicatorTypes()
>>> type_ = IndicatorType(IndicatorType.TERM_IP_WATCHLIST)
>>> itypes.append(type_)
>>> print(len(itypes))
1
```

Add a string value:

```python
>>> from stix.common.vocabs import IndicatorType
>>> itypes = IndicatorTypes()
>>> type(IndicatorType.TERM_IP_WATCHLIST)
<type 'str'>
>>> itypes.append(IndicatorType.TERM_IP_WATCHLIST)
>>> print(len(itypes))
1
```

**Parameters** *args – Variable length argument list of strings or
`stix.common.vocabs.VocabString` instances.

Version: 1.2.0.8

**stix.indicator.sightings Module**

**Classes**

class stix.indicator.sightings.Sighting(timestamp=None, timestamp_precision=None, description=None):
    Bases: stix.base.Entity
add_description(description)
    Adds a description to the descriptions collection.
    This is the same as calling “foo.descriptions.add(bar)”.

description
    A single description about the contents or purpose of this object.
    Default Value: None

Note: If this object has more than one description set, this will return the description with the lowest ordinality value.

Returns An instance of StructuredText

class stix.indicator.sightings.Sightings(sightings_count=None, *args)
    Bases: stix.base.EntityList

class stix.indicator.sightings.RelatedObservables(scope=None, *args)
    Bases: stix.common.related.GenericRelationshipList

Version: 1.2.0.8

stix.indicator.test_mechanism Module

Classes

class stix.indicator.test_mechanism._BaseTestMechanism(id_=None, idref=None)
    Bases: stix.base.Entity

Functions

stix.indicator.test_mechanism.add_extension(cls)
    Registers a stix.Entity class as an implementation of an xml type.
    Classes must have an _XSI_TYPE class attributes to be registered. The value of this attribute must be a valid xsi:type.

Note: This was designed for internal use.

Version: 1.2.0.8

stix.indicator.valid_time Module

Classes

class stix.indicator.valid_time.ValidTime(start_time=None, end_time=None)
    Bases: mixbox.entities.Entity
3.1.10 STIX Report

Modules located in the stix.report package

Version: 1.2.0.8

stix.report Module

Overview

The stix.report module implements Report.
A Report defines a contextual wrapper for a grouping of STIX content.

Documentation Resources

• Report Data Model

Classes

class stix.report.Report (id_=None, idref=None, timestamp=None, header=None, courses_of_action=None, exploit_targets=None, indicators=None, observables=None, incidents=None, threat_actors=None, ttps=None, campaigns=None, related_reports=None)

Bases: stix.base.Entity

A STIX Report Object.

Parameters

• id (optional) – An identifier. If None, a value will be generated via mixbox.idgen.create_id(). If set, this will unset the idref property.
• idref (optional) – An identifier reference. If set this will unset the id_ property.
• timestamp (optional) – A timestamp value. Can be an instance of datetime.datetime or str.
• header – A Report Header object.
• campaigns – A collection of Campaign objects.
• courses_of_action – A collection of CourseOfAction objects.
• exploit_targets – A collection of ExploitTarget objects.
• incidents – A collection of Incident objects.
• indicators – A collection of Indicator objects.
• threat_actors – A collection of ThreatActor objects.
• ttps – A collection of TTP objects.
• related_reports – A collection of RelatedReport objects.

add(entity)

Adds entity to a top-level collection. For example, if entity is an Indicator object, the entity will be added to the indicators top-level collection.
add_campaign(campaign)
    Adds a Campaign object to the campaigns collection.

add_course_of_action(course_of_action)
    Adds a CourseOfAction object to the courses_of_action collection.

add_exploit_target(exploit_target)
    Adds an ExploitTarget object to the exploit_targets collection.

add_incident(incident)
    Adds an Incident object to the incidents collection.

add_indicator(indicator)
    Adds an Indicator object to the indicators collection.

add_observable(observable)
    Adds an Observable object to the observables collection.
    If observable is not an Observable instance, an effort will be made to convert it to one.

add_related_report(related_report)
    Adds an RelatedReport object to the related_reports collection.

add_threat_actor(threat_actor)
    Adds an ThreatActor object to the threat_actors collection.

addttp(ttp)
    Adds an TTP object to the ttps collection.

Version: 1.2.0.8

stix.report.header Module

Classes

class stix.report.header.Header(title=None, description=None, short_description=None, handling=None, intents=None, information_source=None)

Bases: stix.base.Entity

The Report Header.

Parameters

• handling – The data marking section of the Header.
• information_source – The InformationSource section of the Header.
• intents – A collection of VocabString defining the intent of the parent Report.
• description – A description of the intent or purpose of the parent Report.
• short_description – A short description of the intent or purpose of the parent Report.
• title – The title of the Report.

title
    The title of the parent Report.

add_description(description)
    Adds a description to the descriptions collection.
    This is the same as calling “foo.descriptions.add(bar)”. 
add_intent(intent)
    Adds VocabString object to the intents collection.
    If the input is not an instance of VocabString, an effort will be made to convert it into an instance of ReportIntent.

add_short_description(description)
    Adds a description to the short_descriptions collection.
    This is the same as calling “foo.short_descriptions.add(bar)”.

description
    A single description about the contents or purpose of this object.
    Default Value: None

    Note: If this object has more than one description set, this will return the description with the lowest ordinality value.

    Returns An instance of StructuredText

short_description
    A single short description about the contents or purpose of this object.
    Default Value: None

    Note: If this object has more than one short description set, this will return the description with the lowest ordinality value.

    Returns An instance of StructuredText

3.1.11 STIX Threat Actor

Modules located in the stix.threat_actor package

Version: 1.2.0.8

stix.threat_actor Module

Overview

The stix.threat_actor module implements ThreatActor.
ThreatActors are characterizations of malicious actors (or adversaries) representing a cyber attack threat including presumed intent and historically observed behavior.

Documentation Resources
    • Threat Actor Data Model
### Classes

**class** `stix.threat_actor.ThreatActor`(
  `id_=None, idref=None, timestamp=None, title=None, description=None, short_description=None`
)

**Bases:** `stix.base.BaseCoreComponent`

Implementation of the STIX Threat Actor.

**Parameters**

- `id` *(optional)* – An identifier. If None, a value will be generated via `mixbox.idgen.create_id()`. If set, this will unset the `idref` property.
- `idref` *(optional)* – An identifier reference. If set this will unset the `id_` property.
- `timestamp` *(optional)* – A timestamp value. Can be an instance of `datetime.datetime` or `str`.
- `description` – A description of the purpose or intent of this object.
- `short_description` – A short description of the intent or purpose of this object.
- `title` – The title of this object.

**add_description** *(description)*

Adds a description to the `descriptions` collection.

This is the same as calling “foo.descriptions.add(bar)”.

**add_intended_effect** *(value)*

Adds a `Statement` object to the `intended_effects` collection.

If `value` is a string, an attempt will be made to convert it into an instance of `Statement`.

**add_motivation** *(value)*

Adds a `Motivation` object to the `motivations` collection.

**add_planning_and_operational_support** *(value)*

Adds a `VocabString` object to the `planning_and_operational_supports` collection.

If `value` is a string, an attempt will be made to convert it to an instance of `PlanningAndOperationalSupport`.

**add_short_description** *(description)*

Adds a description to the `short_descriptions` collection.

This is the same as calling “foo.short_descriptions.add(bar)”.

**add_sophistication** *(value)*

Adds a `VocabString` object to the `sophistications` collection.

If `value` is a string, an attempt will be made to convert it to an instance of `ThreatActorSophistication`.

**add_type** *(value)*

Adds a `VocabString` object to the `types` collection.

If set to a string, an attempt will be made to convert it into an instance of `ThreatActorType`.

**description**

A single description about the contents or purpose of this object.

Default Value: `None`
Note: If this object has more than one description set, this will return the description with the lowest ordinality value.

Returns An instance of StructuredText

```python
find(id_)
```

Searches the children of a Entity implementation for an object with an id_ property that matches id_.

**short_description**

A single short description about the contents or purpose of this object.

Default Value: None

Note: If this object has more than one short description set, this will return the description with the lowest ordinality value.

Returns An instance of StructuredText

```python
to_dict()
```

Convert to a dict

Subclasses can override this function.

Returns Python dict with keys set from this Entity.

```python
to_json()
```

Export an object as a JSON String.

```python
to_obj(ns_info=None)
```

Convert to a GenerateDS binding object.

Subclasses can override this function.

Returns An instance of this Entity's _binding_class with properties set from this Entity.

class stix.threat_actor.AssociatedActors (scope=None, *args)
    Bases: stix.common.related.GenericRelationshipList

class stix.threat_actor.AssociatedCampaigns (scope=None, *args)
    Bases: stix.common.related.GenericRelationshipList

class stix.threat_actor.ObservedTTPs (scope=None, *args)
    Bases: stix.common.related.GenericRelationshipList

3.1.12 STIX Tactics, Techniques, and Procedures (TTP)

Modules located in the stix.ttp package

Version: 1.2.0.8

**stix.ttp Module**

**Overview**

The stix.ttp module implements TTP.
TTPs are representations of the behavior or modus operandi of cyber adversaries.

Documentation Resources

- TTP Data Model

Classes

class stix.ttp.TTP (id_=None, idref=None, timestamp=None, title=None, description=None, short_description=None)
Bases: stix.base.BaseCoreComponent

Implementation of the STIX TTP.

Parameters

- **id** (optional) – An identifier. If None, a value will be generated via mixbox.idgen.create_id(). If set, this will unset the idref property.
- **idref** (optional) – An identifier reference. If set this will unset the id_ property.
- **timestamp** (optional) – A timestamp value. Can be an instance of datetime.datetime or str.
- **description** – A description of the purpose or intent of this object.
- **short_description** – A short description of the intent or purpose of this object.
- **title** – The title of this object.

add_description (description)

Adds a description to the descriptions collection.

This is the same as calling “foo.descriptions.add(bar)”.

add_intended_effect (value)

Adds a Statement object to the intended_effects collection.

If value is a string, an attempt will be made to convert it into an instance of Statement.

add_kill_chain_phase (value)

Adds a KillChainPhaseReference to the kill_chain_phases collection.

Parameters value – A KillChainPhase, KillChainPhaseReference or a str representing the phase_id of. Note that you if you are defining a custom Kill Chain, you need to add it to the STIX package separately.

add_related_package (value)

Adds a RelatedPackageRef object to the related_packages collection.

Parameters value – A RelatedPackageRef or a STIXPackage object.

add_short_description (description)

Adds a description to the short_descriptions collection.

This is the same as calling “foo.short_descriptions.add(bar)”.

description

A single description about the contents or purpose of this object.

Default Value: None
Note: If this object has more than one description set, this will return the description with the lowest ordinality value.

Returns An instance of `StructuredText`

`find(id_=None)`
Searches the children of a `Entity` implementation for an object with an `id_` property that matches `id_`.

`short_description`
A single short description about the contents or purpose of this object.
Default Value: `None`

Note: If this object has more than one short description set, this will return the description with the lowest ordinality value.

Returns An instance of `StructuredText`

`to_dict()`
Convert to a dict
Subclasses can override this function.

Returns Python dict with keys set from this `Entity`.

`to_json()`
Export an object as a JSON String.

`to_obj(ns_info=None)`
Convert to a GenerateDS binding object.
Subclasses can override this function.

Returns An instance of this `Entity`'s `_binding_class` with properties set from this `Entity`.

Version: 1.2.0.8

`stix.ttp.attack_pattern` Module

Classes

`class stix.ttp.attack_pattern.AttackPattern(id_=None, idref=None, title=None, description=None, short_description=None)`
Bases: `stix.base.Entity`

`add_description(description)`
Adds a description to the `descriptions` collection.
This is the same as calling “foo.descriptions.add(bar)”.

`add_short_description(description)`
Adds a description to the `short_descriptions` collection.
This is the same as calling “foo.short_descriptions.add(bar)”.

3.1. API Reference
**description**
A single description about the contents or purpose of this object.

Default Value: None

**Note:** If this object has more than one description set, this will return the description with the lowest ordinality value.

**Returns** An instance of *StructuredText*

**short_description**
A single short description about the contents or purpose of this object.

Default Value: None

**Note:** If this object has more than one short description set, this will return the description with the lowest ordinality value.

**Returns** An instance of *StructuredText*

---

Version: 1.2.0.8

**stix.ttp.behavior Module**

**Classes**

**class** `stix.ttp.behavior.Behavior` *(malware_instances=None, attack_patterns=None, exploits=None)*

Bases: `stix.base.Entity`

Version: 1.2.0.8

**stix.ttp.exploit Module**

**Classes**

**class** `stix.ttp.exploit.Exploit` *(id_=None, idref=None, title=None, description=None, short_description=None)*

Bases: `stix.base.Entity`

**add_description**(description)

Adds a description to the *descriptions* collection.

This is the same as calling “foo.descriptions.add(bar)”.

**add_short_description**(description)

Adds a description to the *short_descriptions* collection.

This is the same as calling “foo.short_descriptions.add(bar)”.

**description**
A single description about the contents or purpose of this object.

Default Value: None
Note: If this object has more than one description set, this will return the description with the lowest ordinality value.

Returns An instance of `StructuredText`

**short_description**
A single short description about the contents or purpose of this object.
Default Value: None

Note: If this object has more than one short description set, this will return the description with the lowest ordinality value.

Returns An instance of `StructuredText`

Version: 1.2.0.8

**stix.ttp.exploit_targets Module**

**Classes**

