STIX 2.0 Specification

Objects and Vocabularies

Version 2.0-draft-1

Document Table of Contents

1. STIX Domain Objects

Each STIX Domain Object (SDO) represents a node within the STIX Object Model (Relationship Objects describe the edges between the nodes). Each SDO allows a producer to describe some part of Cyber Threat Intelligence (CTI) in a structured way. SDOs describe separate but related CTI data, and they are designed to allow producers flexibility in how SDOs are associated. This modularity allows a large number of scenarios to be described, which maximizes the usefulness of STIX in the CTI space.

For each SDO defined below, we provide property information, relationship information, and examples. Property information includes common properties as well as properties that are specific to each SDO. Relationship information includes embedded relationships (e.g., created_by_ref), common relationships (e.g., related_to), and SDO-specific relationships. While it would be sufficient to list only forward relationships (i.e., relationships *from* the SDO of focus *to* other SDOs), for convenience, we have also included reverse relationships.

1.1. Attack Pattern

Type Name: `attack-pattern`

An Attack Pattern is a mechanism for describing and documenting how an attack against one or more targets may be executed. Each Attack Pattern defines how an adversary may attempt to compromise the target, provides a description of the common technique(s) used, and presents recommended methods for mitigating the threat described by the Attack Pattern. The Attack Pattern object helps categorize attacks in a meaningful way to provide coherent and detailed information about how attacks are performed.

In a structured sense, the Attack Pattern object captures information about the techniques attackers use to carry out attacks. It can describe either general attack patterns (e.g., phishing) or specific attack patterns (e.g., phishing as used by XYZ Campaign) and is often used to reference a CAPEC pattern ID.

1.1.1. Properties

Common Properties
type, id, created_by_ref, labels, version, created, modified, revoked, version_comment, external_references, confidence, object_markings_refs, granular_markings

Attack Pattern Specific Properties		
name, description, kill_chain_phases		
Property Name	**Type**	**Description**
type (required)	`string`	The value of this field MUST be `attack-pattern`
external_references (optional)	`list` of type `external-reference`	A list of external references which refer to non-STIX information. This field **MAY** be used to provide one or more Attack Pattern identifiers, such as a CAPEC ID <TODO: add reference>. When specifying a CAPEC ID, the **source** field of the external reference **MUST** be set to `capec` and the **external_id** field **MUST** be formatted as `CAPEC-[id]`.
name (required)	`string`	The name used to identify the Attack Pattern.
description (optional)	`string`	A description that provides more details and context about the Attack Pattern, potentially including its purpose and its key characteristics.
kill_chain_phases (optional)	`list` of type `kill-chain-phase`	The list of Kill Chain phases for which this Attack Pattern is used.

1.1.2. Relationships

These are the relationships explicitly defined between the Attack Pattern object and other objects. The first section lists the embedded relationships by property name along with their corresponding target. The rest of the table identifies the relationships that can be made from this object by way of the Relationship Object. The reverse relationships (relationships "to" this object) are included as a convenience. For their definitions, please see the objects for which they represent a "from" relationship.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	source
object_markings_refs	marking-definition

Common Relationships			
duplicate-of, related-to			
Source	**Name**	**Target**	**Description**
attack-pattern	exploits	vulnerability	This relationship is used to document the Vulnerabilities this Attack Pattern targets.
attack-pattern	targets	victim-target	This relationship is used to document the Victim Targets this Attack Pattern targets.
attack-pattern	uses	malware, tool	This relationship is used to document the Malware and/or Tools that are used to perform the behavior identified in the Attack Pattern.
Reverse Relationships			
incident	attributed-to	attack-pattern	See forward relationship for definition.
indicator	detects	attack-pattern	See forward relationship for definition.
course-of-action	mitigates	attack-pattern	See forward relationship for definition.
campaign, intrusion-set, threat-actor	uses	attack-pattern	See forward relationship for definition.

1.1.3. Examples

```
{
  "type": "attack-pattern",
  "id": "attack-pattern--0c7b5b88-8ff7-4a4d-aa9d-feb398cd0061",
  "version": 1,
  "created": "2016-05-12T08:17:27.000000Z",
  "modified": "2016-05-12T08:17:27.000000Z",
```

```
  "name": "Spear Phishing",
  "description": "...",
  "external_references": [
    {
      "source": "capec",
      "id": "CAPEC-49"
    }
  ]
}
```

1.2. Campaign

Type Name: `campaign`

A Campaign is a set of related instances of a threat actor or group of threat actors pursuing a specific objective or result. These operations are often observed through related SDOs, and can exist potentially across organizational or sector boundaries.

In a structured sense, the Campaign object is used to describe and document a pattern of malicious activity and the motivations of an adversary that has a particular objective over a period of time. It may also reference a set of other objects that are believed to be part of this campaign during a discrete time frame.

For example, a Campaign would be used to describe a crime syndicate's attack against the customers of ACME Bank in Brazil during the summer of 2015.

1.2.1. Properties

Common Properties		
type, id, created_by_ref, labels, version, created, modified, revoked, version_comment, external_references, confidence, object_markings_refs, granular_markings		
Campaign Specific Properties		
name, description, aliases, first_seen, first_seen_precision, objectives, resource_level, primary_motivation, secondary_motivations, origin		
Property Name	**Type**	**Description**

type (required)	`string`	The value of this field **MUST** be `campaign`
name (required)	`string`	The name used to identify the Campaign.
description (optional)	`string`	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
aliases (optional)	`list` of type `string`	Alternative names used to identify this Campaign
first_seen (optional)	`timestamp`	The time that his Campaign was first seen.
first_seen_precision (optional)	`timestamp-precision`	The precision of the **first_seen** timestamp.
objectives (optional)	`list` of type `open-vocab`	This field defines the Campaign's primary goal, objectives, desired outcomes, or intended effect — what the Campaign hopes to accomplish with this Campaign. The Campaign may use many methods to achieve this goal, and the primary goal may have secondary or ancillary effects. This is an open vocabulary and values **SHOULD** come from the `attack-objectives-ov` vocabulary.
resource_level (optional)	`open-vocab`	This defines the organizational level at which this Campaign typically works, which in turn determines the resources available to this Campaign for use in an attack. This is an open vocabulary and values **SHOULD** come from the `attack-resource-level-ov` vocabulary.

primary_motivation (optional)	`open-vocab`	The primary reason, motivation, or purpose behind this Campaign. The primary motivation is the archetypical,single most prevalent and descriptive motivation of this Campaign. This motivation is intrinsic to the Campaign and the primary cause of the Campaign's actions. This is an open vocabulary and values **SHOULD** come from the `attack-motivation-ov` vocabulary.
secondary_motivations (optional)	`list` of type `open-vocab`	The secondary reasons, motivations, or purposes behind this Campaign. Secondary motivations can exist as an equal or near-equal cause to the Primary Motivation. It does not replace or magnify the Primary Motivation, but might indicate additional asset or attack targeting. This is an open vocabulary and values **SHOULD** come from the `attack-motivation-ov` vocabulary.
origin (optional)	`string`	The country of origin for this Campaign. When representing nationalities, the value **MUST** be from [ISO Ref].

