STIX 2

Data Model Foundations

Training Introduction

- Module 1: STIX 2.0 Data Model Overview
- Module 2: STIX 2.0 Data Model Deep Dive
- Module 3: Working with STIX 2.0 Objects
- Module 4: STIX 2.0 Patterning Core Concepts

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

STIX 2

Module 1: Data Model Overview

OASIS

Module 1: Training Objectives

- In this module, you will:
  - Gain an understanding of the STIX 2.0 use cases and design principles
  - Begin to understand the high-level design of STIX 2.0 based on its set of objects and relationships

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Use Cases

- Analyzing & Sharing Cyber Threat Intelligence (CTI)
  - TTPs
  - Adversary Information
  - Indicators
  - Campaigns & Intrusion Sets
- Supporting the CTI Lifecycle
  - Sharing Sightings
- Managing Cyber Threat Response Activities
  - Cyber Threat Prevention
  - Cyber Threat Detection
  - Incident Response

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

STIX 2.0 vs STIX 1.x Highlights

- **JSON, not XML**: Preferred by developers, easier to understand
- **Simplicity and Clarity**: Less flexibility, more standardization
- **Pragmatism**: Fewer, but better-understood objects and properties
- **One Standard**: Merge CybOX into STIX
- **Relationships as first-class objects**: Easier for the community to contribute
- **Graph-based data model**: Better enables the use of STIX objects in the CTI lifecycle

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

STIX 2.0 by the Numbers

- **STIX 2.0 defines:**
  - 12 Domain Objects
  - 2 Relationship Objects
  - 16 Cyber Observable Objects
  - 11 Common Data Types
  - 5 Cyber Observable-specific Data Types
  - 14 Vocabularies

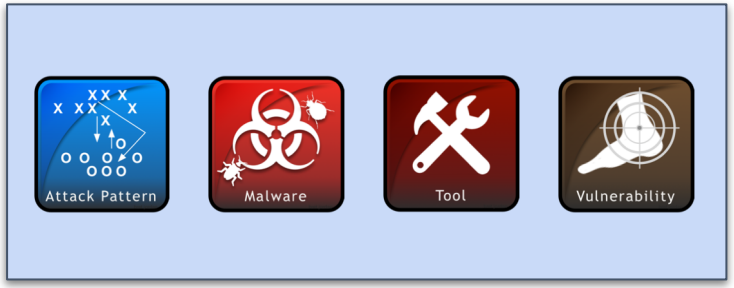
Building blocks for Cyber Threat Intelligence

Image credit: https://isaacmorehouse.com/wp-content/uploads/2015/06/legos1-1024x544.jpg

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

STIX 2.0 Domain Objects (SDOs)

Adversary Objects



TTP Objects

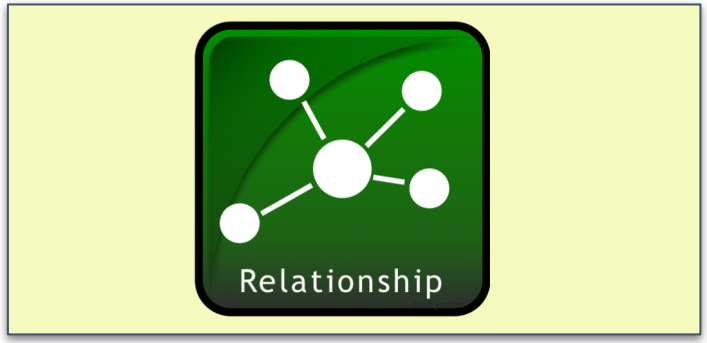
Supporting Objects

Remediation Objects

Detection Objects

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

STIX 2.0 Relationship Objects (SROs)