```python
class stix.ttp.exploit_targets.ExploitTargets(scope=None, *args)
    Bases: stix.common.related.GenericRelationshipList
```

Version: 1.2.0.8

**stix.ttp.infrastructure Module**

**Classes**

```python
class stix.ttp.infrastructure.Infrastructure(id_=None, idref=None, title=None, description=None, short_description=None)
    Bases: stix.base.Entity
```

**add_description** *(description)*
Adds a description to the `descriptions` collection.
This is the same as calling “foo.descriptions.add(bar)”.

**add_short_description** *(description)*
Adds a description to the `short_descriptions` collection.
This is the same as calling “foo.short_descriptions.add(bar)”.

**description**
A single description about the contents or purpose of this object.
Default Value: None

Note: If this object has more than one description set, this will return the description with the lowest ordinality value.
Returns An instance of `StructuredText`

**short_description**
A single short description about the contents or purpose of this object.
Default Value: None

Note: If this object has more than one short description set, this will return the description with the lowest ordinality value.

Returns An instance of `StructuredText`

Version: 1.2.0.8

**stix.ttp.malware_instance Module**

**Classes**

```python
class stix.ttp.malware_instance.MalwareInstance (id_=None, idref=None, title=None, description=None, short_description=None)
```

Bases: `stix.base.Entity`

**add_description (description)**
Adds a description to the descriptions collection.
This is the same as calling “foo.descriptions.add(bar)”.

**add_short_description (description)**
Adds a description to the short_descriptions collection.
This is the same as calling “foo.short_descriptions.add(bar)”.

**description**
A single description about the contents or purpose of this object.
Default Value: None

Note: If this object has more than one description set, this will return the description with the lowest ordinality value.

Returns An instance of `StructuredText`

**short_description**
A single short description about the contents or purpose of this object.
Default Value: None

Note: If this object has more than one short description set, this will return the description with the lowest ordinality value.
**Functions**

```python
txt.ttp.malware_instance.add_extension(cls)
```

Registers a stix.Entity class as an implementation of an xml type.

Classes must have an _XSI_TYPE class attributes to be registered. The value of this attribute must be a valid xsi:type.

---

**Note:** This was designed for internal use.

---

**Version:** 1.2.0.8

**stix.ttp.related_ttps Module**

**Classes**

```python
class stix.ttp.related_tpps.RelatedTTPs(scope=None, *args)
    Bases: stix.common.related.GenericRelationshipList
```

**Version:** 1.2.0.8

**stix.ttp.resource Module**

**Classes**

```python
class stix.ttp.resource.Resource(tools=None, infrastructure=None, personas=None)
    Bases: stix.base.Entity
```

**Version:** 1.2.0.8

**stix.ttp.victim_targeting Module**

**Classes**

```python
class stix.ttp.victim_targeting.VictimTargeting
    Bases: stix.base.Entity
```

**3.1.13 STIX Utils**

Modules located in the stix.utils package

**Version:** 1.2.0.8

**stix.utils Module**

**Functions**

```python
txt.utils.is_cdata(text)
```

Returns True if text contains a CDATA block.
Example

```python
>>> is_cdata("<!CDATA[Foo]>")
True
>>> is_cdata("NOPE")
False
```

**stix.utils.strip_cdata(text)**

Removes all CDATA blocks from `text` if it contains them.

**Note:** If the function contains escaped XML characters outside of a CDATA block, they will be unescaped.

**Parameters**

`text` string containing one or more CDATA blocks. (A) –

Returns An XML unescaped string with CDATA block qualifiers removed.

**stix.utils.cdata(text)**

Wraps the input `text` in a `<![CDATA[ ]]>` block.

If the text contains CDATA sections already, they are stripped and replaced by the application of an outer-most CDATA block.

**Parameters**

`text` – A string to wrap in a CDATA block.

Returns The `text` value wrapped in `<![CDATA[]]>`

**stix.utils.raise_warnings(func)**

Function decorator that causes all Python warnings to be raised as exceptions in the wrapped function.

Example

```python
>>> @raise_warnings
>>> def foo():
>>>     warnings.warn("this will raise an exception")
```

**stix.utils.silence_warnings(func)**

Function decorator that silences/ignores all Python warnings in the wrapped function.

Example

```python
>>> @silence_warnings
>>> def foo():
>>>     warnings.warn("this will not appear")
```

**stix.utils.xml_bool(value)**

Returns `True` if `value` is an acceptable xs:boolean True value. Returns `False` if `value` is an acceptable xs:boolean False value. If `value` is `None`, this function will return `None`.

**Version:** 1.2.0.8

**stix.utils.dates Module**
Functions

`stix.util.dates.parse_value(value)`
Attempts to parse `value` into an instance of `datetime.datetime`. If `value` is `None`, this function will return `None`.

Parameters `value` – A timestamp. This can be a string or `datetime.datetime` value.

`stix.util.dates.serialize_value(value)`
Attempts to convert `value` into an ISO8601-compliant timestamp string. If `value` is `None`, `None` will be returned.

Parameters `value` – A `datetime.datetime` value.

Returns An ISO8601 formatted timestamp string.

`stix.util.dates.parse_date(value)`
Attempts to parse `value` into an instance of `datetime.date`. If `value` is `None`, this function will return `None`.

Parameters `value` – A timestamp. This can be a string, `datetime.date`, or `datetime.datetime` value.

`stix.util.dates.serialize_date(value)`
Attempts to convert `value` into an ISO8601-compliant timestamp string. If `value` is `None`, `None` will be returned.

Parameters `value` – A `datetime.datetime` value.

Returns An ISO8601 formatted timestamp string.

`stix.util.dates.now()`
Returns the current UTC `datetime.datetime` timestamp.

Version: 1.2.0.8

`stix.util.nsparser` Module

Constants

`stix.util.nsparser.NS_CAMPAIGN_OBJECT = Namespace(name='http://stix.mitre.org/Campaign-1', prefix='campaign')`
A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

`stix.util.nsparser.NS_CAPEC_OBJECT = Namespace(name='http://capec.mitre.org/capec-2', prefix='capec', schema_location='')`
A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

`stix.util.nsparser.NS_CVRF_OBJECT = Namespace(name='http://www.icasi.org/CVRF/schema/cvrf/1.1', prefix='cvrf', schema_location='')`
A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.
URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_ET_OBJECT = Namespace(name='http://stix.mitre.org/ExploitTarget-1', prefix='et', schema_location='http://stix.mitre.org/XMLSchema/exploit_target/1.2/exploit_target.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_GENERICTM_OBJECT = Namespace(name='http://stix.mitre.org/extensions/TestMechanism#Generic-1', prefix='genericTM', schema_location='http://stix.mitre.org/XMLSchema/extensions/test_mechanism/generic/1.2/generic_test_mechanism.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_INCIDENT_OBJECT = Namespace(name='http://stix.mitre.org/Incident-1', prefix='incident', schema_location='http://stix.mitre.org/XMLSchema/incident/1.2/incident.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_INDICATOR_OBJECT = Namespace(name='http://stix.mitre.org/Indicator-2', prefix='indicator', schema_location='http://stix.mitre.org/XMLSchema/indicator/2.2/indicator.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_IOC_OBJECT = Namespace(name='http://schemas.mandiant.com/2010/ioc', prefix='ioc', schema_location='')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_IOCTR_OBJECT = Namespace(name='http://schemas.mandiant.com/2010/ioc/TR/', prefix='ioc-tr', schema_location='')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_MARKING_OBJECT = Namespace(name='http://data-marking.mitre.org/Marking-1', prefix='marking', schema_location='http://stix.mitre.org/XMLSchema/data_marking/1.2/data_marking.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_OVALDEF_OBJECT = Namespace(name='http://oval.mitre.org/XMLSchema/oval-definitions-5', prefix='oval-def', schema_location='')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_OVALVAR_OBJECT = Namespace(name='http://oval.mitre.org/XMLSchema/oval-variables-5', prefix='oval-var', schema_location='')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.
A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_SNORTTM_OBJECT = Namespace(name='http://stix.mitre.org/extensions/TestMechanism#Snort-1', prefix='snortTM', schema_location='http://stix.mitre.org/XMLSchema/extensions/test_mechanism/snort/1.2/snort_test_mechanism.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_STIX_OBJECT = Namespace(name='http://stix.mitre.org/stix-1', prefix='stix', schema_location='http://stix.mitre.org/XMLSchema/core/1.2/stix_core.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_STIXVOCABS_OBJECT = Namespace(name='http://stix.mitre.org/default_vocabularies-1', prefix='stixVocabs', schema_location='http://stix.mitre.org/XMLSchema/default_vocabularies/1.2.0/stix_default_vocabularies.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_TA_OBJECT = Namespace(name='http://stix.mitre.org/ThreatActor-1', prefix='ta', schema_location='http://stix.mitre.org/XMLSchema/threat_actor/1.2/threat_actor.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_STIXCIQADDRESS_OBJECT = Namespace(name='http://stix.mitre.org/extensions/Address#CIQAddress3.0-1', prefix='stix-ciqaddress', schema_location='http://stix.mitre.org/XMLSchema/extensions/address/ciq_3.0/1.2/ciq_3.0_address.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_STIXCAPEC_OBJECT = Namespace(name='http://stix.mitre.org/extensions/AttackPattern#CAPEC2.7-1', prefix='stix-capec', schema_location='http://stix.mitre.org/XMLSchema/extensions/attack_pattern/capec_2.7/1.1/capec_2.7_attack_pattern.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_STIXCVRF_OBJECT = Namespace(name='http://stix.mitre.org/extensions/Vulnerability#CVRF-1', prefix='stix-cvrf', schema_location='http://stix.mitre.org/XMLSchema/extensions/vulnerability/cvrf_1.1/1.2/cvrf_1.1_vulnerability.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_STIXOPENIOC_OBJECT = Namespace(name='http://stix.mitre.org/extensions/TestMechanism#OpenIOC2010-1', prefix='stix-openioc', schema_location='http://stix.mitre.org/XMLSchema/extensions/test_mechanism/open_ioc_2010/1.2/open_ioc_2010_test_mechanism.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_STIXOVAL_OBJECT = Namespace(name='http://stix.mitre.org/extensions/TestMechanism#OVAL5.10-1', prefix='stix-oval', schema_location='http://stix.mitre.org/XMLSchema/extensions/test_mechanism/oval_5.10/1.2/oval_5.10_test_mechanism.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```python
stix.utils.nsparser.NS_STIXCOMMON_OBJECT = Namespace(name='http://stix.mitre.org/common-1', prefix='stix_common', schema_location='http://stix.mitre.org/XMLSchema/common/1.2/stix_common.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.
A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```
stix.utils.nsparser.NS_TTP_OBJECT = Namespace(name='http://stix.mitre.org/TTP-1', prefix='ttp', schema_location='http://stix.mitre.org/XMLSchema/ttp/1.2/ttp.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

```
stix.utils.nsparser.NS_YARATM_OBJECT = Namespace(name='http://stix.mitre.org/extensions/TestMechanism#YARA-1', prefix='yaraTM', schema_location='http://stix.mitre.org/XMLSchema/extensions/test_mechanism/yara/1.2/yara_test_mechanism.xsd')
```

A convenience class which represents simplified XML namespace info, consisting of exactly one namespace URI, and an optional prefix and schema location URI. This is handy for building up big tables of namespace data.

**Version**: 1.2.0.8

---

### `stix.utils.parser` Module

#### Classes

```
class stix.utils.parser.UnsupportedVersionError (message, expected=None, found=None)
Bases: exceptions.Exception
```

A parsed document is a version unsupported by the parser.

```
class stix.utils.parser.UnknownVersionError
Bases: exceptions.Exception
```

A parsed document contains no version information.