1.2.2. Relationships

These are the relationships explicitly defined between the Campaign object and other objects. The first section lists the embedded relationships by property name along with their corresponding target. The rest of the table identifies the relationships that can be made from this object by way of the Relationship Object. The reverse relationships (relationships "to" this object) are included as a convenience. For their definitions, please see the objects for which they represent a "from" relationship.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	source
object_markings_refs	marking-definition

Common Relationships			
duplicate-of, related-to			
Source	**Name**	**Target**	**Description**
campaign	attributed-to	intrusion-set, threat-actor	This relationship is used to document which Intrusion Sets and Threat Actors are involved with this Campaign.
campaign	targets	victim-target, vulnerability	This relationship is used to document the type of Victim Targets or Vulnerabilities this Campaign targets.
campaign	uses	attack-pattern, malware, tool	This relationship is used to document the Attack Patterns, Malware, or Tools that a Campaign uses or that are used during the Campaign.
Reverse Relationships			
indicator	indicates	campaign	See forward relationship for definition.
incident	attributed-to	campaign	See forward relationship for definition.

1.2.3. Examples

```
{
  "type": "campaign",
  "id": "campaign--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "version": 1,
  "created": "2016-04-06T20:03:48Z",
  "modified": "2016-04-06T20:03:48Z",
  "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "name": "Green Group Attacks Against Finance",
```

```
  "description": "Campaign by Green Group against a series of targets in the financial
services sector."
}
```

1.4. Course of Action

Type Name: `course-of-action`

A Course of Action is used to describe a course of action to taken either to prevent an attack or to respond to an attack that is already occurring. Courses of Action may describe technical, automatable responses (applying patches, reconfiguring firewalls) but can also describe higher level actions like employee training or policy changes.

The `mitigates` relationship is used to link the Course of Action to the activity or vulnerability that it can mitigate. The relationships to Incident can link it to incidents where the Course of Action is suggested or has actually been taken.

1.4.1. Properties

Common Properties		
`type`, `id`, `created_by_ref`, `labels`, `version`, `created`, `modified`, `revoked`, `version_comment`, `external_references`, `confidence`, `object_markings_refs`, `granular_markings`		
Course of Action Specific Properties		
`name`, `description`, `action`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this field **MUST** be `course-of-action`
`labels` (required)	`list` of type `open-vocab`	This field is an Open Vocabulary that specifies the type of indicator. This is an open vocabulary and values **SHOULD** come from the `course-of-action-label-ov` vocabulary.

name (required)	`string`	The name used to identify the Course of Action
description (optional)	`string`	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
action (reserved)	`RESERVED`	RESERVED - To capture structured/automated courses of action.

1.4.2. Relationships

These are the relationships explicitly defined between the Course of Action object and other objects. The first section lists the embedded relationships by property name along with their corresponding target. The rest of the table identifies the relationships that can be made from this object by way of the Relationship Object. The reverse relationships (relationships "to" this object) are included as a convenience. For their definitions, please see the objects for which they represent a "from" relationship.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships			
created_by_ref		`source`	
object_markings_refs		`marking-definition`	
Common Relationships			
`duplicate-of`, `related-to`			
Source	**Name**	**Target**	**Description**
`course-of-action`	`mitigates`	`attack-pattern`, `malware`, `vulnerability`, `tool`, `incident`	This relationship links the Course of Action to the objects it can mitigate.
Reverse Relationships			

incident	uses	course-of-action	See forward relationship for definition.

1.4.3. Examples

```
[
  {
    "type": "course-of-action",
    "id": "course-of-action--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
    "version": 1,
    "created": "2016-04-06T20:03:48Z",
    "modified": "2016-04-06T20:03:48Z",
    "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
    "name": "Add TCP port 80 Filter Rule to the existing Block UDP 1434 Filter",
    "description": "This is how to add a filter rule to block inbound access to TCP port 80
the existing UDP 1434 filter ..."
  },
  {
    "type": "relationship",
    "id": "relationship--44298a74-ba52-4f0c-87a3-1824e67d7fad",
    "version": 1,
    "created": "2016-04-06T20:06:37Z",
    "modified": "2016-04-06T20:06:37Z",
    "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
    "source_ref": "course-of-action--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
    "target_ref": "malware--31b940d4-6f7f-459a-80ea-9c1f17b5891b",
    "name": "mitigates"
  },
  {
    "type": "malware",
    "id": "malware--31b940d4-6f7f-459a-80ea-9c1f17b5891b",
    "version": 1,
    "created": "2016-04-06T20:07:09Z",
    "modified": "2016-04-06T20:07:09Z",
    "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
    "name": "Poison Ivy"
  }
]
```

1.5. Incident

Type Name: `incident`

Incidents are discrete instances of threats affecting an organization along with information discovered or decided during an incident response investigation. They consist of data such as

time-related information, parties involved, impact assessment, related indicators, related observables, leveraged attack techniques, attribution, responses taken or recommended.

1.5.1. Properties

Common Properties		
type, id, created_by_ref, labels, version, created, modified, revoked, version_comment, external_references, confidence, object_markings_refs, granular_markings		
Indicator Specific Properties		
name, description, initial_compromise, initial_compromise_precision, discovery, discovery_precision, remediation, remediation_precision		
Property Name	**Type**	**Description**
type (required)	`string`	The value of this field **MUST** be `incident`
labels (required)	`list` of type `open-vocab`	This field is an Open Vocabulary that specifies the type of indicator. This is an open vocabulary and values **SHOULD** come from the `incident-label-ov` vocabulary.
name (required)	`string`	The name used to identify the Incident.
description (optional)	`string`	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
initial_compromise (optional)	`timestamp`	Specifies the time that the initial compromise occurred for the Incident
initial_compromise_precision (optional)	`timestamp-precision`	The timestamp precision of **initial_compromise.**.

discovery (optional)	timestamp	Specifies the first time at which the organization learned the Incident had occurred.
discovery_precision (optional)	timestamp-precision	The timestamp precision of **discovery**.
remediation (optional)	timestamp	Specifies the first time at which the Incident is remediated.
remediation_precision (optional)	timestamp-precision	The timestamp precision of **remediation**.