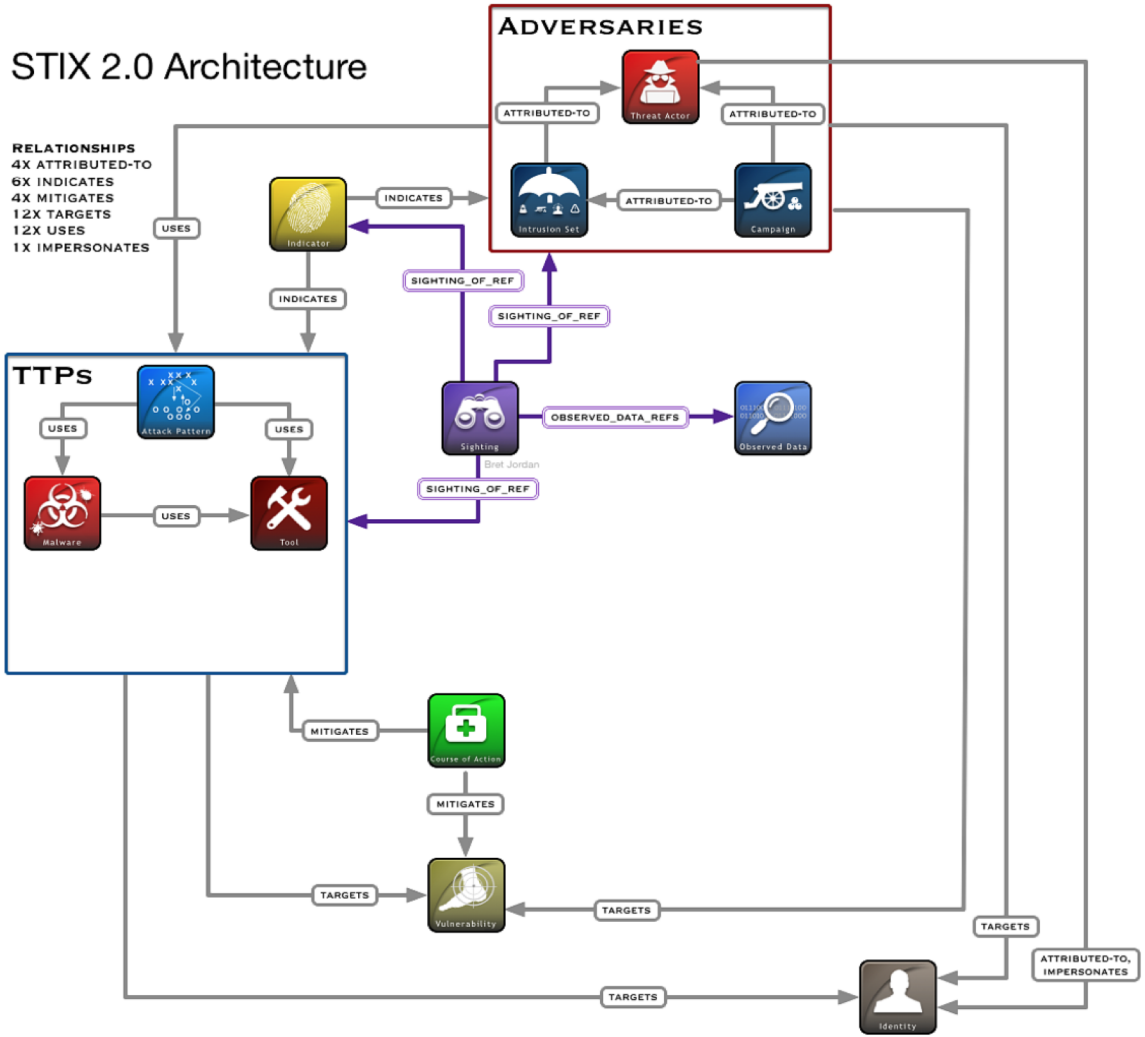


Standard Relationship Objects

Special Relationship Objects

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

STIX 2.0 Architecture



STIX 2.0 Cyber Observable Objects

- STIX 2.0 includes a set of Cyber Observables which provide a data model for describing *things* you've either *actually seen*, or *are looking for:*
  - Artifact
  - AS
  - Directory
  - Email Address
  - Email Message
  - File
    - Archive Extension
    - NTFS File Extension
    - PDF File Extension
    - Raster Image File Extension
    - Windows PE Binary File Extension
  - IPv4 Address
  - IPv6 Address
  - MAC Address
  - Mutex
  - Network Traffic
    - HTTP Request Extension
    - ICMP Extension
    - Network Socket Extension
    - TCP Extension
  - Process
    - Windows Process Extension
    - Windows Service Extension
  - Software
  - User Account
    - UNIX Account Extension
  - Windows Registry Key
  - X.509 Certificate

Several examples are included for each object and extension in the STIX specification.

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

STIX 2.0 Data Model Design Methodology

- **Less abstraction, more granularity**
  - For better semantic consistency and interoperability, STIX defines a larger number of more granular domain objects.
- **Don't reinvent the wheel**
  - STIX leverages widely-used concepts in CTI, like TLP data markings, as much as possible.
- **Consistent object definitions**
  - Each STIX object has an identical set of common properties for entities like IDs, labels, external references, etc.
- **Support customization**
  - STIX recognizes that the standard data model doesn't cover everything, and therefore has native support for custom objects and properties.
- **Support the object lifecycle**
  - STIX includes native support for object versioning and revocation.

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Module 1: Checking In

- Q: How is STIX 2.0 serialized?
- Q: Name two STIX 2.0 Domain Objects?
- Q: Which objects would you use to describe the use of Malware FooBot by an APT group?

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

STIX 2

Module 2: Data Model Deep Dive

OASIS

Module 2: Training Objectives

- In this module, you will:
  - Gain an understanding of the purpose and available default relationships for each STIX Domain Object
  - See how the graph-based STIX Data Model can be used to relate STIX Domain Objects through the use of STIX Relationship Objects

OASIS

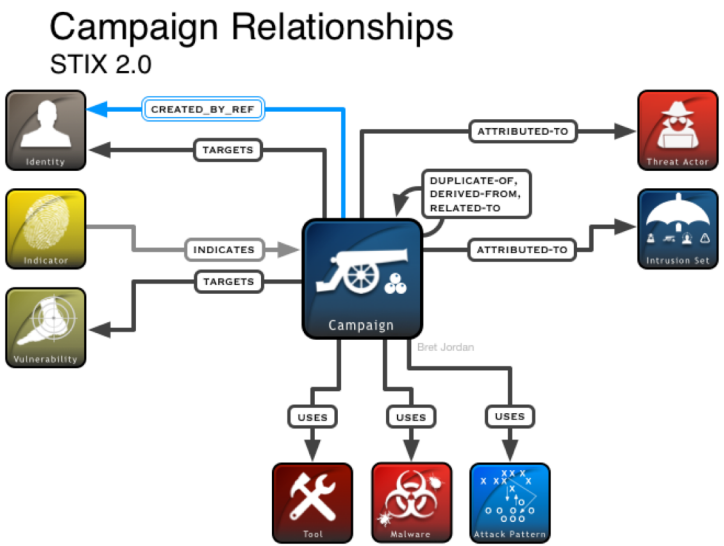
Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Adversary SDOs: Campaign

A Campaign describes a set of malicious activities that occur over a period of time against a specific set of targets. These activities usually are composed of specific behaviors, have well defined objectives, and may be part of an Intrusion Set.

Examples

- Campaign by Green Group against a set of targets in the financial services sector
- Campaign by Red Group against a set of targets in the critical infrastructure sector and leveraging the BadNewz Malware Family



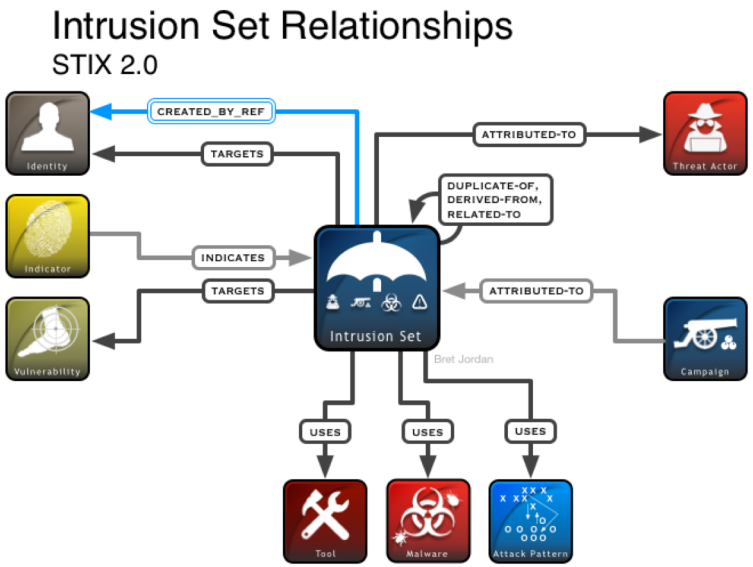
Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Adversary SDOs: Intrusion Set

An Intrusion Set is a grouped set of adversarial behaviors and resources with common properties that is believed to be orchestrated by a single organization. They may capture multiple Campaigns that are all tied together by shared attributes indicating a common Threat Actor.