```
stix.utils.parser.UnsupportedRootElement
alias of.UnsupportedRootElementError
```

```
class stix.utils.parser.EntityParser
Bases: mixbox.parser.EntityParser
```

**Version**: 1.2.0.8
3.2 API Coverage

The python-stix APIs currently provide partial coverage of all STIX-defined constructs. Development is ongoing toward the goal of providing full STIX language support in the APIs. Until such time that full coverage is provided, an overview of which constructs are available in these APIs will be maintained below.

Note: Many STIX constructs can contain CybOX constructs. The python-cybox project provides its own APIs for interacting with the CybOX specification. Please see the CybOX API Documentation for information about CybOX API coverage.

### 3.2.1 STIX Core

<table>
<thead>
<tr>
<th>STIX Construct</th>
<th>API Coverage</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>STIX Package</td>
<td>Full</td>
<td>stix.core.stix_package.STIXPackage</td>
</tr>
<tr>
<td>STIX Header</td>
<td>Full</td>
<td>stix.core.stix_header.STIXHeader</td>
</tr>
<tr>
<td>Related Packages</td>
<td>Full</td>
<td>stix.core.stix_package.RelatedPackages</td>
</tr>
</tbody>
</table>

### 3.2.2 STIX Top-level Constructs

<table>
<thead>
<tr>
<th>STIX Construct</th>
<th>API Coverage</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Campaign</td>
<td>Full</td>
<td>stix.campaign.Campaign</td>
</tr>
<tr>
<td>Course of Action</td>
<td>Full</td>
<td>stix.coa.CourseOfAction</td>
</tr>
<tr>
<td>Exploit Target</td>
<td>Full</td>
<td>stix.exploit_target.ExploitTarget</td>
</tr>
<tr>
<td>Incident</td>
<td>Partial</td>
<td>stix.incident.Incident</td>
</tr>
<tr>
<td>Indicator</td>
<td>Full</td>
<td>stix.indicator.indicator.Indicator</td>
</tr>
<tr>
<td>Observable</td>
<td>Provided by CybOX</td>
<td></td>
</tr>
<tr>
<td>Threat Actor</td>
<td>Full</td>
<td>stix.threat_actor.ThreatActor</td>
</tr>
<tr>
<td>TTP</td>
<td>Partial</td>
<td>stix.ttp.TTP</td>
</tr>
</tbody>
</table>

### 3.2.3 STIX Features

<table>
<thead>
<tr>
<th>STIX Construct</th>
<th>API Coverage</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Confidence</td>
<td>Partial</td>
<td>stix.common.confidence.Confidence</td>
</tr>
<tr>
<td>Handling</td>
<td>Full</td>
<td>stix.data_marking.Marking</td>
</tr>
<tr>
<td>Markup in Structured Text</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Relationships</td>
<td>Full</td>
<td></td>
</tr>
</tbody>
</table>
### 3.2.4 STIX Extensions

<table>
<thead>
<tr>
<th>STIX Construct</th>
<th>API Coverage</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address Extensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CIQ Address</td>
<td>× None</td>
<td></td>
</tr>
<tr>
<td><strong>Attack Pattern Extensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CAPEC 2.7</td>
<td>× None</td>
<td></td>
</tr>
<tr>
<td><strong>Identity Extensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CIQ Identity</td>
<td>Partial</td>
<td>stix.extensions.identity.ciq_identity_3_0.CIQIdentity3_0Instance</td>
</tr>
<tr>
<td><strong>Malware Extensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAEC</td>
<td>Full</td>
<td>stix.extensions.malware.maec_4_1_malware.MAECInstance</td>
</tr>
<tr>
<td><strong>Marking Extensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simple Marking</td>
<td>Full</td>
<td>stix.extensions.marking.simple_marking.SimpleMarkingStructure</td>
</tr>
<tr>
<td>TLP</td>
<td>Full</td>
<td>stix.extensions.marking.tlp.TLPMarkingStructure</td>
</tr>
<tr>
<td>Terms of Use</td>
<td>Full</td>
<td>stix.extensions.marking.terms_of_use_marking.TermsOfUseMarkingStructure</td>
</tr>
<tr>
<td><strong>Structured COA Extensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Generic Structured COA</td>
<td>Full</td>
<td>stix.extensions.structured_coa.generic_structured_coa.GenericStructuredCOA</td>
</tr>
<tr>
<td><strong>Test Mechanism Extensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Generic Test Mechanism</td>
<td>Full</td>
<td>stix.extensions.test_mechanism.generic_test_mechanism.GenericTestMechanism</td>
</tr>
<tr>
<td>OVAL</td>
<td>× None</td>
<td>stix.extensions.test_mechanism.oval.OVAL</td>
</tr>
<tr>
<td>OpenIOC</td>
<td>Full</td>
<td>stix.extensions.test_mechanism.open_ioc_2010_test_mechanism.OpenIOCTestMechanism</td>
</tr>
<tr>
<td>SNORT</td>
<td>Full</td>
<td>stix.extensions.test_mechanism.snort.SnortTestMechanism</td>
</tr>
<tr>
<td>YARA</td>
<td>Full</td>
<td>stix.extensions.test_mechanism.yara_test_mechanism.YaraTestMechanism</td>
</tr>
<tr>
<td><strong>Vulnerability Extensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CVRF</td>
<td>× None</td>
<td></td>
</tr>
</tbody>
</table>

### 3.2.5 STIX Vocabularies
<table>
<thead>
<tr>
<th>STIX Construct</th>
<th>API Coverage</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AssetTypeVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.Asset_type_1_0</td>
</tr>
<tr>
<td>AttackerInfrastructureTypeVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.AttackerInfrastructureType_v1_0</td>
</tr>
<tr>
<td>AttackerToolTypeVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.AttackerToolType_v1_0</td>
</tr>
<tr>
<td>AvailabilityLossTypeVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.AvailabilityLossType_v1_0</td>
</tr>
<tr>
<td>AvailabilityLossTypeVocab-1.1.1</td>
<td>Full</td>
<td>stix.common.vocabs.AvailabilityLossType_v1_1.1</td>
</tr>
<tr>
<td>COASTageVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.COASTage_v1_0</td>
</tr>
<tr>
<td>CampaignStatusVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.CampaignStatus_v1_0</td>
</tr>
<tr>
<td>CourseOfActionTypeVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.CourseOfActionType_v1_0</td>
</tr>
<tr>
<td>DiscoveryMethodVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.DiscoveryMethod_v1_0</td>
</tr>
<tr>
<td>DiscoveryMethodVocab-2.0</td>
<td>Full</td>
<td>stix.common.vocabs.DiscoveryMethod_v2_0</td>
</tr>
<tr>
<td>HighMediumLowVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.HighMediumLow_v1_0</td>
</tr>
<tr>
<td>ImpactQualificationVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.ImpactQualification_v1_0</td>
</tr>
<tr>
<td>ImpactRatingVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.ImpactRating_v1_0</td>
</tr>
<tr>
<td>IncidentCategoryVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.IncidentCategory_v1_0</td>
</tr>
<tr>
<td>IncidentEffectVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.IncidentEffect_v1_0</td>
</tr>
<tr>
<td>IncidentStatusVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.IncidentStatus_v1_0</td>
</tr>
<tr>
<td>IndicatorTypeVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.IndicatorType_v1_0</td>
</tr>
<tr>
<td>IndicatorTypeVocab-1.1</td>
<td>Full</td>
<td>stix.common.vocabs.IndicatorType_v1_1</td>
</tr>
<tr>
<td>InformationSourceRoleVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.InformationSourceRole_v1_0</td>
</tr>
<tr>
<td>InformationTypeVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.InformationType_v1_0</td>
</tr>
<tr>
<td>IntendedEffectVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.IntendedEffect_v1_0</td>
</tr>
<tr>
<td>LocationClassVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.LocationClass_v1_0</td>
</tr>
<tr>
<td>LossDurationVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.LossDuration_v1_0</td>
</tr>
<tr>
<td>LossPropertyVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.LossProperty_v1_0</td>
</tr>
<tr>
<td>MalwareTypeVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.MalwareType_v1_0</td>
</tr>
<tr>
<td>ManagementClassVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.ManagementClass_v1_0</td>
</tr>
<tr>
<td>MotivationVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.Motivation_v1_0</td>
</tr>
<tr>
<td>MotivationVocab-1.0.1</td>
<td>Full</td>
<td>stix.common.vocabs.Motivation_v1_1.1</td>
</tr>
<tr>
<td>MotivationVocab-1.1</td>
<td>Full</td>
<td>stix.common.vocabs.Motivation_v1_1</td>
</tr>
<tr>
<td>OwnershipClassVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.OwnershipClass_v1_0</td>
</tr>
<tr>
<td>PackageIntentVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.PackageIntent_v1_0</td>
</tr>
<tr>
<td>PlanningAndOperationalSupportVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.PlanningAndOperationalSupport_v1_0</td>
</tr>
<tr>
<td>PlanningAndOperationalSupportVocab-1.0.1</td>
<td>Full</td>
<td>stix.common.vocabs.PlanningAndOperationalSupport_v1.1_0</td>
</tr>
<tr>
<td>SecurityCompromiseVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.SecurityCompromise_v1_0</td>
</tr>
<tr>
<td>SystemTypeVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.SystemType_v1_0</td>
</tr>
<tr>
<td>ThreatActorSophisticationVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.ThreatActorSophistication_v1_0</td>
</tr>
<tr>
<td>ThreatActorTypeVocab-1.0</td>
<td>Full</td>
<td>stix.common.vocabs.ThreatActorType_v1_0</td>
</tr>
</tbody>
</table>

3.2. API Coverage 105
• My RAM consumption rises when processing a large amount of files. This problem is caused by a python-
cybox caching mechanism that is enabled by default. To prevent this issue from happening use the
cybox.utils.caches.cache_clear() method in your code/script to release the cached resources
as appropriate. Refer to the cybox documentation for more details.
If a bug is found, a feature is missing, or something just isn’t behaving the way you’d expect it to, please submit an issue to our tracker. If you’d like to contribute code to our repository, you can do so by issuing a pull request and we will work with you to try and integrate that code into our repository.
Indices and tables

- genindex
- modindex
- search
stix.base, 17
stix.campaign, 19
stix.coa, 51
stix.coa.objective, 52
stix.common, 21
stix.common.activity, 21
stix.common.confidence, 21
stix.common.datetime_with_precision, 22
stix.common.identity, 22
stix.common.information_source, 23
stix.common.kill_chains, 23
stix.common.related, 24
stix.common.statement, 25
stix.common.structured_text, 26
stix.common.tools, 28
stix.common.vocabs, 28
stix.core.stix_header, 46
stix.core.stix_package, 47
stix.corettps, 50
stix.data_marking, 18
stix.exploit_target, 53
stix.exploit_target.configuration, 55
stix.exploit_target.vulnerability, 56
stix.exploit_target.weakness, 57
stix.extensions.identity.ciq_identity_3_0, 58
stix.extensions.malware.maec_4_1_malware, 61
stix.extensions.marking.ais, 61
stix.extensions.marking.simple_marking, 67
stix.extensions.marking.terms_of_use_marking, 68
stix.extensions.marking.tlp, 68
stix.extensions.structured_coa.generic_structured_coa, 68
stix.extensions.test_mechanism.generic_test_mechanism, 68
stix.extensions.test_mechanism.open_ioc_2010_test_mechanism, 69
stix.extensions.test_mechanism.snort_test_mechanism, 69
stix.extensions.test_mechanism.yara_test_mechanism, 69