1.5.2. Relationships

These are the relationships explicitly defined between the Incident object and other objects. The first section lists the embedded relationships by property name along with their corresponding target. The rest of the table identifies the relationships that can be made from this object by way of the Relationship Object. The reverse relationships (relationships "to" this object) are included as a convenience. For their definitions, please see the objects for which they represent a "from" relationship.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the related-to relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	source
object_markings_refs	marking-definition

Common Relationships			
duplicate-of, related-to			
Source	**Name**	**Target**	**Description**
incident	attributed-to	attack-pattern, campaign, intrusion-set, malware, threat-actor	This relationship is used to document the objects that are suspected of being responsible for this Incident.

`incident`	`exploits`	`victim-target`	This relationship is used to describe the victims that were actually exploited during the attack that created this Incident.
`incident`	`targets`	`victim-target`	This relationship is used to describe the intended or actual victims targeted during the attack that created this Incident.
`incident`	`uses`	`course-of-action`	This relationship is used to describe the course of action taken during the Incident.
Reverse Relationships			
`course-of-action`	`mitigates`	`incident`	See forward relationship for definition.

1.5.3. 1.4.3.Examples

```
{
  "type": "incident",
  "id": "incident--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "version": 1,
  "created": "2016-04-06T20:03:48Z",
  "modified": "2016-04-06T20:03:48Z",
  "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "name": "Green Group Infiltration of Web Servers",
  "description": "Green group was able to infiltrate the web server infrastructure and caused sporadic and unpredictable content defacement issues."
}
```

1.6. Indicator

Type Name: `indicator`

An Indicator is used to detect activities of interest and for conveying specific patterns combined with their contextual information. Indicators represent patterns of artifacts and/or behaviors of interest within a cyber domain.

The Indicator object is used to document a pattern of activity, using a structured patterning language that represents something that you might see or something to look for. The CybOX patterning <todo add reference to CybOX spec> language is preferred wherever possible. While implementers are free to support whatever additional structured patterning language(s) they

choose (e.g., Snort or Yara), the CybOX patterning language is the MTI format for STIX Indicators and **MUST** be supported. The Indicator pattern is intended to be used by consumers to match against data collected from their environment, such as network and host logs.

For example, using relationships with an Indicator you can express concepts such as, if you see this Indicator then there is evidence that you have this malware, tool, or are under attack by this Campaign or Threat Actor.

1.6.1. Properties

Common Properties		
`type`, `id`, `created_by_ref`, `labels`, `version`, `created`, `modified`, `revoked`, `version_comment`, `external_references`, `confidence`, `object_markings_refs`, `granular_markings`		
Indicator Specific Properties		
`name`, `description`, `pattern`, `pattern_lang`, `valid_from`, `valid_from_precision`, `valid_to`, `valid_to_precision`, `kill_chain_phases`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this field **MUST** be `indicator`
`labels` (required)	`list` of type `open-vocab`	This field is an Open Vocabulary that specifies the type of indicator. This is an open vocabulary and values **SHOULD** come from the `indicator-label-ov` vocabulary.
`name` (optional)	`string`	The name used to identify the Indicator.
`description` (optional)	`string`	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
`pattern` (required)	`string`	The detection pattern for this indicator. The default language is

		CybOX Patterning; implementations **MUST** support processing of CybOX patterns and **MAY** support others.
pattern_lang (optional)	`open-vocab`	The language used to define the pattern (in the **pattern** field). The default is `cybox` if the field is omitted. This is an open vocabulary and values **SHOULD** come from the `pattern-lang-ov` vocabulary.
valid_from (required)	`timestamp`	The time from which this indicator should be considered valuable intelligence.
valid_from_precision (optional)	`timestamp-precision`	The precision of the start timestamp.
valid_to (optional)	`timestamp`	The time at which this indicator should no longer be considered valuable intelligence.
valid_to_precision (optional)	`timestamp-precision`	The precision of the end timestamp.
kill_chain_phases (optional)	`list` of type `kill-chain-phase`	The phases of the kill chain that this indicator detects. <todo: Fix this definition.>

1.6.2. Relationships

These are the relationships explicitly defined between the Indicator object and other objects. The first section lists the embedded relationships by property name along with their corresponding target. The rest of the table identifies the relationships that can be made from this object by way of the Relationship Object. The reverse relationships (relationships "to" this object) are included as a convenience. For their definitions, please see the objects for which they represent a "from" relationship.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	source
object_markings_refs	marking-definition

Common Relationships			
duplicate-of, related-to			
Source	**Name**	**Target**	**Description**
indicator	detects	attack-pattern, malware, tool	This relationship is used to document that an Attack Pattern, Malware, or Tool may exist if this Indicator is triggered. For example, you can send a relationships that points from an Indicator to some Malware with a value of "detects". What that means is if you see that Indicator it indicates that you have that piece of Malware running on your computer / in your network to the level of confidence expressed in the relationship.
indicator	indicates	campaign, intrusion-set, threat-actor	This relationship is used to document that you may be seeing this Campaign, Intrusion Set, or Threat Actor of this Indicator is triggered. For example, if you have an IP address or domain name that is believed to be part of a Campaign, Intrusion Set, or Threat Actor, you can link them to an Indicator and say that this IP address or domain name, for example, indicates the presence of this Campaign, Intrusion Set, or Threat Actor.
Reverse Relationships			