Examples

- A set of Campaigns perpetrated over multiple years by the same Threat Actor in order to achieve a diverse set of objectives



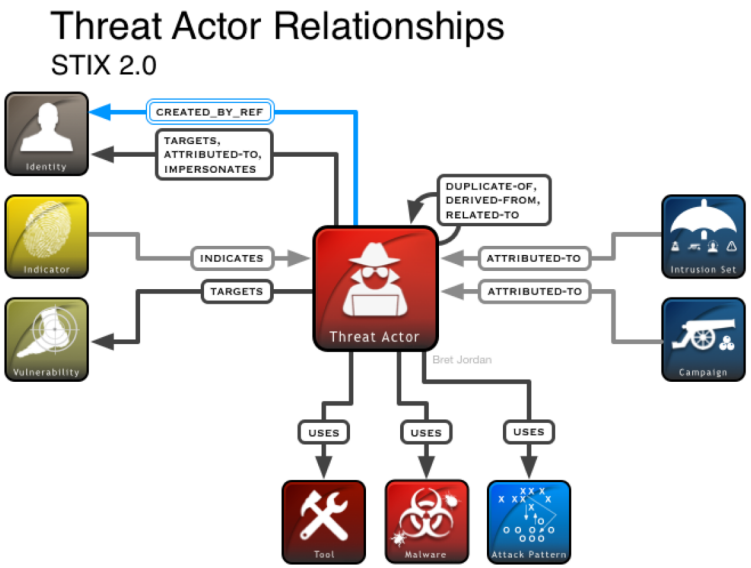
Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Adversary SDOs: Threat Actor

Threat Actors are actual individuals, groups, or organizations believed to be operating with malicious intent.

Examples

- Evil Org, an organization
- John Doe, a malware author associated with Evil Org and several other criminal groups



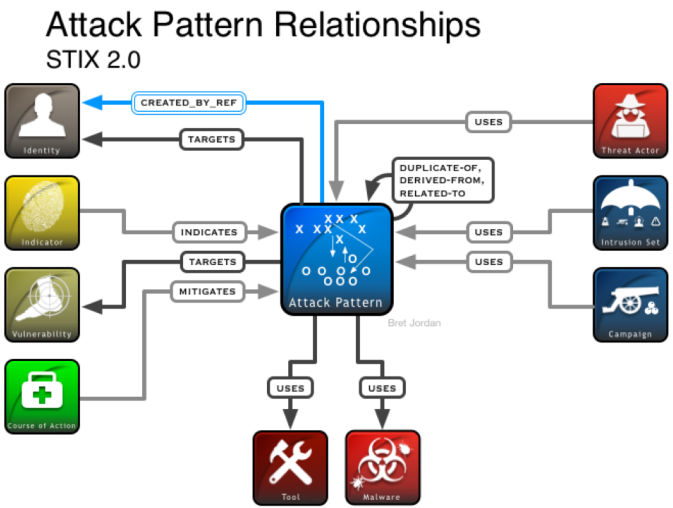
Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

TTP SDOs: Attack Pattern

Attack Patterns describe ways that adversaries attempt to compromise targets. They're used to help generalize specific attacks to the patterns that they follow and provide detailed information about how attacks are performed.

Examples

- Spear Phishing (generic)
- Spear Phishing as practiced by APT3
- Privilege Escalation
- DLL Injection
- UDP Flood
- DNS Spoofing



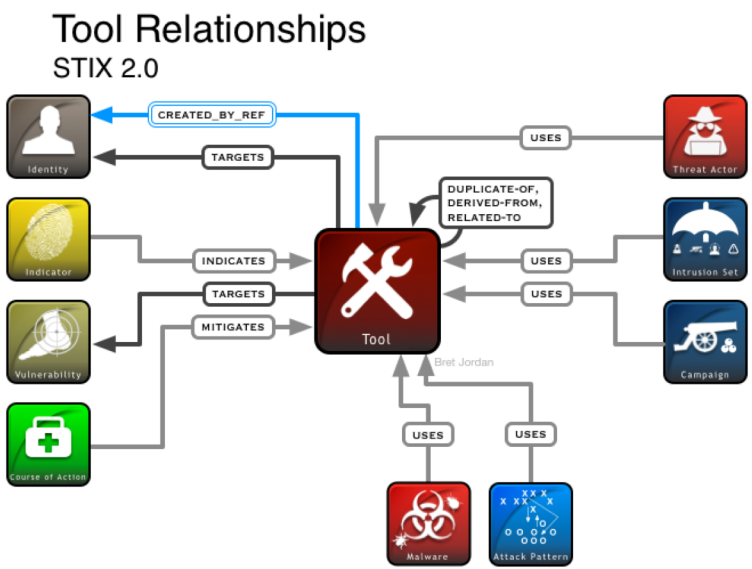
Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

TTP SDOs: Tool

Tools are legitimate software (and therefore distinct from malware) that can be used by threat actors to perform attacks.

Examples

- NMap, used by Threat Actor Foo for network mapping
- NetCat, used by Threat Actor Bar for TCP tunneling



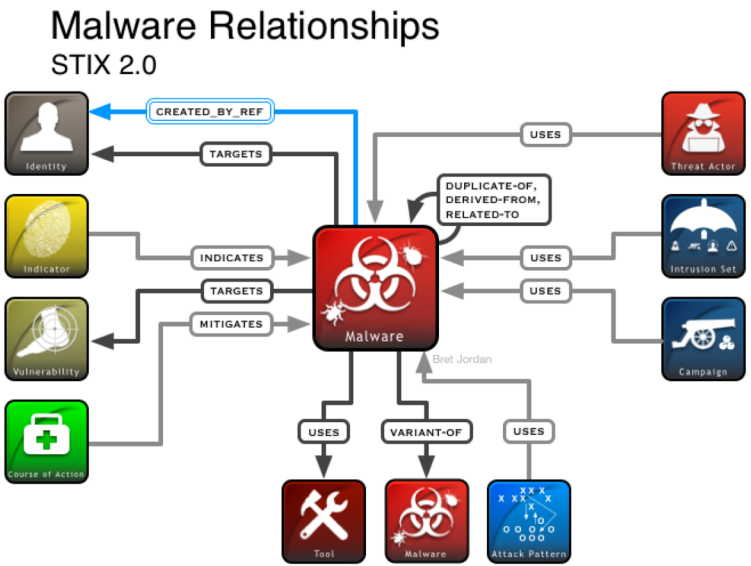
Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

TTP SDOs: Malware

Malware characterizes Malware Instances or Malware Families, including identifying information, metadata, and data that may be derived from various forms of malware analysis.