stix.incident, 70
stix.incident.affected_asset, 73
stix.incident.coa, 73
stix.incident.contributors, 73
stix.incident.direct_impact_summary, 74
stix.incident.external_id, 74
stix.incident.history, 74
stix.incident.indirect_impact_summary, 75
stix.incident.loss_estimation, 75
stix.incident.property_affected, 75
stix.incident.time, 75
stix.incident.total_loss_estimation, 76
stix.indicator.indicator, 76
stix.indicator.sightings, 85
stix.indicator.test_mechanism, 86
stix.indicator.valid_time, 86
stix.report, 87
stix.report.header, 88
stix.threat_actor, 89
stix.ttp, 91
stix.ttp.attack_pattern, 93
stix.ttp.behavior, 94
stix.ttp.exploit, 94
stix.ttp.exploit_targets, 95
stix.ttp.infrastructure, 95
stix.ttp.malware_instance, 96
stix.ttp.relatedttps, 97
stix.ttp.resource, 97
stix.ttp.victim_targeting, 97
stix.util, 97
stix.test_mechanism, 98
stix.util.nsparser, 99
stix.utils, 102

Symbols

_BaseNameElement (class in stix.extensions.identity.ciq_identity_3_0), 59
_BaseRelated (class in stix.common.related), 24
_BaseTestMechanism (class in stix.indicator.test_mechanism), 86
__delitem__() (stix.common.structured_text.StructuredTextList method), 26
__getitem__() (stix.common.structured_text.StructuredTextList method), 26
__iter__() (stix.common.structured_text.StructuredTextList method), 27
__str__() (stix.common.structured_text.StructuredText method), 26
__unicode__() (stix.common.structured_text.StructuredText method), 26

A

Activity (class in stix.common.activity), 21
add() (stix.common.structured_text.StructuredTextList method), 27
add() (stix.core.stix_package.STIXPackage method), 48
add() (stix.report.Report method), 87
add_activity() (stix.campaign.Campaign method), 19
add_affected_asset() (stix.incident.Incident method), 70
add_aiz_marking() (in module stix.extensions.marking.aiz), 61
add_alternative_id() (stix.indicator.indicator.Indicator method), 77
add_campaign() (stix.core.stix_package.STIXPackage method), 48
add_campaign() (stix.report.Report method), 87
add_category() (stix.incident.Incident method), 70
add_coa_requested() (stix.incident.Incident method), 70
add_coa_taken() (stix.incident.Incident method), 70
add_configuration() (stix.exploit_target.ExploitTarget method), 53
add_coordinator() (stix.incident.Incident method), 70
add_course_of_action() (stix.core.stix_package.STIXPackage method), 48
add_description() (stix.campaign.Campaign method), 19
add_description() (stix.coa.CourseOfAction method), 51
add_description() (stix.coa.objective.Objective method), 52
add_description() (stix.common.activity.Activity method), 21
add_description() (stix.common.confidence.Confidence method), 21
add_description() (stix.common.information_source.InformationSource method), 23
add_description() (stix.common.statement.Statement method), 25
add_description() (stix.core.stix_header.STIXHeader method), 47
add_description() (stix.exploit_target.configuration.Configuration method), 56
add_description() (stix.exploit_target.ExploitTarget method), 54
add_description() (stix.exploit_target.vulnerability.Vulnerability method), 57
add_description() (stix.exploit_target.weakness.Weakness method), 58
add_description() (stix.extensions.structured_coa.generic_structured_coa.GenericStructuredCOA method), 68
add_description() (stix.extensions.test_mechanism.generic_test_mechanism.GenericTestMechanism method), 68
add_description() (stix.incident.affected_asset.AffectedAsset method), 73
add_description() (stix.incident.Incident method), 70
add_description() (stix.indicator.indicator.Indicator method), 77
add_description() (stix.indicator.sightings.Sighting method), 85
add_description() (stix.report.header.Header method), 88
add_description() (stix.threat_actor.ThreatActor method), 90
add_description() (stix.ttp.attack_pattern.AttackPattern method), 93
add_description() (stix.ttp.exploit.Exploit method), 94
add_description() (stix.ttp.infrastructure.Infrastructure method), 95
add_description() (stix.ttp.malware_instance.MalwareInstance method), 96
add_description() (stix.ttp.TTP method), 92
add_discovery_method() (stix.incident.Incident method), 70
add_exploit_target() (stix.core.stix_package.STIXPackage method), 48
add_exploit_target() (stix.report.Report method), 88
add_extension() (in module stix.common.identity), 22
add_extension() (in module stix.data_marking), 18
add_extension() (in module stix.indicator.test_mechanism), 86
add_extension() (in module stix.ttp.malware_instance), 97
add_external_id() (stix.incident.Incident method), 71
add_incident() (stix.core.stix_package.STIXPackage method), 49
add_incident() (stix.report.Report method), 88
add_indicated_ttp() (stix.indicator.indicator.Indicator method), 77
add_indicator() (stix.core.stix_package.STIXPackage method), 49
add_indicator() (stix.report.Report method), 88
add_indicator_type() (stix.indicator.indicator.Indicator method), 77
add_intended_effect() (stix.incident.Incident method), 71
add_intended_effect() (stix.threat_actor.ThreatActor method), 90
add_intended_effect() (stix.ttp.TTP method), 92
add_intent() (stix.report.header.Header method), 88
add_kill_chain_phase() (stix.indicator.indicator.Indicator method), 77
add_kill_chain_phase() (stix.ttp.TTP method), 92
add_motivation() (stix.threat_actor.ThreatActor method), 90
add_object() (stix.indicator.indicator.Indicator method), 78
add_observable() (stix.core.stix_package.STIXPackage method), 49
add_observable() (stix.indicator.indicator.Indicator method), 78
add_observable() (stix.report.Report method), 88
add_package_intent() (stix.core.stix_header.STIXHeader method), 47
add_planning_and_operational_support() (stix.threat_actor.ThreatActor method), 90
add_profile() (stix.core.stix_header.STIXHeader method), 47
add_related_campaign() (stix.indicator.indicator.Indicator method), 78
add_related_campaign() (stix.indicator.indicator.Indicator method), 79
add_related_campaign() (stix.indicator.indicator.Indicator method), 79
add_related_observable() (stix.incident.Incident method), 71
add_related_package() (stix.core.stix_package.STIXPackage method), 49
add_related_package() (stix.ttp.TTP method), 92
add_related_report() (stix.report.Report method), 88
add_report() (stix.core.stix_package.STIXPackage method), 49
add_responder() (stix.incident.Incident method), 71
add_short_description() (stix.campaign.Campaign method), 19
add_short_description() (stix.coa.CourseOfAction method), 51
add_short_description() (stix.coa.objective.Objective method), 52
add_short_description() (stix.common.tools.ToolInformation method), 28
add_short_description() (stix.core.stix_header.STIXHeader method), 47
add_short_description() (stix.exploit_target.configuration.Configuration method), 56
add_short_description() (stix.exploit_target.ExploitTarget method), 54
add_short_description() (stix.exploit_target.vulnerability.Vulnerability method), 57
add_short_description() (stix.incident.Incident method), 71
add_short_description() (stix.indicator.indicator.Indicator method), 79
add_short_description() (stix.report.header.Header method), 89
add_short_description() (stix.threat_actor.ThreatActor method), 90
add_short_description() (stix.ttp.attack_pattern.AttackPattern method), 93
add_short_description() (stix.ttp.exploit.Exploit method), 94
add_short_description() (stix.ttp.infrastructure.Infrastructure method), 95
add_short_description() (stix.ttp.malware_instance.MalwareInstance method), 96
add_short_description() (stix.ttp.TTP method), 92
add_sophistication() (stix.threat_actor.ThreatActor method), 90
add_test_mechanism() (stix.indicator.indicator.Indicator method), 79
add_test_mechanism() (stix.threat_actor.ThreatActor method), 90
add_threat_actor() (stix.core.stix_package.STIXPackage method), 49
add_threat_actor() (stix.report.Report method), 88
add_ttp() (stix.core.stix_package.STIXPackage method), 49
add_ttp() (stix.report.Report method), 88

Index
add_type() (stix.threat_actor.ThreatActor method), 90
add_valid_time_position()
    (stix.indicator.indicator.Indicator method), 79
add_victim() (stix.incident.Incident method), 72
add_vocabulary() (stix.exploit_target.ExploitTarget method), 54
add_weakness() (stix.exploit_target.ExploitTarget method), 54
Address (class in stix.extensions.identity.ciq_identity_3_0), 59
AdministrativeArea (class in stix.extensions.identity.ciq_identity_3_0), 59
AffectedAsset (class in stix.incident.affected_asset), 73
AffectedSoftware (class in stix.exploit_target.vulnerability), 57
AISMarkingStructure (class in stix.extensions.marking.ais), 61
AssetType (class in stix.incident.affected_asset), 73
AssetType (in module stix.common.vocabs), 44
AssetType_1_0 (class in stix.common.vocabs), 28
AssociatedActors (class in stix.threat_actor), 91
AssociatedCampaigns (class in stix.campaign), 20
AssociatedCampaigns (class in stix.threat_actor), 91
AttackerInfrastructureType (in module stix.common.vocabs), 44
AttackerInfrastructureType_1_0 (class in stix.common.vocabs), 30
AttackerToolType (in module stix.common.vocabs), 44
AttackerToolType_1_0 (class in stix.common.vocabs), 31
AttackPattern (class in stix.ttp.attack_pattern), 93
AttributedThreatActors (class in stix.incident), 72
Attribution (class in stix.campaign), 20
AvailabilityLossType (in module stix.common.vocabs), 44
AvailabilityLossType_1_0 (class in stix.common.vocabs), 31
AvailabilityLossType_1_1 (class in stix.common.vocabs), 32

B
Behavior (class in stix.ttp.behavior), 94

C
Campaign (class in stix.campaign), 19
CampaignStatus (module stix.common.vocabs), 44
CampaignStatus_1_0 (class in stix.common.vocabs), 32
cdata() (module stix.utils), 98
CHEMICAL_SECTOR (module stix.extensions.marking.ais), 66
CIQIdentity3_0Instance (class in stix.extensions.identity.ciq_identity_3_0), 58
COARequested (class in stix.incident.coa), 73
COAStage (module stix.common.vocabs), 44
COAStage_1_0 (class in stix.common.vocabs), 32
COATaken (class in stix.incident.coa), 73
COATime (class in stix.incident.coa), 73
COMMERCIAL_FACILITIES_SECTOR (module stix.extensions.marking.ais), 66
COMMUNICATIONS_SECTOR (module stix.extensions.marking.ais), 66
CompositeIndicatorExpression (class in stix.indicator.indicator), 82
Confidence (class in stix.common.confidence), 21
Configuration (class in stix.exploit_target.configuration), 55
ContactNumber (class in stix.extensions.identity.ciq_identity_3_0), 59
ContactNumberElement (class in stix.extensions.identity.ciq_identity_3_0), 59
ContributingSources (class in stix.common.information_source), 23
Contributors (class in stix.incident.contributors), 74
Country (class in stix.extensions.identity.ciq_identity_3_0), 59
CourseOfAction (class in stix.coa), 51
CourseOfActionType (module stix.common.vocabs), 45
CourseOfActionType_1_0 (class in stix.common.vocabs), 32
CRITICAL_MANUFACTURING_SECTOR (module stix.extensions.marking.ais), 66
CVSSVector (class in stix.exploit_target.vulnerability), 57

D