1.6.3. Examples

Indicator Itself, with Context

```
[
  {
    "type": "indicator",
    "id": "indicator--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
    "created": "2016-04-06T20:03:48Z",
    "modified": "2016-04-06T20:03:48Z",
    "version": 1,
    "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
    "name": "Poison Ivy Malware",
    "description": "This file is part of Poison Ivy",
    "pattern": "file-object.hashes.md5 = '3773a88f65a5e780c8dff9cdc3a056f3'",
    "pattern_lang": "cybox",
    "valid_from": "2016-01-01T00:00:00Z"
  },
  {
    "type": "relationship",
    "id": "relationship--44298a74-ba52-4f0c-87a3-1824e67d7fad",
    "created": "2016-04-06T20:06:37Z",
    "modified": "2016-04-06T20:03:48Z",
    "version": 1,
    "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
    "source_ref": "indicator--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
    "target_ref": "malware--31b940d4-6f7f-459a-80ea-9c1f17b5891b",
    "name": "indicates"
  },
  {
    "type": "malware",
    "id": "malware--31b940d4-6f7f-459a-80ea-9c1f17b5891b",
    "created": "2016-04-06T20:07:09Z",
    "modified": "2016-04-06T20:03:48Z",
    "version": 1,
    "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
    "name": "Poison Ivy"
  }
]
```


1.7. Intrusion Set

Type Name: `intrusion-set`

A grouped set of activity with common properties that is believed to be orchestrated by a single organization. An Intrusion Set may capture multiple Campaigns, Incidents or activity that were all tied together by a shared attributes indicating a common known or unknown threat actor.

An Intrusion Set relates a set of Campaigns, Incidents, Indicators, Observed Data, or Tools, that are grouped together to show a believed attribution back to an entity. For example, a set of

Incidents may share a common IP range. The Threat Actors behind the attack may not be known but the activity can be grouped together and new activity can be attributed to that Intrusion Set. Threat Actors could move from supporting one Intrusion Set, to supporting another, or they may support multiple Intrusion Sets. An Intrusion Set is usually tracked over a long period of time. While sometimes an Intrusion Set goes silent, or changes focus, it is usually difficult to know if it has truly disappeared or ended. Analysts may have varying level of fidelity on attributing an Intrusion Set back to Threat Actors. The analysts may be able to only attribute it back to a nation-state, perhaps back to an organization within that nation-state, or perhaps back to the individuals within that organization.

Different sharing groups or organizations may have different naming conventions for Intrusion Sets. For this reason, aliases or an equality relationship is required between Intrusion Sets.

1.7.1. Properties

Common Properties		
`type`, `id`, `created_by_ref`, `labels`, `version`, `created`, `modified`, `revoked`, `version_comment`, `external_references`, `confidence`, `object_markings_refs`, `granular_markings`		
Campaign Specific Properties		
`name`, `description`, `aliases`, `first_seen`, `first_seen_precision`, `objectives`, `resource_level`, `primary_motivation`, `secondary_motivations`, `origin`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this field **MUST** be `intrusion-set`
`name` (required)	`string`	The name used to identify this Intrusion Set.
`description` (optional)	`string`	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
`aliases` (optional)	`list` of type `string`	Alternative names used to identify this Intrusion Set.
`first_seen` (optional)	`timestamp`	The time that this Intrusion Set was first seen.

first_seen_precision (optional)	`timestamp-precision`	The precision value for the **first_seen** field
objectives (optional)	`list` of type `open-vocab`	This field defines the Intrusion Set's primary goal, objectives, desired outcomes, or intended effect — what the Intrusion Set hopes to accomplish. The Intrusion Set may use many methods to achieve this goal, and the primary goal may have secondary or ancillary effects. This is an open vocabulary and values **SHOULD** come from the `attack-objectives-ov` vocabulary.
resource_level (optional)	`open-vocab`	This defines the organizational level at which this Intrusion Set typically works, which in turn determines the resources available to this Intrusion Set for use in an attack. This is an open vocabulary and values **SHOULD** come from the `attack-resource-level-ov` vocabulary.
primary_motivation (optional)	`open-vocab`	The primary reason, motivation, or purpose behind this Intrusion Set. The primary motivation is the archetypical,single most prevalent and descriptive motivation of this Intrusion Set. This motivation is intrinsic to the Intrusion Set and the primary cause of the Intrusion Set's actions. This is an open vocabulary and values **SHOULD** come from the `attack-motivation-ov` vocabulary.

secondary_motivations (optional)	`list` of type `open-vocab`	The secondary reasons, motivations, or purposes behind this Intrusion Set. Secondary motivations can exist as an equal or near-equal cause to the Primary Motivation. It does not replace or magnify the Primary Motivation, but might indicate additional asset or attack targeting. This is an open vocabulary and values **SHOULD** come from the `attack-motivation-ov` vocabulary.
origin (optional)	`string`	The country of origin for this Intrusion Set. When representing nationalities, the value **MUST** be from <todo ISO Ref>.

1.7.2. Relationships

These are the relationships explicitly defined between the Intrusion Set object and other objects. The first section lists the embedded relationships by property name along with their corresponding target. The rest of the table identifies the relationships that can be made from this object by way of the Relationship Object. The reverse relationships (relationships "to" this object) are included as a convenience. For their definitions, please see the objects for which they represent a "from" relationship.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	`source`
object_markings_refs	`marking-definition`
Common Relationships	
`duplicate-of`, `related-to`	

Source	Name	Target	Description
`intrusion-set`	`attributed-to`	`threat-actor`	This relationship is used to document which Threat Actors are involved with this Intrusion Set.
`intrusion-set`	`targets`	`victim-target`, `vulnerability`	This relationship is used to document the type of Victim Targets or Vulnerabilities this Intrusion Set targets.
`intrusion-set`	`uses`	`attack-pattern`, `malware`, `tool`	This relationship is used to document the Attack Patterns, Malware, or Tools that an Intrusion Set uses or that are used during the attack.
Reverse Relationships			
`campaign, incident`	`attributed-to`	`intrusion-set`	See forward relationship for definition.
`indicator`	`indicates`	`intrusion-set`	See forward relationship for definition.

1.7.3. Example

```
{
  "type": "intrusion-set",
  "id": "intrusion-set--4e78f46f-a023-4e5f-bc24-71b3ca22ec29",
  "name": "Bobcat Breakin",
  "description": "Incidents usually feature a shared TTP of a bobcat being released within the building containing network access, scaring users to leave their computers without locking them first.  Still determining where the threat actors are getting the bobcats.",
  "aliases": ["Zookeeper"],
  "objectives": ["acquisition-theft", "harassment", "damage"]
}
```

1.8. Malware

Type Name: `malware`

Malware, also known as malicious code and malicious software, refers to a program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim's data, applications, or operating system (OS) or of otherwise annoying or disrupting the victim. Malware such as viruses and worms is usually designed to perform these nefarious functions in such a way that users are unaware of them, at least initially.[1]

The Malware object helps categorize, characterize, and identify malware samples and may be associated with MAEC[2] content. This provides detailed information about how the malware works and what it does.