Examples

- Poison Ivy (Malware Family)
- Trojan.Zeus.AZ (Malware Instance)



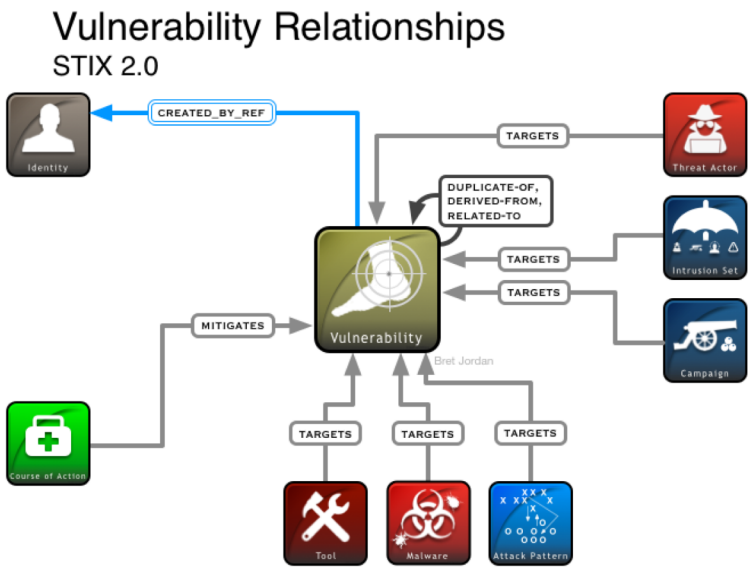
Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

TTP SDOs: Vulnerability

A Vulnerability is, according to CVE, "a mistake in software that can be directly used by a hacker to gain access to a system or network”.

Examples

- A Malware Instance exploits a Vulnerability identified as CVE-2015-12345
- Campaign Foo is known to target a specific Vulnerability for exploitation



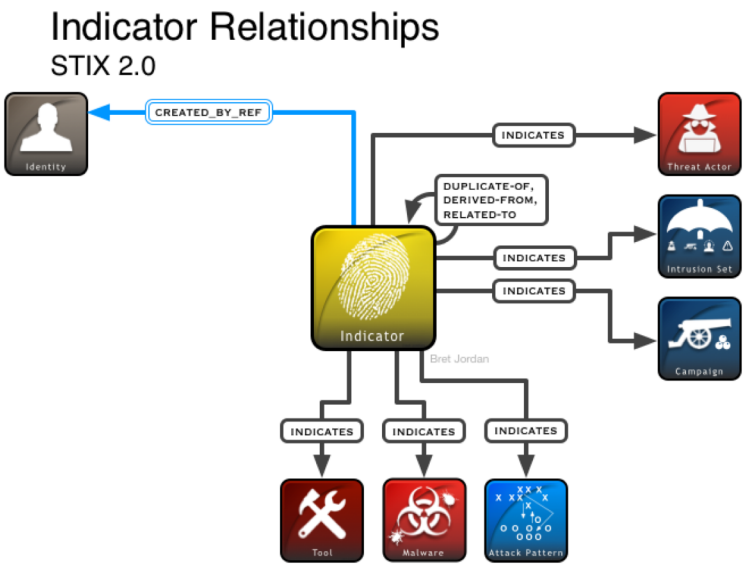
Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Detection SDOs: Indicator

Indicators contain a pattern that can be used to detect suspicious or malicious cyber activity. They use the STIX Patterning Language for specifying the actual syntax and content of the patterns.

Examples

- A SHA-256 hash that detects a particular Malware Instance
- An email address that detects attempted Spear Phishing by a particular Campaign



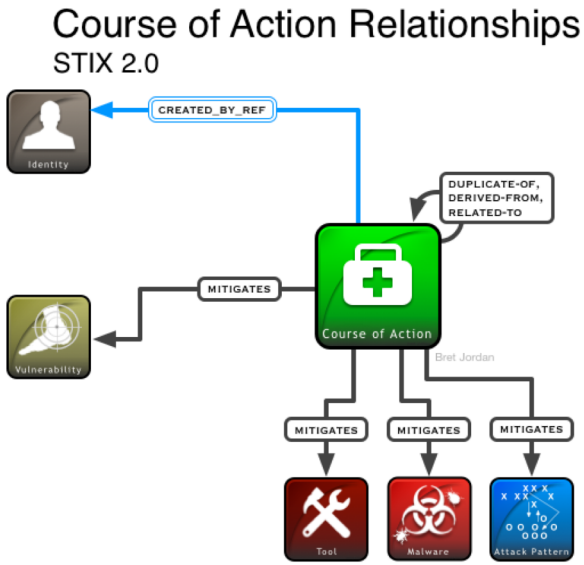
Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Remediation SDOs: Course of Action

A Course of Action is an action taken either to prevent an attack or to respond to an attack that is in progress. It may describe technical, automatable responses or higher level actions like employee training or policy changes.

Examples

- Adding a new TCP Filter rule to a Firewall
- Quarantining all files associated with a particular Malware Family
- Applying a particular software patch



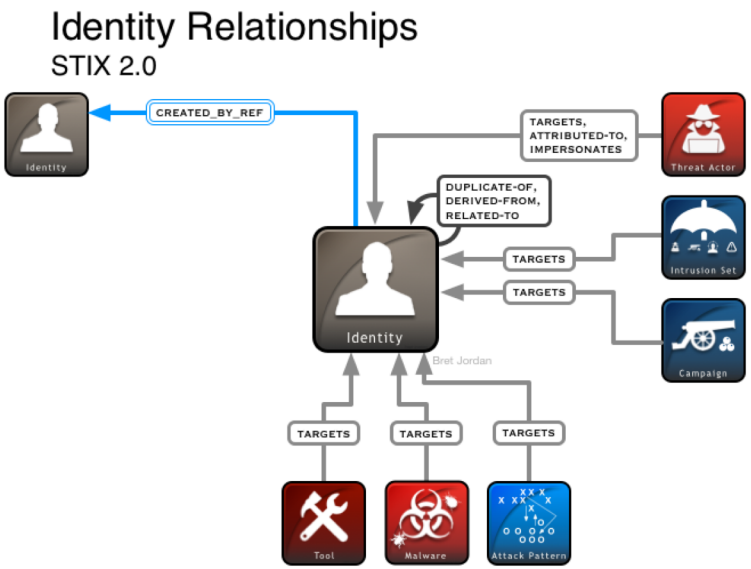
Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Supporting SDOs: Identity

Identities can represent actual individuals, organizations, or groups as well as classes of individuals, organizations, or groups.

Examples

- An Individual named John Smith
- A Company named ACME, Inc.
- The Healthcare Industry Sector



Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Supporting SDOs: Observed Data

Observed Data conveys information that was observed on systems and networks, and makes use of the data models developed as part of STIX Cyber Observables for characterizing this information.

Examples

- An Observation of a particular File, as represented a File Object with a set of hashes and a file name
- An Observation of a particular domain name of unknown provenance, as represented by a Domain Name Object

Relationships

- N/A – there are no top-level relationships between the Observed Data Object and other Objects.
- Instead, Observed Data is a direct target of the Sighting STIX Relationship Object.