DAMS_SECTOR (module stix.extensions.marking.ais), 66
DATE_PRECISION_VALUES (module stix.common.datetimewithprecision), 22
DATETIME_PRECISION_VALUES (module stix.common.datetimewithprecision), 22
DateTimeWithPrecision (class in stix.common.datetimewithprecision), 22
DEFENSE_INDUSTRIAL_BASE_SECTOR (module stix.extensions.marking.ais), 66
description (stix.campaign.Campaign attribute), 19
description (stix.coa.CourseOfAction attribute), 51
description (stix.coa.objective.Objective attribute), 52
description (stix.common.activity.Activity attribute), 21
description (stix.common.confidence.Confidence attribute), 21

Index 117
description (stix.common.information_source.InformationSource attribute), 23
description (stix.common.statement.Statement attribute), 25
description (stix.core.stix_header.STIXHeader attribute), 47
description (stix.exploit_target.configuration.Configuration attribute), 56
description (stix.exploit_target.ExploitTarget attribute), 54
description (stix.exploit_target.vulnerability.Vulnerability attribute), 57
description (stix.exploit_target.weakness.Weakness attribute), 58
description (stix.extensions.structured_coa.generic_structured_coa.GenericStructuredCOA attribute), 68
description (stix.extensions.test_mechanism.generic_test_mechanism.GenericTestMechanism attribute), 69
description (stix.incident.affected_asset.AffectedAsset attribute), 73
description (stix.incident.Incident attribute), 72
description (stix.indicator.indicator.Indicator attribute), 79
description (stix.indicator.sightings.Sighting attribute), 86
description (stix.report.header.Header attribute), 89
description (stix.threat_actor.ThreatActor attribute), 91
description (stix.ttp.attack_pattern.AttackPattern attribute), 93
description (stix.ttp.exploit.Exploit attribute), 94
description (stix.ttp.infrastructure.Infrastructure attribute), 95
description (stix.ttp.TTP attribute), 92
description_of_effect (stix.incident.property_affected.PropertyAffected attribute), 75
DirectImpactSummary (class in stix.incident.direct_impact_summary), 74
DiscoveryMethod (in module stix.common.vocabs), 45
DiscoveryMethod_1_0 (class in stix.common.vocabs), 33
DiscoveryMethod_2_0 (class in stix.common.vocabs), 33

E
ElectronicAddressIdentifier (class in stix.extensions.identity.ciq_identity_3_0), 59
EMERGENCY_SERVICES_SECTOR (in module stix.extensions.marking.ais), 66
EncodedCDATA (class in stix.common), 21
ENERGY_SECTOR (in module stix.extensions.marking.ais), 66
Entity (class in stix.base), 17
EntityList (class in stix.base), 18
EntityParser (class in stix.utils.parser), 102
Exploit (class in stix.ttp.exploit), 94
ExploitTarget (class in stix.exploit_target), 53
ExploitTargets (class in stix.ttp.exploit_targets), 95
ExternalID (class in stix.incident.external_id), 74

F
FINANCIAL_SERVICES_SECTOR (in module stix.extensions.marking.ais), 67
find() (stix.base.Entity method), 17
find() (stix.campaign.Campaign method), 20
find() (stix.coa.CourseOfAction method), 51
find() (stix.core.stix_package.STIXPackage method), 49
find() (stix.exploit_target.ExploitTarget method), 54
find() (stix.incident.Incident method), 72
find() (stix.indicator.indicator.Indicator method), 80
find() (stix.threat_actor.ThreatActor method), 91
from_xml() (stix.core.stix_package.STIXPackage class method), 49

G
GenericRelationship (class in stix.common.related), 24
GenericRelationshipList (class in stix.common.related), 24
GenericStructuredCOA (class in stix.extensions.structured_coa.generic_structured_coa), 68
GenericTestMechanism (class in stix.extensions.test_mechanism.generic_test_mechanism), 68
get_produced_time() (stix.indicator.indicator.Indicator method), 80
get_received_time() (stix.indicator.indicator.Indicator method), 80
GOVERNMENT_FACILITIES_SECTOR (in module stix.extensions.marking.ais), 67

H
Header (class in stix.report.header), 88
HEALTH_CARE_AND_PUBLIC_HEALTH_SECTOR (in module stix.extensions.marking.ais), 67
HighMediumLow (in module stix.common.vocabs), 45
HighMediumLow_1_0 (class in stix.common.vocabs), 33
History (class in stix.incident.history), 74
HistoryItem (class in stix.incident.history), 74

118 Index
Index

I
id_ (stix.common.structured_text.StructuredText attribute), 26
Identity (class in stix.common.identity), 22
ImpactAssessment (class in stix.incident.impact_assessment), 74
ImpactQualification (in module stix.common.vocabs), 45
ImpactQualification_1_0 (class in stix.common.vocabs), 34
ImpactRating (in module stix.common.vocabs), 45
ImpactRating_1_0 (class in stix.common.vocabs), 34
Incident (class in stix.incident), 70
IncidentCategory (in module stix.common.vocabs), 45
IncidentCategory_1_0 (class in stix.common.vocabs), 34
IncidentEffect (in module stix.common.vocabs), 45
IncidentEffect_1_0 (class in stix.common.vocabs), 34
IncidentStatus (in module stix.common.vocabs), 45
IncidentStatus_1_0 (class in stix.common.vocabs), 35
Indicator (class in stix.indicator.indicator), 76
IndicatorType (in module stix.common.vocabs), 45
IndicatorType_1_0 (class in stix.common.vocabs), 35
IndicatorType_1_1 (class in stix.common.vocabs), 35
IndicatorTypes (class in stix.indicator.indicator), 85
IndirectImpactSummary (class in stix.incident.indirect_impact_summary), 75
INFORMATION_TECHNOLOGY_SECTOR (in module stix.extensions.marking.ais), 67
InformationSource (class in stix.common.information_source), 23
InformationSourceRole (in module stix.common.vocabs), 45
InformationSourceRole_1_0 (class in stix.common.vocabs), 36
InformationType (in module stix.common.vocabs), 45
InformationType_1_0 (class in stix.common.vocabs), 36
Infrastructure (class in stix.ttp.infrastructure), 95
insert() (stix.common.structured_text.StructuredTextList method), 27
IntendedEffect (in module stix.common.vocabs), 45
IntendedEffect_1_0 (class in stix.common.vocabs), 36
is_cdata() (in module stix.utils), 97
is_plain() (stix.common.vocabs.VocabString method), 44
is_plain() (stix.incident.affected_asset.AssetType method), 73

J
JournalEntry (class in stix.incident.history), 74

K
KillChain (class in stix.common.kill_chains), 23
KillChainPhase (class in stix.common.kill_chains), 23
KillChainPhaseReference (class in stix.common.kill_chains), 23
KillChains (class in stix.common.kill_chains), 23
L
Language (class in stix.extensions.identity.ciq_identity_3_0), 59
LeveragedTTPs (in module stix.incident), 72
LocationClass (in module stix.common.vocabs), 45
LocationClass_1_0 (class in stix.common.vocabs), 37
LossDuration (in module stix.common.vocabs), 45
LossDuration_1_0 (class in stix.common.vocabs), 37
LossEstimation (class in stix.incident.loss_estimation), 75
LossProperty (in module stix.common.vocabs), 45
LossProperty_1_0 (class in stix.common.vocabs), 38
M
MAECInstance (class in stix.extensions.malware.maec_4_1_malware), 61
MalwareInstance (class in stix.ttp.malware_instance), 96
MalwareType (in module stix.common.vocabs), 45
MalwareType_1_0 (class in stix.common.vocabs), 38
ManagementClass (in module stix.common.vocabs), 45
ManagementClass_1_0 (class in stix.common.vocabs), 38
Marking (class in stix.data_marking), 18
MarkingSpecification (class in stix.data_marking), 18
MarkingStructure (class in stix.data_marking), 18
Motivation (in module stix.common.vocabs), 45
Motivation_1_0 (class in stix.common.vocabs), 38
Motivation_1_0_1 (class in stix.common.vocabs), 39
Motivation_1_1 (class in stix.common.vocabs), 39
N
NameElement (class in stix.extensions.identity.ciq_identity_3_0), 59
NameLine (class in stix.extensions.identity.ciq_identity_3_0), 59
Names (class in stix.campaign), 20
next_ordinality (stix.common.structured_text.StructuredTextList attribute), 27
NonPublicDataCompromised (class in stix.incident.property_affected), 75
now() (in module stix.utils.dates), 99
NS_CAMPAIGN_OBJECT (in module stix.utils.nsparser), 99
NS_CAPEC_OBJECT (in module stix.utils.nsparser), 99
NS_CIIQIDENTITY_OBJECT (in module stix.utils.nsparser), 99
NS_COA_OBJECT (in module stix.utils.nsparser), 99
NS_CVRF_OBJECT (in module stix.utils.nsparser), 99
Objective (class in stix.coa.objective), 52
observables (stix.indicator.indicator.Indicator attribute), 80
ObservedTTPs (class in stix.threat_actor), 91
OP_AND (stix.indicator.indicator.CompositeIndicatorExpression attribute), 82
OP_OR (stix.indicator.indicator.CompositeIndicatorExpression attribute), 82
OpenIOCTestMechanism (class in stix.extensions.test_mechanism.open_ioc_2010_test_mechanism), 69
operator (stix.indicator.indicator.CompositeIndicatorExpression attribute), 82
OPERATORS (stix.indicator.indicator.CompositeIndicatorExpression attribute), 82
OrganisationInfo (class in stix.extensions.identity.ciq_identity_3_0), 60
OrganisationName (class in stix.extensions.identity.ciq_identity_3_0), 60
OrganisationNameElement (class in stix.extensions.identity.ciq_identity_3_0), 60
OTHER (in module stix.extensions.marking.ais), 67
OwnershipClass (in module stix.common.vocabs), 45
OwnershipClass_1_0 (class in stix.common.vocabs), 40
PackageIntent (in module stix.common.vocabs), 45
PackageIntent_1_0 (class in stix.common.vocabs), 40
parse_date() (in module stix.utils.dates), 99
parse_value() (in module stix.utils.dates), 99
PartyName (class in stix.extensions.identity.ciq_identity_3_0), 60
PersonName (class in stix.extensions.identity.ciq_identity_3_0), 60
PersonNameElement (class in stix.extensions.identity.ciq_identity_3_0), 60
PlanningAndOperationalSupport (in module stix.common.vocabs), 45
PlanningAndOperationalSupport_1_0 (class in stix.common.vocabs), 40
PotentialCOAs (class in stix.exploit_target), 55
profiles (stix.core.stix_header.STIXHeader attribute), 47
PropertyAffected (class in stix.incident.property_affected), 75
<table>
<thead>
<tr>
<th>R</th>
<th>set_received_time() (stix.indicator.indicator.Indicator method), 81</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>short_description (stix.campaign.Campaign attribute), 20</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.coa.CourseOfAction attribute), 52</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.coa.objective.Objective attribute), 53</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.common.tools.ToolInformation attribute), 28</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.core.stix_header.STIXHeader attribute), 47</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.exploit_target.configuration.Configuration attribute), 56</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.exploit_target.ExploitTarget attribute), 54</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.exploit_target.vulnerability.Vulnerability attribute), 57</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.incident.Incident attribute), 72</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.indicator.indicator.Indicator attribute), 81</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.report.header.Header attribute), 89</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.threat_actor.ThreatActor attribute), 91</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.ttp.attack_pattern.AttackPattern attribute), 94</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.ttp.exploit.Exploit attribute), 95</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.ttp.infrastructure.Infrastructure attribute), 96</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.ttp.malware_instance.MalwareInstance attribute), 96</td>