1.8.1. Properties

Common Properties		
`type`, `id`, `created_by_ref`, `labels`, `version`, `created`, `modified`, `revoked`, `version_comment`, `external_references`, `confidence`, `object_markings_refs`, `granular_markings`		
Malware Specific Properties		
`name`, `description`, `kill_chain_phases`, `maec`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this field **MUST** be `malware`
`labels` (required)	`list` of type `open-vocab`	The type of malware being described. This is an open vocabulary and values **SHOULD** come from the `malware-labels-ov` vocabulary.
`external_references` (optional)	`list` of type `external-reference`	A list of external references which refer to non-STIX information. This field **MAY** be used to capture names for the malware across anti-virus or anti-malware tools. When doing so, the **`source`** property

[1] NIST SP 800-83. http://csrc.nist.gov/publications/nistpubs/800-83/SP800-83.pdf.
[2] Malware Attribute Enumeration and Characterization. http://maecproject.github.io.

		SHOULD be used to capture the vendor or tool name and the **external_id** property **SHOULD** be used to capture the exact name it's known by. For example, to capture that an AV tool called "anti-malware-tool" detects the malware as "very-bad-malware", an external reference could be added with a **source** of `anti-malware-tool` and an **external_id** of `very-bad-malware`.
name (required)	`string`	The name used to identify the Malware.
description (optional)	`string`	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
kill_chain_phases (optional)	`list` of type `kill-chain-phase`	The list of Kill Chain Phases for which this Malware instance can be used.
maec (optional)	maec-container	The MAEC content that describes the Malware.

1.8.2. Relationships

These are the relationships explicitly defined between the Malware object and other objects. The first section lists the embedded relationships by property name along with their corresponding target. The rest of the table identifies the relationships that can be made from this object by way of the Relationship Object. The reverse relationships (relationships "to" this object) are included as a convenience. For their definitions, please see the objects for which they represent a "from" relationship.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	`source`

object_markings_refs	marking-definition
Common Relationships	
duplicate-of, related-to	

Source	Name	Target	Description
malware	exploits	vulnerability	This relationship is used to document the Vulnerabilities this Attack Pattern targets.
malware	targets	victim-target	This relationship is used to document the Victim Target who is being targeted by this Malware
malware	variant-of	malware	This relationship is used to document that one piece of Malware is a variant of another piece of Malware. For example, TorrentLocker is a variant of Cryptolocker.
Reverse Relationships			
incident	attributed-to	malware	See forward relationship for definition.
indicator	detects	malware	See forward relationship for definition.
course-of-action	mitigates	malware	See forward relationship for definition.
attack-pattern, campaign, intrusion-set, threat-actor	uses	malware	See forward relationship for definition.

1.8.3. Examples

```
{
  "type": "malware",
  "id": "malware--0c7b5b88-8ff7-4a4d-aa9d-feb398cd0061",
  "version": "1",
  "created": "2016-05-12T08:17:27.000000Z",
  "modified": "2016-05-12T08:17:27.000000Z",
  "name": "Cryptowall",
  "description": "...",
```

```
  "labels": ["ransomware"]
}
```

1.9. Observed Data

Type Name: `observed-data`

This object documents objects and actions that were observed at a specific time. The `observed-data` object uses CybOX objects to describe the details about what was seen.

For example, `observed-data` can capture the observation of an IP address, of a network connection, of a file, or of a registry key. Future versions of CybOX will also allow the capture of actions, such as a registry key modification.

Observed data is often linked to Sightings and Incidents to capture the technical details of the sighting or malicious behavior.

1.9.1. Properties

Common Properties		
`type, id, created_by_ref, labels, version, created, modified, revoked, version_comment, external_references, confidence, object_markings_refs, granular_markings`		
Observed Data Specific Properties		
`first_observed, last_observed, count, cybox`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this field **MUST** be `observed-data`
`first_observed` (required)	`timestamp`	The starting time of this Observed Data.
`last_observed` (required)	`timestamp`	The ending time this Observed Data. For single point in time data, this should match the **`start`** time. If the count equals 1, then the `start` and `end` timestamps **MUST**

		be equal.
count (required)	`integer`	This is an integer between 0 and 999,999,999 inclusive.
cybox (required)	`cybox-container`	The CybOX content that describes what was seen.

1.9.2. Relationships

These are no relationships explicitly defined between the Observed Data object and other objects, other than those defined as common relationships. The first section lists the embedded relationships by property name along with their corresponding target.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	`source`
object_markings_refs	`marking-definition`
Common Relationships	
`duplicate-of`, `related-to`	

1.9.3. Examples

Observed Data of a file object

```
{
  "type": "observed-data",
  "id": "observed-data--b67d30ff-02ac-498a-92f9-32f845f448cf",
  "created": "2016-04-06T19:58:16Z",
  "modified": "2016-04-06T19:58:16Z",
  "version": 1,
  "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "first_observed": "2015-12-21T19:00:00Z",
  "last_observed": "2015-12-21T19:00:00Z",
  "count": 50,
  "cybox": {
    "objects": {
      "1": {
```

```
        "type": "file-object",
        "file_name": "malware.exe",
        "hashes": {
          "md5": "3773a88f65a5e780c8dff9cdc3a056f3",
          "sha1": "cac35ec206d868b7d7cb0b55f31d9425b075082b"
        }
      }
    }
  }
}
```

1.10. Report

Type Name: `report`

The Report object is used to relay related cyber threat intelligence via a set of related STIX objects which provide context and details. This enables producers to publish a comprehensive cyber threat story using STIX in the same way they might produce a report for a website or blog.

In a structured sense, the Report object is used to describe and document, by reference id, a set of STIX Domain Objects, Relationship Objects, and Marking Definition Objects that are related and have shared contextual meaning.

For example, a threat report by an intel provider discussing the campaigns, techniques, Malware used by a Threat Actor would be represented with this object.