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Supporting SDOs: Report

Reports are collections of threat intelligence focused on one or more topics. They are used to group related threat intelligence together so that it can be published as a comprehensive cyber threat story.

Examples

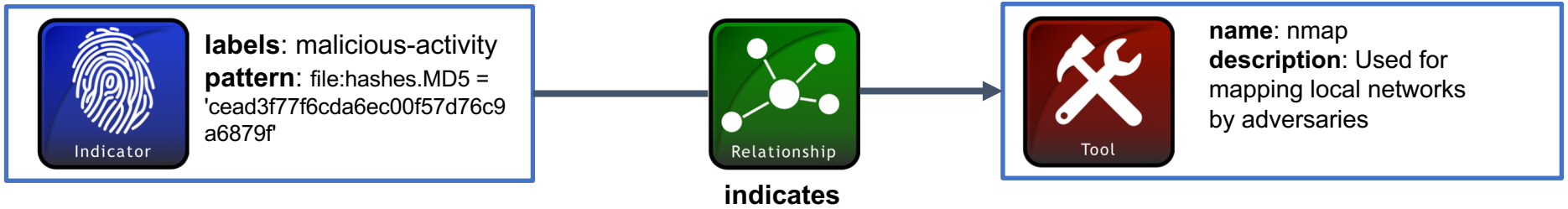
- A Report produced by ACME Defense, Inc. about the Purple Gorilla Campaign

Relationships

- N/A – there are no top-level relationships between the Report Object and other Objects.
- Instead, the Report Object has an embedded relationship that can point to any STIX Object.

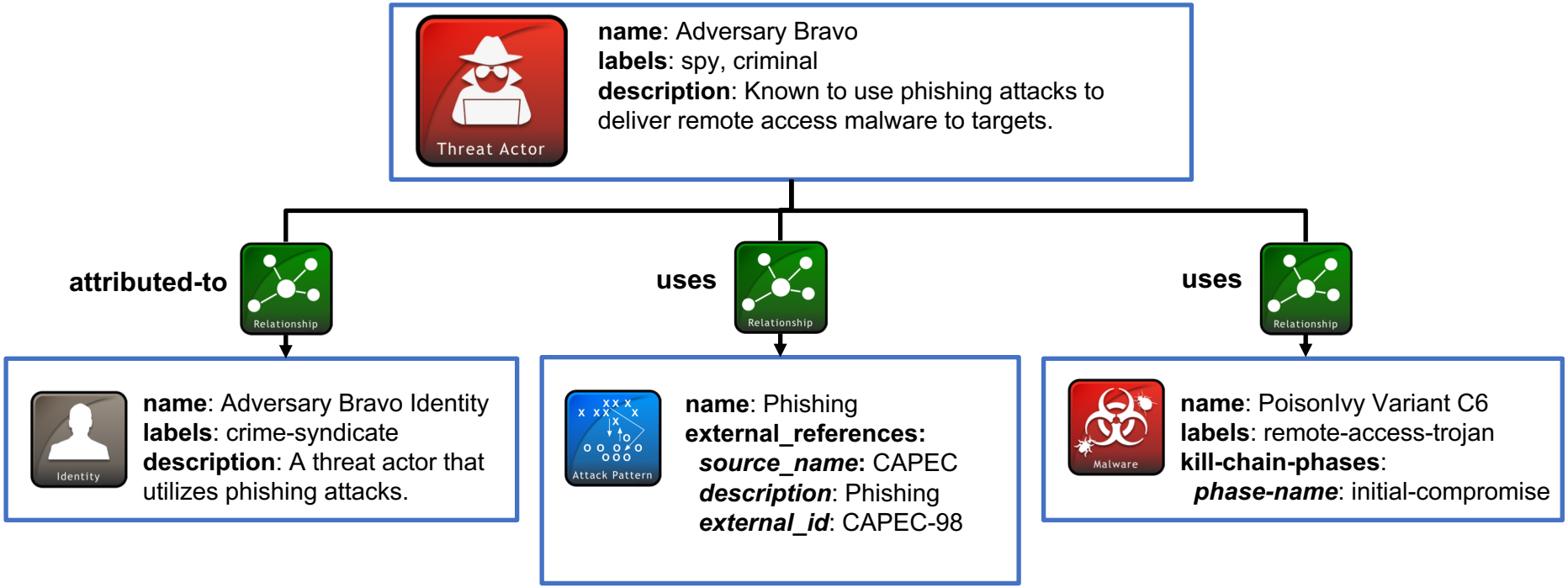
Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Example: Indicator with Relationships



Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Example: Threat Actor with Relationships



Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Example: Campaign with Relationships

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Module 2: Checking In

- Q: Name the commonly used SDO that is used for future detection of malicious activity?

- Q: How many objects (including relationships) would you need to characterize the usage of two distinct malware families by a Threat Actor?

- Q: Which SDO can encompass one or more Campaigns?