</tr>
<tr>
<td>R</td>
<td>short_description (stix.ttp.TTP attribute), 93</td>
</tr>
<tr>
<td>R</td>
<td>Sighting (class in stix.indicator.sightings), 85</td>
</tr>
<tr>
<td>R</td>
<td>silence_warnings() (in module stix.utils), 86</td>
</tr>
<tr>
<td>R</td>
<td>SimpleMarkingStructure (class in stix.extensions.marking.simple_marking), 67</td>
</tr>
<tr>
<td>R</td>
<td>SnortTestMechanism (class in stix.extensions.test_mechanism.snort_test_mechanism), 69</td>
</tr>
<tr>
<td>R</td>
<td>sorted (stix.common.structured_text.StructuredTextList attribute), 27</td>
</tr>
<tr>
<td>S</td>
<td>Statement (class in stix.common.statement), 25</td>
</tr>
<tr>
<td>S</td>
<td>stix.base (module), 17</td>
</tr>
<tr>
<td>S</td>
<td>stix.campaign (module), 19</td>
</tr>
<tr>
<td>S</td>
<td>stix.coa (module), 51</td>
</tr>
<tr>
<td>S</td>
<td>stix.coa.objective (module), 52</td>
</tr>
<tr>
<td>S</td>
<td>stix.common (module), 21</td>
</tr>
<tr>
<td>S</td>
<td>stix.common.activity (module), 21</td>
</tr>
<tr>
<td>S</td>
<td>stix.common.confidence (module), 21</td>
</tr>
<tr>
<td>S</td>
<td>stix.common.datetime_with_precision (module), 22</td>
</tr>
<tr>
<td>S</td>
<td>stix.common.identity (module), 22</td>
</tr>
<tr>
<td>S</td>
<td>stix.common.information_source (module), 23</td>
</tr>
<tr>
<td>S</td>
<td>stix.common.kill_chains (module), 23</td>
</tr>
<tr>
<td>S</td>
<td>stix.common.related (module), 24</td>
</tr>
</tbody>
</table>

| S       | index 121                                                          |

Index 121
stix.common.statement (module), 25
stix.common.structured_text (module), 26
stix.common.tool (module), 28
stix.common.vocabs (module), 28
stix.core.stix_header (module), 46
stix.core.stix_package (module), 47
stix.core.ttps (module), 50
stix.data_marking (module), 18
stix.exploit_target (module), 53
stix.exploit_target.configuration (module), 55
stix.exploit_target.vulnerability (module), 56
stix.exploit_target.weakness (module), 57
stix.extensions.identity.ciq_identity_3_0 (module), 58
stix.extensions.malware.maec_4_1_malware (module), 61
stix.extensions.marking.ais (module), 61
stix.extensions.marking.simple_marking (module), 67
stix.extensions.marking.terms_of_use_marking (module), 68
stix.extensions.marking.tlp (module), 68
stix.extensions.structured_coa.generic_structured_coa (module), 68
stix.extensions.test_mechanism.generic_test_mechanism (module), 68
stix.extensions.test_mechanism.open_ioc_2010_test_mechanism (module), 69
stix.extensions.test_mechanism.snort_test_mechanism (module), 69
stix.extensions.test_mechanism.yara_test_mechanism (module), 69
stix.incident (module), 70
stix.incident.affected_asset (module), 73
stix.incident.coa (module), 73
stix.incident.contributors (module), 73
stix.incident.direct_impact_summary (module), 74
stix.incident.impact_rating (module), 74
stix.incident.history (module), 74
stix.incident.impact_assessment (module), 74
stix.incident.indirect_impact_summary (module), 75
stix.incident.loss_estimation (module), 75
stix.incident.property_affected (module), 75
stix.incident.time (module), 75
stix.incident.total_loss_estimation (module), 76
stix.indicator.indicator (module), 84
stix.indicator.sightings (module), 85
stix.indicator.test_mechanism (module), 86
stix.indicator.valid_time (module), 86
stix.report (module), 87
stix.report.header (module), 88
stix.threat_actor (module), 89
stix.ttp (module), 91
stix.ttp.attack_pattern (module), 93
stix.ttp.behavior (module), 94
stix.ttp.exploit (module), 94
stix.ttp.exploit_targets (module), 95
stix.ttp.infrastructure (module), 95
stix.ttp.malware_instance (module), 96
stix.ttp.related_ttps (module), 97
stix.ttp.resource (module), 97
stix.ttp.victim_targeting (module), 97
stix.utils (module), 97
stix.utils.dates (module), 98
stix.utils.nsparser (module), 99
stix.utilities.parser (module), 102
STIXCIQIdentity3_0 (class in stix.extensions.identity.ciq_identity_3_0), 58
STIXHeader (class in stix.core.stix_header), 46
STIXPackage (class in stix.core.stix_package), 48
strip_cdata() (in module stix.utils), 98
StructuredText (class in stix.common.structured_text), 26
StructuredTextList (class in stix.common.structured_text), 26
structuring_format (stix.common.structured_text.StructuredText attribute), 26
SubDivisionName (class in stix.extensions.identity.ciq_identity_3_0), 60
SuggestedCOAs (class in stix.indicator.indicator), 84
SystemType (in module stix.common.vocabs), 46
SystemType_1_0 (class in stix.common.vocabs), 43

T

TERM_ACCELERATION
(stix.common.vocabs.AvailabilityLossType_1_0 attribute), 31
TERM_ACCELERATION
(stix.common.vocabs.AvailabilityLossType_1_1_1 attribute), 32
TERM_ACCESS_READER
(stix.common.vocabs.AssetType_1_0 attribute), 28
TERM_ACCOUNT_TAKEOVER
(stix.common.vocabs.IntendedEffect_1_0 attribute), 36
TERM_ACCOUNTABILITY
(stix.common.vocabs.LossProperty_1_0 attribute), 38
TERM_ADMINISTRATOR
(stix.common.vocabs.AssetType_1_0 attribute), 28
TERM_ADVANTAGE
(stix.common.vocabs.IntendedEffect_1_0 attribute), 36
TERM_ADVANTAGE_ECONOMIC
(stix.common.vocabs.IntendedEffect_1_0 attribute), 36
TERM_ADVANTAGE_MILITARY
(stix.common.vocabs.IntendedEffect_1_0 attribute), 36
TERM_ADVANTAGE_POLITICAL (stix.common.vocabs.IntendedEffect_1_0 attribute), 37
TERM_ADWARE (stix.common.vocabs.MalwareType_1_0 attribute), 38
TERM_AGENT_DISCLOSURE (stix.common.vocabs.DiscoveryMethod_1_0 attribute), 33
TERM_AGENT_DISCLOSURE (stix.common.vocabs.DiscoveryMethod_2_0 attribute), 33
TERM_AGENT_DISCLOSURE
TERM_AGGREGATOR (stix.common.vocabs.InformationSourceRole_1_0 attribute), 36
TERM_ANONYMIZATION (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 30
TERM_ANONYMIZATION (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 30
TERM_ANONYMIZATION (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 30
TERM_ANONYMIZATION_PROXY (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 30
TERM_ANONYMIZATION_TUNED_NETWORK (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 30
TERM_ANONYMIZATION_VPN (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 31
TERM_ANTIVIRUS (stix.common.vocabs.DiscoveryMethod_1_0 attribute), 33
TERM_ANTIVIRUS (stix.common.vocabs.DiscoveryMethod_2_0 attribute), 33
TERM_APPLICATION_SCANNER (stix.common.vocabs.AttackerToolType_1_0 attribute), 31
TERM_ASPIRANT (stix.common.vocabs.ThreatActorSophistication attribute), 43
TERM_ATM (stix.common.vocabs.AssetType_1_0 attribute), 28
TERM_ATTACK_PATTERN_CHARACTERIZATION (stix.common.vocabs.PackageIntent_1_0 attribute), 40
TERM_ATTACK_PATTERN_CHARACTERIZATION (stix.common.vocabs.PackageIntent_1_0 attribute), 40
TERM_AUDIT (stix.common.vocabs.DiscoveryMethod_1_0 attribute), 33
TERM_AUDIT (stix.common.vocabs.DiscoveryMethod_2_0 attribute), 33
TERM_AUDITOR (stix.common.vocabs.AssetType_1_0 attribute), 28
TERM_AUTH_TOKEN (stix.common.vocabs.AssetType_1_0 attribute), 28
TERM_AUTHENTICATION_COOKIES (stix.common.vocabs.InformationType_1_0 attribute), 36
TERM_AUTOMATED_TRANSFER_SCRIPTS (stix.common.vocabs.MalwareType_1_0 attribute), 38
TERM_AVAILABILITY (stix.common.vocabs.LossProperty_1_0 attribute), 38
TERM_BACKUP (stix.common.vocabs.AssetTypeInfo_1_0 attribute), 37
TERM_BOT (stix.common.vocabs.MalwareType_1_0 attribute), 38
TERM_BOT_CREDENTIAL_THEFT (stix.common.vocabs.MalwareType_1_0 attribute), 38
TERM_BOT_DDOS (stix.common.vocabs.MalwareType_1_0 attribute), 38
TERM_BOT_LOADER (stix.common.vocabs.MalwareType_1_0 attribute), 38
TERM_BOT_SPAM (stix.common.vocabs.MalwareType_1_0 attribute), 38
TERM_BROADBAND (stix.common.vocabs.AssetType_1_0 attribute), 28
TERM_BROADCAST (stix.common.vocabs.AssetTypeInfo_1_0 attribute), 37
TERM_BROADCAST_OR_IMAGE_DEGRADATION (stix.common.vocabs.IncidentEffect_1_0 attribute), 34
TERM_BROADCASTOR_IMAGE_DEGRADATION
TERM_CALL_CENTER (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_CAMPAIGN_CHARACTERIZATION (stix.common.vocabs.PackageIntent_1_0 attribute), 40
TERM_CAMPAIGN_CHARACTERIZATION (stix.common.vocabs.ReportIntent_1_0 attribute), 42
TERM_CAMPAIGN_CHARACTERIZATION
TERM_CASHIER (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_CAMERA (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_CAMPAIGN_CHARACTERIZATION (stix.common.vocabs.PackageIntent_1_0 attribute), 40
TERM_CAMPAIGN_CHARACTERIZATION
TERM_CATASTROPHIC (stix.common.vocabs.ImpactQualification_1_0 attribute), 34
TERM_CATASTROPHIC
TERM_BRAND_DAMAGE (stix.common.vocabs.IntendedEffect_1_0 attribute), 37
TERM_BRAND_DAMAGE
TERM_BRAND_OR_IMAGE_DEGRADATION (stix.common.vocabs.IncidentEffect_1_0 attribute), 34
TERM_BRAND_OR_IMAGE_DEGRADATION
TERM_CLOSED (stix.common.vocabs.IncidentStatus_1_0 attribute), 35
TERM_COLLECTIVE.THREAT_INTELLIGENCE (stix.common.vocabs.PackageIntent_1_0 attribute), 40
TERM_COLLECTIVE.THREAT_INTELLIGENCE (stix.common.vocabs.ReportIntent_1_0 attribute), 42
TERM_COLOCATED (stix.common.vocabs.LocationClass attribute), 37
TERM_COMANAGEMENT (stix.common.vocabs.ManagementClass_1_0 attribute), 38
TERM_COMMUNICATIONS (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 31
TERM_COMMUNICATIONS_BLOGS (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 31
TERM_COMMUNICATIONS_FORUMS (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 31
TERM_COMMUNICATIONS_INTERNET_RELAY_CHAT (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 31
TERM_COMMUNICATIONS_MICROBLOGS (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 31
TERM_COMMUNICATIONS_MOBILE_COMMUNICATIONS (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 31
TERM_COMMUNICATIONS_SOCIAL_NETWORKS (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 31
TERM_COMMUNICATIONS_USERGENERATED_CONTENT_WEBSITES (stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 31
TERM_COMPETITIVE_ADVANTAGE (stix.common.vocabs.IntendedEffect_1_0 attribute), 37
TERM_COMPROMISED_PKI_Certificate (stix.common.vocabs.IndicatorType_1_1 attribute), 36
TERM_CONFIDENTIALITY (stix.common.vocabs.LossProperty_1_0 attribute), 38
TERM_CONTAINMENT_ACHIEVED (stix.common.vocabs.IncidentStatus_1_0 attribute), 35
TERM_CONTENT_ENHANCERORREFINER (stix.common.vocabs.InformationSourceRole_1_0 attribute), 36
TERM_COURSES_OF_ACTION (stix.common.vocabs.PackageIntent_1_0 attribute), 40
TERM_COURSES_OF_ACTION (stix.common.vocabs.ReportIntent_1_0 attribute), 42
TERM_COURSES_OF_ACTION (stix.common.vocabs.ReportIntent_1_0 at-tribute), 42
TERM_CUSTOMER (stix.common.vocabs.AssetClass_1_0 attribute), 29