1.10.1. Properties

Common Properties		
`type, id, created_by_ref, labels, version, created, modified, revoked, version_comment, external_references, confidence, object_markings_refs, granular_markings`		
Report Specific Properties		
`name, description, published, published_precision, report_refs`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this field **MUST** be `report`

labels (required)	list of type open-vocab	This field is an Open Vocabulary that specifies the primary subject of this report. This is an open vocabulary and values **SHOULD** come from the report-label-ov vocabulary.
name (required)	string	The name used to identify the Report.
description (optional)	string	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
published (required)	timestamp	The date that this report object was officially published by the creator of this report.
published_precision (optional)	timestamp-precision	The precision of the **published** field.
report_refs (required)	list of type identifier	Specifies other top-level objects that are referred to by this Report.

1.10.2. Relationships

These are no relationships explicitly defined between the Report object and other objects, other than those defined as common relationships. The first section lists the embedded relationships by property name along with their corresponding target.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the related-to relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	source
object_markings_refs	marking-definition
report_refs	list of type identifier
Common Relationships	
duplicate-of, related-to	

1.10.3. Examples

```
// Just a report, where the consumer may or may not already have access to the SDOs
{
  "type": "report",
  "id": "report--84e4d88f-44ea-4bcd-bbf3-b2c1c320bcb3",
  "created": "2015-12-21T19:59:11Z",
  "modified": "2016-05-21T19:59:11Z",
  "version": 1,
  "created_by_ref": "source--a463ffb3-1bd9-4d94-b02d-74e4f1658283",
  "name": "The Black Vine Cyberespionage Group",
  "description": "A simple report with an indicator and campaign",
  "labels": ["campaign-report"],
  "report_contains_refs": [
    "indicator--26ffb872-1dd9-446e-b6f5-d58527e5b5d2",
    "campaign--83422c77-904c-4dc1-aff5-5c38f3a2c55c",
    "relationship--f82356ae-fe6c-437c-9c24-6b64314ae68a"
  ]
}


// A full bundle with a report and the SDOs / Relationships that are part of the report
{
  "type": "bundle",
  "id": "bundle--44af6c39-c09b-49c5-9de2-394224b04982",

  "sources": [
    {
      "type": "source",
      "id": "source--a463ffb3-1bd9-4d94-b02d-74e4f1658283",
      "name": "Acme Cybersecurity Solutions",
    }
  ],

   "reports": [
    {
      "type": "report",
      "id": "report--84e4d88f-44ea-4bcd-bbf3-b2c1c320bcbd",
      "created": "2015-12-21T19:59:11Z",
      "modified": "2016-05-21T19:59:11Z",
      "version": 1,
      "created_by_ref": "source--a463ffb3-1bd9-4d94-b02d-74e4f1658283",
      "name": "The Black Vine Cyberespionage Group",
      "description": "A simple report with an indicator and campaign",
      "labels": ["campaign-report"],
      "report_contains_refs": [
        "indicator--26ffb872-1dd9-446e-b6f5-d58527e5b5d2",
        "campaign--83422c77-904c-4dc1-aff5-5c38f3a2c55c"
```

```
      ]
    }
  ],

  "indicators": [
    {
      "type": "indicator",
      "id": "indicator--26ffb872-1dd9-446e-b6f5-d58527e5b5d2",
      "created": "2015-12-21T19:59:17Z",
      "modified": "2016-05-21T19:59:11Z",
      "version": 1,
      "name": "Some indicator",
      "indicator_types": ["anonymization"],
      "created_by_ref": "source--a463ffb3-1bd9-4d94-b02d-74e4f1658283"
    }
  ],

  "campaigns": [
    {
      "type": "campaign",
      "id": "campaign--83422c77-904c-4dc1-aff5-5c38f3a2c55c",
      "created": "2015-12-21T19:59:17Z",
      "modified": "2016-05-21T19:59:11Z",
      "version": 1,
      "name": "Some Campaign",
      "created_by_ref": "source--a463ffb3-1bd9-4d94-b02d-74e4f1658283"
    }
  ],

  “relationships”: [
    {
      "id": "relationship--f82356ae-fe6c-437c-9c24-6b64314ae68a",
      "type": "relationship",
      "created_at": "2015-12-21T19:59:17.000000+00:00",
      "source_ref": "indicator--26ffb872-1dd9-446e-b6f5-d58527e5b5d2",
      "target_ref": "campaign--26ffb872-1dd9-446e-b6f5-d58527e5b5d2",
      "name": "indicates",
      "created_by_ref": "source--a463ffb3-1bd9-4d94-b02d-74e4f1658283"
    },
  ]
}
```

1.11. Source

Type Name: `source`

This object represents basic identifying information for specific individuals and organizations that provide information in STIX. Source information is useful to understand the reliability, quality, and context of information received in STIX. Sources are linked to STIX Objects via the **`created_by_ref`** field.

1.11.1. Properties

Common Properties		
`type, id, created_by_ref, labels, version, created, modified, revoked, version_comment, external_references, confidence, object_markings_refs, granular_markings`		
Source Specific Properties		
`name, description, classification, sector, contact_information`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this field **MUST** be `source`
`name` (required)	`string`	The name of this Source. When referring to a specific entity (e.g., an individual or organization), this field **SHOULD** contain the canonical name of the specific entity. In cases where the specific entity is unknown or the identity is a class (e.g. industry sector), this field **SHOULD** be omitted.
`description` (optional)	`string`	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
`classification` (required)	`open-vocab`	The type of entity that this Source describes, e.g. an individual, organization. This is an open vocabulary and the values **SHOULD** come from the `identity-classification-ov` vocabulary.

sector (optional)	`open-vocab`	The sector of this Source. This is an open vocabulary and values **SHOULD** come from the `industry-sector-ov` vocabulary.
contact_information (optional)	`string`	The contact information (e-mail, phone number, etc.) for this Source.

1.11.2. Relationships

There is a direct embedded reference to Source in all top-level objects (inherited from the Common Properties), **created_by_ref**, that links each object with the Source of the organization or individual that created the object.

These are no relationships explicitly defined between the Source object and other objects, other than those defined as common relationships. The first section lists the embedded relationships by property name along with their corresponding target.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	`source`
object_markings_refs	`marking-definition`
Common Relationships	
`duplicate-of`, `related-to`	

1.11.3. Examples

A Source for an individual named John Smith

```
{
  "type": "source",
  ...,
  "name": "John Smith",
  "classification": "individual"
}
```

A Source for a company named ACME Widget, Inc.

```
{
  "type": "source",
  ...,
  "name": "ACME Widget, Inc.",
  "classification": "organization"
}
```

1.12. Threat Actor

Type Name: `threat-actor`

A Threat Actor object is a mechanism for describing and documenting an individual, a group of individuals, a class of actors, or an organization believed to be operating with malicious intent. These identities can either be specific (e.g. ThreatActor 'BadZebra' is 'John Smith') or refer to a class of individuals or organizations (e.g. ThreatActor 'ZebraGroup' is a Criminal Group located in Europe).