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

STIX 2

Module 3: Working with STIX 2.0 Objects

OASIS

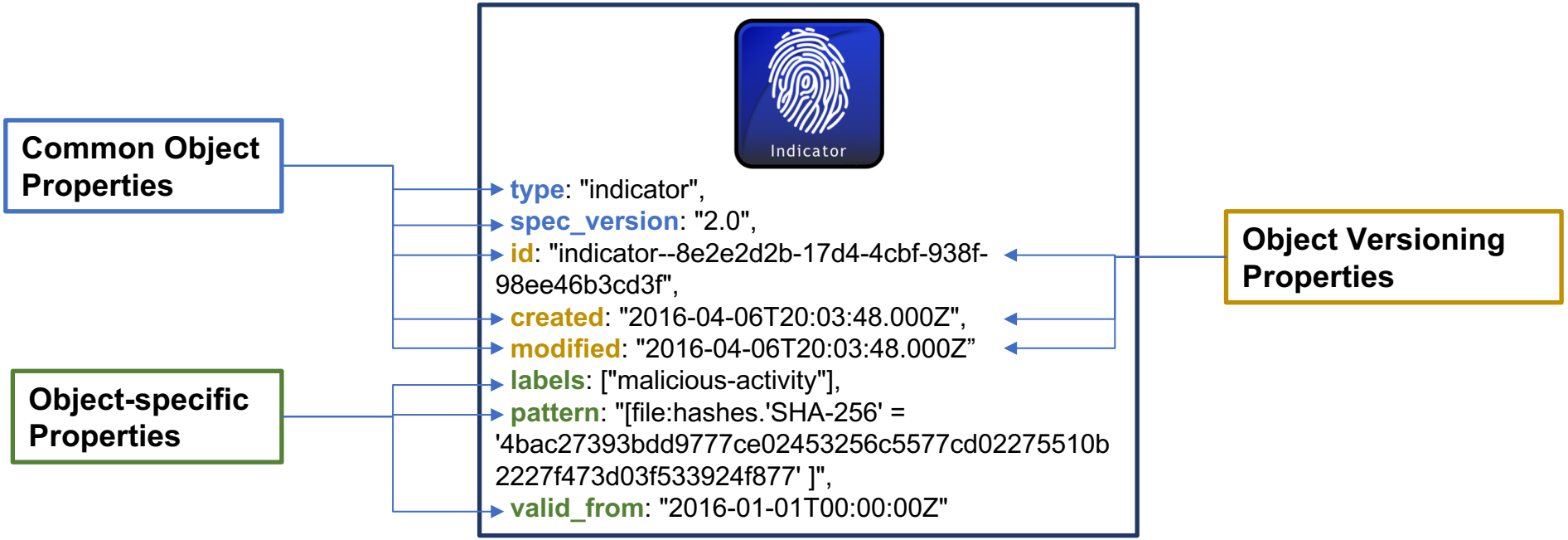
Module 3: Training Objectives

- In this module, you will:
  - Gain an understanding of how to construct STIX 2.0 objects
  - Learn the basics of STIX 2.0 object versioning and what this means with regards to relationships
  - Learn about related topics around STIX 2.0 objects, including data markings, customization, and sightings

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

STIX Object Anatomy



Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Common Object Properties

- All STIX 2.0 Objects have a common set of requirements to support use cases such as identification, versioning, etc.
- Therefore, the STIX 2.0 specification defines a set of *properties* common to all Objects. Required properties are in **bold**:

  - **type**
  - **id**
  - **spec_version**
  - created_by_ref
  - **created**
  - **modified**
  - revoked
  - labels
  - external_references
  - object_marking_refs
  - granular_markings

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Object IDs & References

- Object identifiers, captured in the **id** property, universally and uniquely identify a SDO, SRO, Bundle, or Marking Definition
  - Syntax: *object-type--UUIDv4*

- Object properties that end in **_ref** represent references to other STIX Objects (also valid for Cyber Observables)
  - References point to other Objects via their ID
  - Implementation of ID reference resolution is not defined in the STIX Specification; therefore, references may point to Objects that the consumer/producer does not currently have

- For example:

```
type: "indicator",
spec_version: "2.0",
id: "indicator--8e2e2d2b-17d4",
created_by_ref: "identity--3abeab36-4ac0"
```

```
type: "identity",
spec_version: "2.0",
id: "identity--3abeab36-4ac0"
```

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Creating a Basic STIX 2.0 Object

- There are two major steps to creating a valid STIX 2.0 Object:
  1. Populate all required common properties of the Object
  2. Populate all required Object-specific properties

- For example:

type: "indicator",
spec_version: "2.0",
id: "indicator--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
created: "2016-04-06T20:03:48.000Z",
modified: "2016-04-06T20:03:48.000Z"

Required Common Properties

+

labels: ["malicious-activity"],
pattern: "[file:hashes.'SHA-256' = '4bac27393bdd9777ce02453256c5577cd02275510b2227f473d03f533924f877']",
valid_from: "2016-01-01T00:00:00Z"

Required Object Properties

=

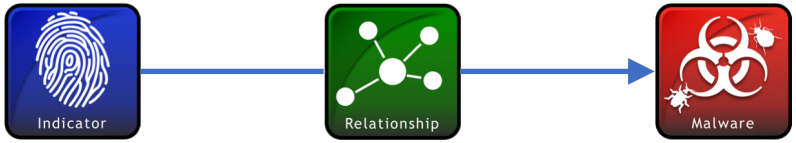
type: "indicator",
spec_version: "2.0",
id: "indicator--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
created: "2016-04-06T20:03:48.000Z",
modified: "2016-04-06T20:03:48.000Z"
labels: ["malicious-activity"],
pattern: "[file:hashes.'SHA-256' = '4bac27393bdd9777ce02453256c5577cd02275510b2227f473d03f533924f877']",
valid_from: "2016-01-01T00:00:00Z"

Valid STIX 2.0 Object

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Object Relationships

- Each STIX Domain Object has its own set of relationship types that are defined specifically for that object. For example, the “indicates” relationship is defined for the Indicator SDO:



indicates

- In addition, STIX defines several relationships that are *common* to all Objects:
  - **derived-from**: the information in the target object is based on information from the source object.
  - **duplicate-of**: the referenced source and target objects are semantically duplicates of each other.
  - **related-to**: the referenced source and target objects are arbitrarily related to each other.

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Object Versioning: Concepts

- Objects may have new versions defined **only** by their *Object Creator*
  - Object Creator: the entity (e.g., system, organization, instance of a tool) that generates the **id** property for a given object. Object creators are represented as Identity objects.
  - Producers other than the Object Creator are not allowed to create new versions of an Object, and instead must create a new Object (with a new **id**)

- Reasons for versioning an existing Object:
  - Updating existing information
  - Adding new information
  - Removing information

- Objects may also be revoked by setting the **revoked** property to true
  - This is permanent – once an Object has been marked as revoked, future versions MUST NOT be created

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Object Versioning: In Practice

- The version of an Object is based on the tuple of its **id** and **modified** properties
  - **id**: immutable – must not be changed after creation
  - **modified**: mutable – represents the time that this *particular version* of an Object was last modified
- For the first version of an Object, the **created** and **modified** timestamps **MUST** have the same value

type: "indicator",
spec_version: "2.0",
id: "indicator--8e2e2d2b-17d4",
created: "2016-04-06T20:03:48.000Z",
modified: "2016-04-06T20:03:48.000Z"

Original Object

→

type: "indicator",
spec_version: "2.0",
id: "indicator--8e2e2d2b-17d4",
created: "2016-04-06T20:03:48.000Z",
modified: "*2016-05-09T12:43:34.000Z*"