TERM_CUSTOMER (stix.common.vocabs.AssetClass_1_0 attribute), 33
TERM_CUSTOMER (stix.common.vocabs.DiscoveryMethod_1_0 attribute), 33
TERM_CUSTOMER (stix.common.vocabs.DiscoveryMethod_2_0 attribute), 33
TERM_CUSTOMEROWNED (stix.common.vocabs.OwnershipClass_1_0 attribute), 40
TERM_CYBER_ESPIONAGE_OPERATIONS (stix.common.vocabs.ThreatActorType_1_0 attribute), 44
TERM_DAMAGE (stix.common.vocabs.ImpactQualification_1_0 attribute), 34
TERM_DATA_BREACH_OR_COMPROMISE (stix.common.vocabs.IncidentEffect_1_0 attribute), 34
TERM_DATA_EXPLOITATION (stix.common.vocabs.PlanningAndOperationalSupport_1_0 attribute), 41
TERM_DATA_EXPLOITATION (stix.common.vocals.PlanningAndOperationalSupport_1_0_1 attribute), 41
TERM_DATA_EXPLOITATION_ANALYTIC_SUPPORT (stix.common.vocals.PlanningAndOperationalSupport_1_0 attribute), 41
TERM_DATA_EXPLOITATION_TRANSLATION_SUPPORT (stix.common.vocals.PlanningAndOperationalSupport_1_0 attribute), 41
TERM_DATABASE (stix.common.vocals.AssetClass_1_0 attribute), 29
TERM_DAYS (stix.common.vocals.LossDuration_1_0 attribute), 37
TERM_DCS (stix.common.vocals.AssetClass_1_0 attribute), 29
TERM_DEGRADATION (stix.common.vocals.AvailabilityLossType_1_1_1 attribute), 32
TERM_DEGRADATION_OF_SERVICE (stix.common.vocals.IncidentEffect_1_0 attribute), 35
TERM_DEGRADATION_OF_SERVICE (stix.common.vocals.IntendedEffect_1_0 attribute), 37
TERM_DEGREDATION
(stix.common.vocabs.AvailabilityLossType_1_0 attribute), 31
TERM_DELETED (stix.common.vocabs.IncidentStatus_1_0 attribute), 35
TERM_DENIAL_AND_DECEPTION
(stix.common.vocabs.IntendedEffect_1_0 attribute), 37
TERM_DENIAL_OF_SERVICE
(stix.common.vocabs.IncidentCategory_1_0 attribute), 34
TERM_DESKTOP (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_DESTRUCTION
(stix.common.vocabs.AvailabilityLossType_1_0 attribute), 31
TERM_DESTRUCTION
(stix.common.vocabs.AvailabilityLossType_1_1_1 attribute), 32
TERM_DESTRUCTION
(stix.common.vocabs.IncidentEffect_1_0 attribute), 35
TERM_DESTRUCTION
(stix.common.vocabs.IntendedEffect_1_0 attribute), 37
TERM_DEVELOPER (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_DHCP (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_DIALER (stix.common.vocabs.MalwareType_1_0 attribute), 38
TERM_DIPLOMATIC_ACTIONS
(stix.common.vocabs.CourseOfActionType_1_0 attribute), 32
TERM_DIRECTORY (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_DISGRUNTLED_CUSTOMER_OR_USER
(stix.common.vocabs.ThreatActorType_1_0 attribute), 44
TERM_DISK_DRIVE (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_DISK_MEDIA (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_DISRUPTION (stix.common.vocabs.IntendedEffect_1_0 attribute), 37
TERM_DISRUPTION_OF_SERVICE
(stix.common.vocabs.IncidentEffect_1_0 attribute), 35
TERM_DISTRACTING (stix.common.vocabs.ImpactQualification_1_0 attribute), 34
TERM_DNS (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_DOCUMENTS (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_DOMAIN_REGISTRATION
(stix.common.vocabs.AttackerInfrastructureType_1_0 attribute), 31
TERM_DOMAIN_REGISTRATION_DYNAMIC_DNS_SERVICES
(stix.common.vocals.AttackerInfrastructureType_1_0 attribute), 31
TERM_DOMAIN_REGISTRATION_LEGITIMATE_DOMAIN_REGISTRATION_SERVICES
(stix.common.vocals.AttackerInfrastructureType_1_0 attribute), 31
TERM_DOMAIN_REGISTRATION_MALICIOUS_DOMAIN_REGISTRARS
(stix.common.vocals.AttackerInfrastructureType_1_0 attribute), 31
TERM_DOMAIN_REGISTRATION_TOPLEVEL_DOMAIN_REGISTRARS
(stix.common.vocals.AttackerInfrastructureType_1_0 attribute), 31
TERM_DOMAIN_WATCHLIST
(stix.common.vocals.IndicatorType_1_0 attribute), 35
TERM_DOMAIN_WATCHLIST
(stix.common.vocals.IndicatorType_1_1 attribute), 36
TERM_DOS_OR_DDOS
(stix.common.vocals.MalwareType_1_0 attribute), 38
TERM_DOS_OR_DDOS_PARTICIPATORY
(stix.common.vocals.MalwareType_1_0 attribute), 38
TERM_DOS_OR_DDOS_SCRIPT
(stix.common.vocals.MalwareType_1_0 attribute), 38
TERM_DOS_OR_DDOS_STRESS_TEST_TOOLS
(stix.common.vocals.MalwareType_1_0 attribute), 38
TERM_ECRIME_ACTOR_CREDENTIAL_THEFT_BOTNET_OPERATOR
(stix.common.vocals.ThreatActorType_1_0 attribute), 44
TERM_ECRIME_ACTOR_CREDENTIAL_THEFT_BOTNET_SERVICE
(stix.common.vocals.ThreatActorType_1_0 attribute), 44
TERM_ECRIME_ACTOR_MALW Are_DEVELOPER
(stix.common.vocals.ThreatActorType_1_0 attribute), 44
TERM_ECRIME_ACTOR_MONEY_LAUNDERING_NETWORK
(stix.common.vocals.ThreatActorType_1_0 attribute), 44
TERM_ECRIME_ACTOR_ORGANIZED_CRIME_ACTOR
(stix.common.vocals.ThreatActorType_1_0 attribute), 44
TERM_ECRIME_ACTOR_SPAM_SERVICE
(stix.common.vocals.ThreatActorType_1_0 attribute), 44
TERM_ECRIME_ACTOR_TRAFFIC_SERVICE
(stix.common.vocals.ThreatActorType_1_0 attribute), 44
TERM_HOURS (stix.common.vocabs.LossDuration_1_0 attribute), 31
TERM_HSM (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_HUMAN_RESOURCES (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_HUMAN_RESOURCES (stix.common.vocabs.IntendedEffect_1_0 attribute), 37
TERM_IDEOLOGICAL (stix.common.vocabs.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL (stix.common.vocabs.Motivation_1_0_1 attribute), 39
TERM_IDEOLOGICAL (stix.common.vocabs.Motivation_1_1 attribute), 39
TERM_IDEOLOGICAL__RELIGIOUS (stix.common.vocabs.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL__ANTI_CORRUPTION (stix.common.vocabs.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocabs.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocabs.Motivation_1_1 attribute), 40
TERM_IDEOLOGICAL__ANTI_CORRUPTION (stix.common.vocabs.Motivation_1_1 attribute), 39
TERM_IDEOLOGICAL__ANTI_CORRUPTION (stix.common.vocabs.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocabs.Motivation_1_1 attribute), 39
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocabs.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocabs.Motivation_1_1 attribute), 40
TERM_IDEOLOGICAL__ANTI_CORRUPTION (stix.common.vocabs.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL__ANTI_CORRUPTION (stix.common.vocabs.Motivation_1_1 attribute), 39
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocabs.Motivation_1_1 attribute), 40
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocabs.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocabs.Motivation_1_1 attribute), 40
TERM_IDEOLOGICAL__ANTI_CORRUPTION (stix.common.vocabs.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL__ANTI_CORRUPTION (stix.common.vocabs.Motivation_1_1 attribute), 39
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocals.Motivation_1_1 attribute), 40
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocals.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocals.Motivation_1_1 attribute), 40
TERM_IDEOLOGICAL__ANTI_CORRUPTION (stix.common.vocals.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL__ANTI_CORRUPTION (stix.common.vocals.Motivation_1_1 attribute), 39
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocals.Motivation_1_1 attribute), 40
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocals.Motivation_1_0 attribute), 39
TERM_IDEOLOGICAL__ANTI_ESTABLISHMENT (stix.common.vocals.Motivation_1_1 attribute), 40
TERM_MILITARY (stix.common.vocabs.Motivation_1_0 attribute), 39
TERM_MILITARY (stix.common.vocabs.Motivation_1_0_1 attribute), 39
TERM_MILITARY (stix.common.vocabs.Motivation_1_1 attribute), 40
TERM_MINOR (stix.common.vocabs.ImpactRating_1_0 attribute), 34
TERM_MINUTES (stix.common.vocabs.LossDuration_1_0 attribute), 37
TERM_MOBILE (stix.common.vocabs.LocationClass_1_0 attribute), 37
TERM_MOBILE_PHONE (stix.common.vocabs.AssetType_1_0 attribute), 29
TERM_MOBILE_SYSTEMS (stix.common.vocabs.SystemType_1_0 attribute), 43
TERM_MOBILE_SYSTEMS_MOBILE_DEVICES (stix.common.vocabs.SystemType_1_0 attribute), 43
TERM_MOBILE_SYSTEMS_MOBILE_OPERATING_SYSTEMS (stix.common.vocabs.SystemType_1_0 attribute), 43
TERM_MOBILE_SYSTEMS_NEAR_FIELD_COMMUNICATIONS (stix.common.vocabs.SystemType_1_0 attribute), 43
TERM_MODERATE (stix.common.vocabs.ImpactRating_1_0 attribute), 34
TERM_MONITORING (stix.common.vocabs.CourseOfActionType_1_0 attribute), 32
TERM_MONITORING_SERVICE (stix.common.vocabs.DiscoveryMethod_1_0 attribute), 33
TERM_MONITORING_SERVICE (stix.common.vocabs.DiscoveryMethod_2_0 attribute), 33
TERM_NETWORK (stix.common.vocabs.AssetType_1_0 attribute), 30
TERM_NEW (stix.common.vocabs.IncidentStatus_1_0 attribute), 35
TERM_NIDS (stix.common.vocabs.DiscoveryMethod_1_0 attribute), 33
TERM_NIDS (stix.common.vocabs.DiscoveryMethod_2_0 attribute), 33
TERM_NO (stix.common.vocabs.SecurityCompromise_1_0 attribute), 42
TERM_NONREPUDIATION (stix.common.vocabs.LossProperty_1_0 attribute), 38
TERM_NOVICE (stix.common.vocabs.ThreatActorSophistication_1_0 attribute), 43
TERM_OBSCUSSION (stix.common.vocabs.AvailabilityLossType_1_0 attribute), 32
TERM_OBSCUSSION (stix.common.vocabs.AvailabilityLossType_1_1_1 attribute), 32
TERM_OBSERVATIONS (stix.common.vocabs.PackageIntent_1_0 attribute), 40
TERM_OBSERVATIONS (stix.common.vocabs.ReportIntent_1_0 attribute), 42
TERM_OBSERVATIONS_EMAIL (stix.common.vocabs.PackageIntent_1_0 attribute), 40
TERM_OBSERVATIONS_EMAIL (stix.common.vocabs.ReportIntent_1_0 attribute), 42
TERM_ONGOING (stix.common.vocabs.CampaignStatus_1_0 attribute), 32
TERM_OPEN (stix.common.vocabs.IncidentStatus_1_0 attribute), 35
TERM_OPPORTUNISTIC (stix.common.vocabs.Motivation_1_0 attribute), 39
TERM_OPPORTUNISTIC (stix.common.vocabs.Motivation_1_0_1 attribute), 39
TERM_OPPORTUNISTIC (stix.common.vocabs.Motivation_1_1 attribute), 40
TERM_OPPORTUNISTIC (stix.common.vocabs.Motivation_1_1_0 attribute), 40
TERM_PAINFUL (stix.common.vocabs.ImpactQualification_1_0 attribute), 34
TERM_PARTNER (stix.common.vocabs.AssetType_1_0 attribute), 30
TERM_PARTNEROWNED (stix.common.vocabs.OwnershipClass_1_0 attribute), 40
TERM_PASSWORD_CRACKING (stix.common.vocabs.AttackerToolType_1_0 attribute), 31
TERM_PATCHING (stix.common.vocabs.CourseOfActionType_1_0 attribute), 32
TERM_PAYMENT_CARD (stix.common.vocabs.AssetType_1_0 attribute), 30
TERM_PAYMENT_SWITCH (stix.common.vocabs.AssetType_1_0 attribute), 30
TERM_NONREPUDIATION (stix.common.vocabs.LossProperty_1_0 attribute), 38
TERM_PBX (stix.common.vocabs.AssetType_1_0 attribute), 30
<table>
<thead>
<tr>
<th>Term</th>
<th>Module</th>
<th>Attribute</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>TERM_PED_PAD</td>
<td>stix.common.vocabs.AssetType_1_0</td>
<td>attribute</td>
<td>30</td>
</tr>
<tr>
<td>TERM_PERIMETER_BLOCKING</td>
<td>stix.common.vocabs.CourseOfActionType_1_0</td>
<td>attribute</td>
<td>31</td>
</tr>
<tr>
<td>TERM_PERIPHERAL</td>
<td>stix.common.vocabs.AssetType_1_0</td>
<td>attribute</td>
<td>30</td>
</tr>
<tr>
<td>TERM_PERMANENT</td>
<td>stix.common.vocabs.LossDuration_1_0</td>
<td>attribute</td>
<td>37</td>
</tr>
<tr>
<td>TERM_PERSON</td>
<td>stix.common.vocabs.AssetType_1_0</td>
<td>attribute</td>
<td>30</td>
</tr>
<tr>
<td>TERM_PHYSICAL_ACCESS_RESTRICTIONS</td>
<td>stix.common.vocabs.CourseOfActionType_1_0</td>