The Threat Actor object captures information about the characteristics of the Threat Actor (names, roles, sophistication level, resource level they have access to) and their objectives and motivations.

1.12.1. Properties

Common Properties		
type, id, created_by_ref, labels, version, created, modified, revoked, version_comment, external_references, confidence, object_markings_refs, granular_markings		
Threat Actor Specific Properties		
name, description, aliases, classification, roles, objectives, sophistication, resource_level, primary_motivation, secondary_motivations, personal_motivations, nationality		
Property Name	**Type**	**Description**
type (required)	`string`	The value of this field **MUST** be `threat-actor`

labels (required)	`list` of type `open-vocab`	This field specifies the type of threat actor, if known or suspected. This is an open vocabulary and values **SHOULD** come from the `threat-actor-label-ov` vocabulary.
name (required)	`string`	A name used to identify this Threat Actor or Threat Actor group.
description (optional)	`string`	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
aliases (optional)	`list` of type `string`	A list of other names that this Threat Actor is believed to use.
classification (required)	`string`	The type of entity that this Threat Actor describes, e.g. an individual, organization. This is an open vocabulary and the values **SHOULD** come from the `identity-classification-ov` vocabulary.
roles (optional)	`list` of type `string`	This is a list of roles the Threat Actor plays. This is an open vocabulary and the values **SHOULD** come from the `threat-actor-roles-ov` vocabulary.
objectives (optional)	`list` of type `open-vocab`	This field defines the Threat Actor's primary goal, objectives, desired outcomes, or intended effect — what the Threat Actor hopes to accomplish with a typical attack. However, with non-hostile Threat Actors, such as an untrained employee, the outcome may be unintentional. The Threat Actor may use many methods to achieve this goal, and the primary goal may have secondary or ancillary effects. This is an open vocabulary and values **SHOULD** come from the `attack-objective-ov` vocabulary.

sophistication (optional)	`open-vocab`	The skill, specific knowledge, special training, or expertise a Threat Actor must have to perform the attack. This is an open vocabulary and values **SHOULD** come from the `attack-sophistication-level-ov` vocabulary.
resource_level (optional)	`open-vocab`	This defines the organizational level at which this Threat Actor typically works, which in turn determines the resources available to this Threat Actor for use in an attack. This attribute is linked to the Sophistication Level attribute — a specific resource level implies that the Threat Actor has access to at least a specific sophistication level. This is an open vocabulary and values **SHOULD** come from the `attack-resource-level-ov` vocabulary.
primary_motivation (optional)	`open-vocab`	The primary reason, motivation, or purpose behind this Threat Actor. The primary motivation is the archetypical, single most prevalent and descriptive motivation of this Threat Actor. This motivation is intrinsic to the Threat Actor and the primary cause of the Threat Actor's actions. A small number of individuals may actually have a different motivation, but for -Threat Actor analysis this motivation is used as the analysis basis. This is an open vocabulary and values **SHOULD** come from the `attack-motivation-ov` vocabulary.
secondary_motivations (optional)	`list` of type `open-vocab`	The secondary reasons, motivations, or purposes behind this Threat Actor. Secondary motivations can exist as an equal or near-equal cause to the Primary Motivation. It does not replace or magnify the Primary Motivation, but might indicate additional asset or attack targeting. For

		example, Crime Syndicates have a Primary Motivation of Organizational Gain, which is the primary cause of their actions. However, they may also use the same actions to establish Dominance, which is a form of competitive advantage in many organized crime domains and must be constantly reinforced. This is an open vocabulary and values **SHOULD** come from the `attack-motivation-ov` vocabulary.
`personal_motivations` (optional)	`list` of type `open-vocab`	The personal reasons, motivations, or purposes of the Threat Actor regardless of organizational goals. Personal motivation, which is independent of the organization's goals, describes what impels an individual to carry out an attack. Personal Motivation may align with the organization's motivation—as is common with activists—but more often it supports personal objectives. For example, an individual analyst may join a Data Miner corporation because his or her values and skills align with the corporation's objectives. But the analyst most likely performs his or her daily work toward those objectives for personal reward in the form of a paycheck. The motivation of personal reward may be even stronger for Threat Actors who commit illegal acts, as it is more difficult for someone to cross that line purely for altruistic reasons. This is an open vocabulary and values **SHOULD** come from the `attack-motivation-ov` vocabulary.
`nationality` (optional)	`string`	The nationality of this Threat Actor. The value **MUST** be from [todo ISO Ref].

1.12.2. Relationships

These are the relationships explicitly defined between the Threat Actor object and other objects. The first section lists the embedded relationships by property name along with their corresponding target. The rest of the table identifies the relationships that can be made from this object by way of the Relationship Object. The reverse relationships (relationships "to" this object) are included as a convenience. For their definitions, please see the objects for which they represent a "from" relationship.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	`source`
object_markings_refs	`marking-definition`

Common Relationships			
`duplicate-of`, `related-to`			
Source	**Name**	**Target**	**Description**
`threat-actor`	`targets`	`victim-target`, `vulnerability`	This relationship is used to document the Victim Targets or Vulnerabilities that this Threat Actor targets.
`threat-actor`	`uses`	`attack-pattern`, `malware`, `tool`	This relationship is used to document the Attack Patterns, Malware, or Tools that a Threat Actor uses or that are used by a Threat Actor.
Reverse Relationships			
`campaign`, `incident`, `intrusion-set`,	`attributed-to`	`threat-actor`	See forward relationship for definition.
`indicator`	`indicates`	`threat-actor`	See forward relationship for definition.

1.12.3. Examples

```
{
  "type": "threat-actor",
  "id": "threat-actor--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "version": 1,
  "created": "2016-04-06T20:03:48Z",
  "modified": "2016-04-06T20:03:48Z",
  "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "name": "Evil Org",
  "description": "The Evil Org threat actor group"
}
```

1.13. Tool

Type Name: `tool`

A Tool object represents legitimate software (or in some cases grayware) that is used by threat actors to perform attacks. These tools or software packages are often found on a system and have legitimate purposes for power users, system administrators, network administrators, or even normal users.