New Version

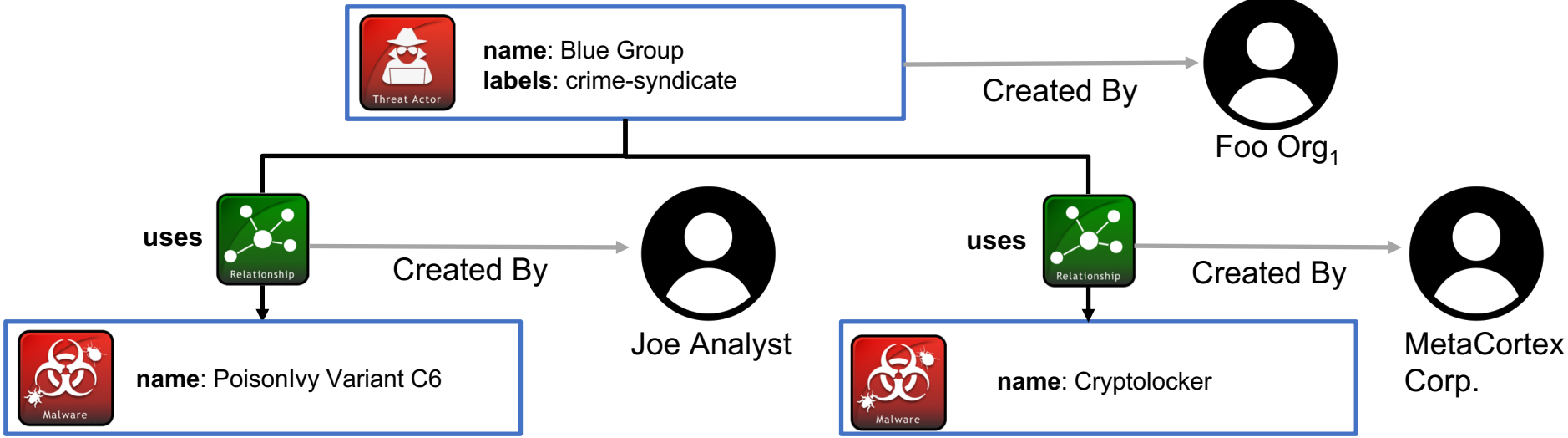
- It's up to the Object Producer to determine when to update an existing object versus creating a new object
  - However, we recommend that anytime an update constitutes a *material change* (a change in the fundamental meaning of an object), a new object with a different **id** should be used

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Versioning & Relationships

- One of the key benefits of STIX 2.0 versioning and relationships is that it allows multiple users to reference STIX Objects *without* necessitating a new version of the object



1- Image credit: https://openclipart.org/detail/247319/abstract-user-icon-3

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Data Markings: Types

- Object markings vs. Granular markings
  - Object markings apply to the STIX Object as a whole
  - Granular markings can be applied to individual properties of STIX objects
    - E.g., The **name** property can be marked differently than the Description property

- Statement markings vs. Traffic Light Protocol (TLP)
  - Statement markings are text based markings
    - E.g., Copyright information
  - TLP markings are predefined as objects in the STIX spec
    - E.g., TLP: RED, TLP: AMBER, TLP: GREEN, TLP: WHITE

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Data Markings: In Practice

- Data markings are separate objects that are then referenced by STIX SDOs using embedded relationships
- Data markings cannot be versioned
  - It would allow for indirect changes to the markings of a STIX object
- STIX Relationship Objects (SROs) to or from a Data Marking object are not allowed
  - Every STIX object has a common property, **object_marking_refs**, that allows you to embed relationships to Data Marking objects

type: "marking-definition",
spec_version: "2.0",
id: "marking-definition--34098fce-860f",
created: "2016-08-01T00:00:00.000Z",
modified: "2016-08-01T00:00:00.000Z",
definition_type: "statement",
definition:
statement: "Copyright 2016, Example Corp"

Example Data Marking

type: "indicator",
spec_version: "2.0",
id: "indicator--8e2e2d2b-17d4",
created: "2016-04-06T20:03:48.000Z",
modified: "2016-04-06T20:03:48.000Z"
object_marking_refs: "marking-definition--34098fce-860f "

Example Data Marking Usage

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Customization and Extension

STIX includes the flexibility to expand the language to meet additional use cases that aren't already in the specification

- Custom properties (STIX and Cyber Observables)
  - Allows you to add additional properties onto existing objects
- Custom objects (STIX and Cyber Observables)
  - Allows you to create entirely new objects that aren't currently defined
- Custom Object Extensions (Cyber Observables only)
  - Allows you to create a *set* of properties that are related to an observable object that already exists


```
type: "x-foo-com-custom-object",
spec_version: "2.0",
id: "x-foo-com-custom-object--34098fce-860f",
created: "2016-08-01T00:00:00.000Z",
modified: "2016-08-01T00:00:00.000Z",
custom_property_1: "foo",
custom_property_2: "bar"
```


Example Custom Object

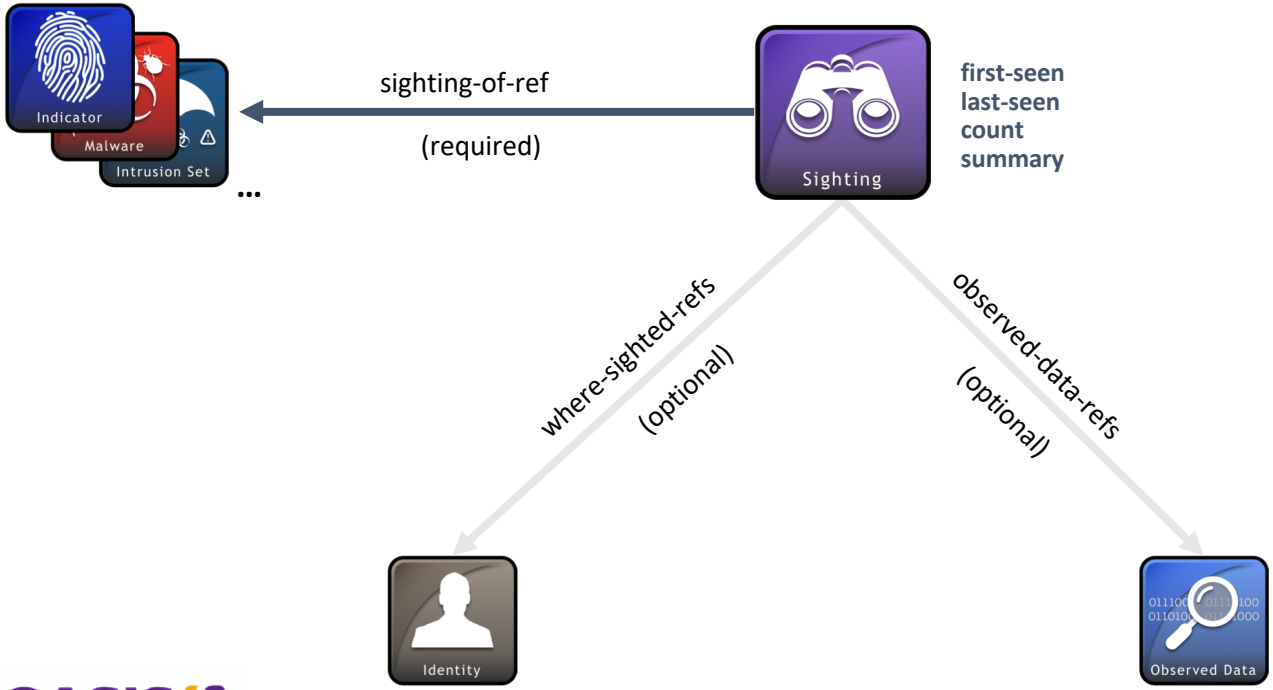

```
type: "indicator",
spec_version: "2.0",
id: "indicator--129a0d3b-471c",
created: "2016-08-01T00:00:00.000Z",
modified: "2016-08-01T00:00:00.000Z",
x_acme_org_threatiness: 2.34
```


Example Custom Property

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Sightings

- A Sighting is a separate STIX Relationship Object that can be produced and shared independently
- Sightings are most likely of an indicator, but any threat object is fine (even Observed Data). Malware and Intrusion set are just given as examples here.