<td>attribute</td>
<td>32</td>
</tr>
<tr>
<td>TERM_PLANNING</td>
<td>stix.common.vocabs.PlanningAndOperationalSupport_1_0</td>
<td>attribute</td>
<td>41</td>
</tr>
<tr>
<td>TERM_PLANNING</td>
<td>stix.common.vocabs.PlanningAndOperationalSupport_1_0</td>
<td>attribute</td>
<td>41</td>
</tr>
<tr>
<td>TERM_PLANNING_OPEN_SOURCE_INTELLIGENCE_OSIINT_GATHERING</td>
<td>stix.common.vocabs.PlanningAndOperationalSupport_1_0</td>
<td>attribute</td>
<td>41</td>
</tr>
<tr>
<td>TERM_PRACTITIONAN</td>
<td>stix.common.vocabs.ThreatActorSophistication_1_0</td>
<td>attribute</td>
<td>39</td>
</tr>
<tr>
<td>TERM_PROXY</td>
<td>stix.common.vocabs.AssetType_1_0</td>
<td>attribute</td>
<td>30</td>
</tr>
<tr>
<td>TERM_PUBLIC_WAN</td>
<td>stix.common.vocabs.AssetType_1_0</td>
<td>attribute</td>
<td>30</td>
</tr>
<tr>
<td>TERM_RANSOMWARE</td>
<td>stix.common.vocabs.MalwareType_1_0</td>
<td>attribute</td>
<td>30</td>
</tr>
<tr>
<td>TERM_REBUILDING</td>
<td>stix.common.vocabs.CourseOfActionType_1_0</td>
<td>attribute</td>
<td>38</td>
</tr>
<tr>
<td>TERM_REDIRECTION</td>
<td>stix.common.vocabs.CourseOfActionType_1_0</td>
<td>attribute</td>
<td>32</td>
</tr>
<tr>
<td>TERM_REDIRECTION_HONEY_POT</td>
<td>stix.common.vocabs.CourseOfActionType_1_0</td>
<td>attribute</td>
<td>32</td>
</tr>
<tr>
<td>TERM_REGULATORY_COMPLIANCE_OR_LEGAL_IMPACT</td>
<td>stix.common.vocabs.IncidentEffect_1_0</td>
<td>attribute</td>
<td>35</td>
</tr>
<tr>
<td>TERM_REJECTED</td>
<td>stix.common.vocabs.IncidentStatus_1_0</td>
<td>attribute</td>
<td>35</td>
</tr>
<tr>
<td>TERM_REMEDY</td>
<td>stix.common.vocabs.COAStage_1_0</td>
<td>attribute</td>
<td>32</td>
</tr>
<tr>
<td>TERM_REMOTE_ACCESS</td>
<td>stix.common.vocabs.AssetType_1_0</td>
<td>attribute</td>
<td>32</td>
</tr>
<tr>
<td>TERM_REMOTE_ACCESS_TROJAN</td>
<td>stix.common.vocabs.MalwareType_1_0</td>
<td>attribute</td>
<td>38</td>
</tr>
<tr>
<td>TERM_POLICY_ACTIONS</td>
<td>stix.common.vocabs.CourseOfActionType_1_0</td>
<td>attribute</td>
<td>32</td>
</tr>
<tr>
<td>TERM_RESPONSE</td>
<td>stix.common.vocabs.COAStage_1_0</td>
<td>attribute</td>
<td>32</td>
</tr>
</tbody>
</table>
TERM_SKILL_DEVELOPMENT_OR_RECRUITMENT_UNIVERSITY_PROGRAMS
(stix.common.vocabs.PlanningAndOperationalSupport_1_0 attribute), 42
TERM_SKILL_DEVELOPMENT_RECRUITMENT
(stix.common.vocabs.PlanningAndOperationalSupport_1_0 attribute), 41
TERM_SKILL_DEVELOPMENT_RECRUITMENT_CONTRACTING_AND_HIRING
(stix.common.vocabs.PlanningAndOperationalSupport_1_0 attribute), 41
TERM_SKILL_DEVELOPMENT_RECRUITMENT_DOCUMENT_EXPLOITATION_DOCEX_TRAINING
(stix.common.vocabs.PlanningAndOperationalSupport_1_0 attribute), 41
TERM_SKILL_DEVELOPMENT_RECRUITMENT_INTERNAL_TRAINING
(stix.common.vocabs.PlanningAndOperationalSupport_1_0 attribute), 41
TERM_SKILL_DEVELOPMENT_RECRUITMENT_SECURITY_HACKER_CONFERENCES
(stix.common.vocabs.PlanningAndOperationalSupport_1_0 attribute), 41
TERM_SKILL_DEVELOPMENT_RECRUITMENT_UNDERGROUND_FORUMS
(stix.common.vocabs.PlanningAndOperationalSupport_1_0 attribute), 41
TERM_SKILL_DEVELOPMENT_RECRUITMENT_UNIVERSITY_PROGRAMS
(stix.common.vocabs.PlanningAndOperationalSupport_1_0 attribute), 41
TERM_SMART_CARD
(stix.common.vocabs.AssetType_1_0 attribute), 30
TERM_STALLED
(stix.common.vocabs.IncidentStatus_1_0 attribute), 35
TERM_STATE_ACTOR_OR_AGENCY
(stix.common.vocabs.ThreatActorType_1_0 attribute), 44
TERM_SUSPECTED
(stix.common.vocabs.SecurityCompromise_1_0 attribute), 42
TERM_TELEPHONE
(stix.common.vocabs.AssetType_1_0 attribute), 30
TERM_THEFT
(stix.common.vocabs.IntendedEffect_1_0 attribute), 37
TERM_THEFT_CREDENTIAL_THEFT
(stix.common.vocabs.IntendedEffect_1_0 attribute), 37
TERM_THEFT_IDENTITY_THEFT
(stix.common.vocabs.IntendedEffect_1_0 attribute), 37
TERM_THEFT_INTELLECTUAL_PROPERTY
(stix.common.vocabs.IntendedEffect_1_0 attribute), 37
TERM_THEFT_THEFT_OF_PROPRIETARY_INFORMATION
(stix.common.vocabs.IntendedEffect_1_0 attribute), 37

Index 133
<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TERM_THIRDPARTY_SERVICES</td>
<td>(stix.common.vocabs.SystemType_1_0 attribute), 43</td>
</tr>
<tr>
<td>TERM_THIRDPARTY_SERVICES_APPLICATION_STORES</td>
<td>(stix.common.vocabs.SystemType_1_0 attribute), 43</td>
</tr>
<tr>
<td>TERM_THIRDPARTY_SERVICES_CLOUD_SERVICES</td>
<td>(stix.common.vocabs.SystemType_1_0 attribute), 43</td>
</tr>
<tr>
<td>TERM_THIRDPARTY_SERVICES_SECURITY_VENDORS</td>
<td>(stix.common.vocabs.SystemType_1_0 attribute), 43</td>
</tr>
<tr>
<td>TERM_THREAT_ACTOR_CHARACTERIZATION</td>
<td>(stix.common.vocabs.PackageIntent_1_0 attribute), 40</td>
</tr>
<tr>
<td>TERM_THREAT_ACTOR_CHARACTERIZATION</td>
<td>(stix.common.vocabs.ReportIntent_1_0 attribute), 42</td>
</tr>
<tr>
<td>TERM_THREAT_REPORT</td>
<td>(stix.common.vocabs.PackageIntent_1_0 attribute), 40</td>
</tr>
<tr>
<td>TERM_THREAT_REPORT</td>
<td>(stix.common.vocabs.ReportIntent_1_0 attribute), 42</td>
</tr>
<tr>
<td>TERM_TRAFFIC_DIVERSION</td>
<td>(stix.common.vocabs.IntendedEffect_1_0 attribute), 37</td>
</tr>
<tr>
<td>TERM_TRAFFIC_SCANNER</td>
<td>(stix.common.vocabs.AttackerToolType_1_0 attribute), 31</td>
</tr>
<tr>
<td>TERM_TRAINING</td>
<td>(stix.common.vocabs.CourseOfActionType_1_0 attribute), 33</td>
</tr>
<tr>
<td>TERM_TRANSFORMERORTRANSLATOR</td>
<td>(stix.common.vocabs.InformationSourceRole_1_0 attribute), 36</td>
</tr>
<tr>
<td>TERM_TTP_INFRASTRUCTURE</td>
<td>(stix.common.vocabs.PackageIntent_1_0 attribute), 40</td>
</tr>
<tr>
<td>TERM_TTP_INFRASTRUCTURE</td>
<td>(stix.common.vocabs.ReportIntent_1_0 attribute), 42</td>
</tr>
<tr>
<td>TERM_TTP_TOOLS</td>
<td>(stix.common.vocabs.PackageIntent_1_0 attribute), 40</td>
</tr>
<tr>
<td>TERM_TTP_TOOLS</td>
<td>(stix.common.vocabs.ReportIntent_1_0 attribute), 42</td>
</tr>
<tr>
<td>TERM_URL_WATCHLIST</td>
<td>(stix.common.vocabs.IndicatorType_1_0 attribute), 35</td>
</tr>
<tr>
<td>TERM_URL_WATCHLIST</td>
<td>(stix.common.vocabs.IndicatorType_1_1 attribute), 36</td>
</tr>
<tr>
<td>TERM_USER</td>
<td>(stix.common.vocabs.DiscoveryMethod_1_0 attribute), 33</td>
</tr>
<tr>
<td>TERM_UNSIGNED</td>
<td>(stix.common.vocabs.DiscoveryMethod_2_0 attribute), 34</td>
</tr>
</tbody>
</table>

**Index**

- [Index](#)
TERM_USER (stix.common.vocabs.DiscoveryMethod_2_0 attribute), 34
TERM_USER_DATA_LOSS (stix.common.vocabs.IncidentEffect_1_0 attribute), 35
TERM_USER_DEVICE (stix.common.vocabs.AssetType_1_0 attribute), 30
TERM_USERS (stix.common.vocabs.SystemType_1_0 attribute), 43
TERM_USERS_APPLICATION_AND_SOFTWARE (stix.common.vocabs.SystemType_1_0 attribute), 43
TERM_USERS_REMOVABLE_MEDIA (stix.common.vocabs.SystemType_1_0 attribute), 43
TERM_USERS_WORKSTATION (stix.common.vocabs.SystemType_1_0 attribute), 43
TERM_VOIP_ADAPTER (stix.common.vocabs.AssetType_1_0 attribute), 30
TERM_VOIP_PHONE (stix.common.vocabs.AssetType_1_0 attribute), 30
TERM_VULNERABILITY_SCANNER (stix.common.vocabs.AttackerToolType_1_0 attribute), 31
TERM_WEB_APPLICATION (stix.common.vocabs.AssetType_1_0 attribute), 30
TERM_WEEKS (stix.common.vocabs.LossDuration_1_0 attribute), 37
TERM_WLAN (stix.common.vocabs.AssetType_1_0 attribute), 30
TERM_YES (stix.common.vocabs.SecurityCompromise_1_0 attribute), 43
TermsOfUseMarkingStructure (class in stix.extensions.marking.terms_of_use_marking), 68
ThreatActor (class in stix.threat_actor), 90
ThreatActorSophistication (in module stix.common.vocabs), 46
ThreatActorSophistication_1_0 (class in stix.common.vocabs), 43
ThreatActorType (in module stix.common.vocabs), 46
ThreatActorType_1_0 (class in stix.common.vocabs), 44
Time (class in stix.incident.time), 76
TIME_PRECISION_VALUES (in module stix.common.datetime Precision), 22
title (stix.core.stix_header.STIXHeader attribute), 47
title (stix.report.header.Header attribute), 88
TLPMarkingStructure (class in stix.extensions.marking.tlp), 68
to_dict() (stix.coa.CourseOfAction method), 52
to_dict() (stix.common.structured_text.StructuredText method), 26
to_dict() (stix.common.structured_text.StructuredTextList method), 27
to_dict() (stix.core.stix_package.STIXPackage method), 49
to_dict() (stix.exploit_target.ExploitTarget method), 55
to_dict() (stix.incident.Incident method), 72
to_dict() (stix.indicator.indicator.Indicator method), 81
to_dict() (stix.threat_actor.ThreatActor method), 91
to_dict() (stix.ttp.TTP method), 93
to_json() (stix.campaign.Campaign method), 20
to_json() (stix.coa.CourseOfAction method), 52
to_json() (stix.core.stix_package.STIXPackage method), 49
to_json() (stix.exploit_target.ExploitTarget method), 55
to_json() (stix.incident.Incident method), 72
to_json() (stix.indicator.indicator.Indicator method), 81
to_json() (stix.threat_actor.ThreatActor method), 91
to_json() (stix.ttp.TTP method), 93
to_obj() (stix.campaign.Campaign method), 20
to_obj() (stix.coa.CourseOfAction method), 52
to_obj() (stix.common.structured_text.StructuredText method), 26
to_obj() (stix.common.structured_text.StructuredTextList method), 27
to_obj() (stix.core.stix_package.STIXPackage method), 49
to_obj() (stix.exploit_target.ExploitTarget method), 55
to_obj() (stix.incident.Incident method), 72
to_obj() (stix.indicator.indicator.CompositeIndicatorExpression method), 82
to_obj() (stix.indicator.indicator.Indicator method), 81
to_obj() (stix.indicator.indicator.Indicator method), 81
to_obj() (stix.indicator.indicator.Indicator method), 81
to_obj() (stix.indicator.indicator.Indicator method), 81
to_obj() (stix.indicator.indicator.Indicator method), 81
to_obj() (stix.indicator.indicator.Indicator method), 81
to_obj() (stix.indicator.indicator.Indicator method), 81
to_obj() (stix.indicator.indicator.Indicator method), 81
to_obj() (stix.indicator.indicator.Indicator method), 81
TotalLossEstimation (class in stix.core.tools), 28
TLPMarkingStructure (class in stix.extensions.marking.tlp), 68
to_dict() (stix.campaign.Campaign method), 20

Index 135
UnknownVersionError (class in stix.utils.parser), 102
UnsupportedRootElement (in module stix.utils.parser), 102
UnsupportedVersionError (class in stix.utils.parser), 102
update() (stix.common.structured_text.StructuredTextList method), 27
ValidTime (class in stix.indicator.valid_time), 86
value (stix.common.structured_text.StructuredText attribute), 26
Versioning_1_0 (class in stix.common.vocabs), 44
VictimTargeting (class in stix.ttp.victim_targeting), 97
VocabString (class in stix.common.vocabs), 44
Vulnerability (class in stix.exploit_target.vulnerability), 56
WATER_AND_WASTEWATER_SYSTEMS_SECTOR (in module stix.extensions.marking.ais), 67
Weakness (class in stix.exploit_target.weakness), 58
xml_bool() (in module stix.utils), 98
XML_NS_STIX_EXT (in module stix.extensions.identity.ciq_identity_3_0), 60
XML_NS_XAL (in module stix.extensions.identity.ciq_identity_3_0), 60
XML_NS_XNL (in module stix.extensions.identity.ciq_identity_3_0), 60
XML_NS_XPIL (in module stix.extensions.identity.ciq_identity_3_0), 60
YaraTestMechanism (class in stix.extensions.test_mechanism.yara_test_mechanism), 69