Examples

- That indicator was seen
- That indicator was seen by an organization in the energy sector in San Antonio, Texas
- That indicator was seen by IBM
- That indicator was seen, and here's a network traffic dump
- That intrusion set was seen 3 times by organizations in the defense sector in the United States
- This is a summary sighting of an indicator that was seen 426 times between 1/1/18 and 1/31/18 dates in the finance sector.

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Module 3: Checking In

- Q: Name two properties that are required on all STIX Objects?
- Q: Can you create a new version of an existing STIX Object if it was originally created by someone other than yourself?
- Q: Which relationship type should be used if you wish to express a generic connection between two STIX Objects?

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

STIX 2

Module 4: STIX Patterning Core Concepts

OASIS

Module 4: Training Objectives

- In this module, you will:
  - Gain an understanding of the purpose, history, and design principles of STIX Patterning
  - Learn about the capabilities that the STIX Patterning language can bring in terms of detecting cyber threats
  - See some practical examples of STIX Patterning in use

STIX 2.0 Patterning: Overview & History

- STIX Patterning: a standard language to describe what to look for in a cyber environment
  - In STIX 1.x, this was defined as a native part of the Observable object and used in Indicators
  - In STIX 2.0, this is a separate language with its own grammar and is used in the Indicator Domain Object
- Unlike most other patterning languages, STIX Patterning permits patterns to be written against **both** host-based and network data
  - For example, try expressing the following in YARA or Snort: malware matching signature **X** will beacon with traffic that looks like **Y** before dropping **Z**

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

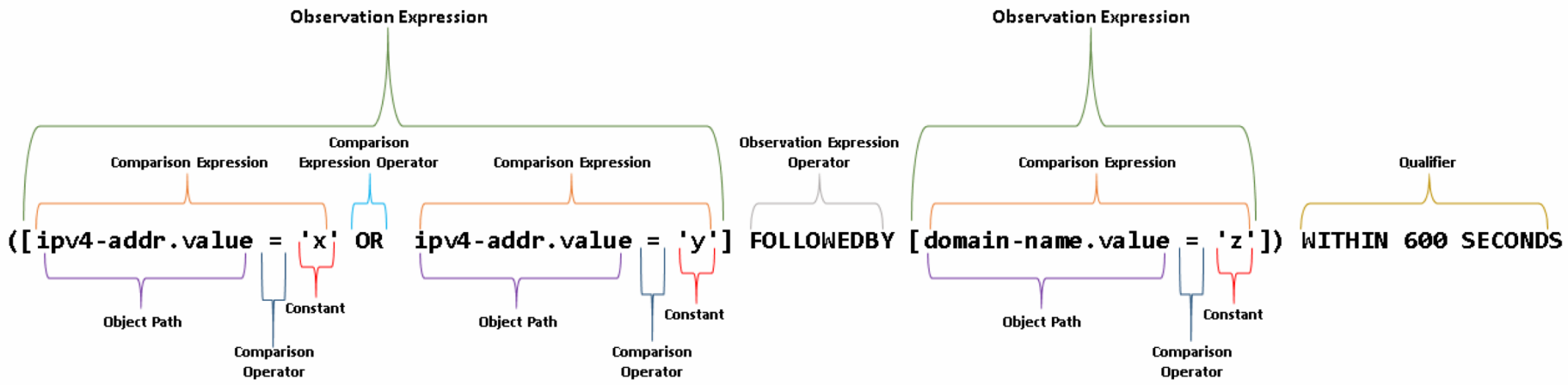
STIX 2.0 Patterning: Basic Design Principles

- One way to do things (not 12)!
- Base things on a **grammar**, not nested XML or JSON
  - Makes things easier for humans to understand, **and** for machines to parse!
- Base that grammar on something that as many folks are familiar with as possible
  - Candidates: SQL, Lucene, YARA, Snort/OpenSig…
  - We ended up with SQL-like after some debate
- Define a grammar that allows sharing descriptions of advanced threats, not just simple atomic IOCs (ip = 1.2.3.4)
- Define it in a way that was expandable in the future without breaking changes

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Basic Structure of a STIX 2.0 Pattern



Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

How this ties to STIX 2.0 Cyber Observables

- STIX Cyber Observables provide a data model for describing things you've *actually* seen
- STIX Patterning is a language for describing chaotic maliciousness one *might* see
- SCO (STIX Cyber Observables) : nouns :: STIX Patterning : language
- SCO : DB Tables :: STIX Patterning : SQL

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Basic Examples

Finding an IP

```
[ip-addr.value = '8.8.8.8']
```

Finding a URL

```
[url:value MATCHES
'^(?:https?:\/\/)?(?:www\.)?example\.com\/.*']
```

Finding one of two registry keys

```
[windows-registry-key:key =
'HKEY_CURRENT_USER\\Software\\CryptoLocker\\Files'
OR windows-registry-key:key =
'HKEY_CURRENT_USER\\Software\\Microsoft\\CurrentVer
sion\\Run\\CryptoLocker_0388']
```

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

More Complex Example: Necurs Botnet

Looks for a particular malware payload followed by HTTP beaconing traffic generated by the payload:

```
[file:name = 'rekakva32.exe' AND file:parent_directory_ref.path MATCHES
'C:\\Users\\[\\w\\s]+\\AppData\\Local\\Temp'] FOLLOWEDBY [network-
traffic:protocols[*] = 'http' AND network-traffic:extensions.'http-
request-ext'.request_method = 'post' AND network-
traffic:extensions.'http-request-ext'.request_header.'User-Agent' =
'Windows-Update-Agent']
```

Source: https://isc.sans.edu/forums/diary/Necurs+Botnet+malspam+pushes+Locky+using+DDE+attack/22946/

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved

Module 4: Checking In

- Q: What can STIX Patterning do that most other patterning languages cannot?
- Q: In which STIX 2.0 Domain Object is STIX Patterning currently used?
- Q: Does STIX Patterning permit the construction of patterns that make use of temporal logic?

OASIS

Copyright © OASIS Open 2018. Portions copyright LookingGlass Inc. and Symantec Inc. All Rights Reserved