The Tool object helps characterize the properties of these software tools and can be used as a basis for making an assertion about how a threat actor uses them during an attack. This object **MUST NOT** be used to document malware. Further, this object **MUST NOT** be used to document tools used as part of a course of action in response to an attack.

Remote access tools (RDP) and network scanning tools (NMAP) are examples of Tools that may be used by a threat actor during an attack. Knowing how and when threat actors use such tools is important for understanding how campaigns are executed.

1.13.1. Properties

Common Properties
type, id, created_by_ref, labels, version, created, modified, revoked, version_comment, external_references, confidence, object_markings_refs, granular_markings
Tool Specific Properties
name, description, tool_version, kill_chain_phases

Property Name	Type	Description
type (required)	`string`	The value of this field **MUST** be `tool`
labels (required)	`list` of type `open-vocab`	The kind(s) of tool(s) being described. This is an open vocabulary and values **SHOULD** come from the `tool-label-ov` vocabulary.
name (required)	`string`	The name used to identify the Tool.
description (optional)	`string`	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
tool_version (optional)	`string`	The version identifier associated with the tool.
kill_chain_phases (optional)	`array` of type `kill-chain-phase`	The list of Kill Chain Phases for which this tool instance can be used.

1.13.2. Relationships

These are the relationships explicitly defined between the Tool object and other objects. The first section lists the embedded relationships by property name along with their corresponding target. The rest of the table identifies the relationships that can be made from this object by way of the Relationship Object. The reverse relationships (relationships "to" this object) are included as a convenience. For their definitions, please see the objects for which they represent a "from" relationship.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	`source`
object_markings_refs	`marking-definition`
Common Relationships	

duplicate-of, related-to			
Source	**Name**	**Target**	**Description**
Reverse Relationships			
indicator	detects	tool	See forward relationship for definition
course-of-action	mitigates	tool	See forward relationship for definition
attack-pattern, campaign, intrusion-set, threat-actor	uses	tool	See forward relationship for definition

1.14. Examples

```
{
  "type": "tool",
  "id": "tool--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "version": 1,
  "created": "2016-04-06T20:03:48Z",
  "modified": "2016-04-06T20:03:48Z",
  "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "name": "VNC"
}
```

1.15. Victim Target

Type Name: victim-target

The Victim Target object describes the target of an attack. It is used both to represent actual targets, for example the victims of an incident, and general targets, such as "energy companies in Europe".

1.15.1. Properties

Common Properties

type, id, created_by_ref, labels, version, created, modified, revoked, version_comment, external_references, confidence, object_markings_refs, granular_markings		
Target Specific Properties		
name, description, classification, roles, sectors, regions		
Property Name	**Type**	**Description**
type (required)	`string`	The value of this field **MUST** be `victim-target`
name (required)	`string`	The name of this Victim Target. When referring to a specific entity (e.g., an individual or organization), this field **SHOULD** contain the canonical name of the specific entity.
description (optional)	`string`	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
classification (required)	`string`	The type of entity that this Victim Target describes, e.g. an individual, organization. This is an open vocabulary and the values **SHOULD** come from the `identity-classification-ov` vocabulary.
roles (optional)	`list` of type `string`	The list of roles that this Victim Target performs (eg. CEO, Domain Administrators, Doctors, Hospital, or Retailer)
sectors (optional)	`list` of type `open-vocab`	The list of sectors that the Victim Target of the attack belongs to. This is an open vocabulary and values **SHOULD** come from the `industry-sector-ov` vocabulary.
regions (optional)	`list` of type `string`	The list of regions (localities,

		nationalities) for this Victim Target. When representing nationalities, the value **MUST** be from [ISO Ref].

1.15.2. Relationships

These are the relationships explicitly defined between the Victim Target object and other objects. The first section lists the embedded relationships by property name along with their corresponding target. The rest of the table identifies the relationships that can be made from this object by way of the Relationship Object. The reverse relationships (relationships "to" this object) are included as a convenience. For their definitions, please see the objects for which they represent a "from" relationship.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships			
`created_by_ref`		`source`	
`object_markings_refs`		`marking-definition`	
Common Relationships			
`duplicate-of`, `related-to`			
Source	**Name**	**Target**	**Description**
Reverse Relationships			
`incident`	`exploits`	`victim-target`	See forward relationship for definition.
`attack-pattern`, `campaign`, `incident`, `intrusion-set`, `malware`, `threat-actor`	`targets`	`victim-target`	See forward relationship for definition.

1.15.3. Examples

Targeting of domain administrators:

```
{
  "type": "victim-target",
  "id": "victim-target--0c7b5b88-8ff7-4a4d-aa9d-feb398cd0061",
  "version": 1,
  "created": "2016-05-12T08:17:27.000000Z",
  "modified": "2016-05-12T08:17:27.000000Z",
  "name": "Domain Administrators",
  "classification": "class",
  "roles": ["domain-administrators"]
}
```

Targeting of hospitals in the United States:

```
{
  "type": "victim-target",
  "id": "victim-target--0c7b5b88-8ff7-4a4d-aa9d-feb398cd0061",
  "version": 1,
  "created": "2016-05-12T08:17:27.000000Z",
  "modified": "2016-05-12T08:17:27.000000Z",
  "name": "Hospitals in the United States",
  "classification": "class",
  "roles": ["hospitals"],
  "sectors": ["healthcare"],
  "regions": ["us"]
}
```

Targeting of the British military:

```
{
  "type": "victim-target",
  "id": "victim-target--0c7b5b88-8ff7-4a4d-aa9d-feb398cd0061",
  "version": 1,
  "created": "2016-05-12T08:17:27.000000Z",
  "modified": "2016-05-12T08:17:27.000000Z",
  "name": "British Military",
  "classification": "organization",
  "roles": ["military"],
  "regions": ["gb"]
}
```

1.16. Vulnerability

Type Name: `vulnerability`

A Vulnerability is a mistake in software that can be directly used by a hacker to gain access to a system or network. Each Attack Pattern defines how an adversary may attempt to compromise the target, provides a description of the common technique(s) used, and presents recommended methods for mitigating the threat described by the Attack Pattern. The Attack Pattern object helps categorize attacks in a meaningful way to provide coherent and detailed information about how attacks are performed.

In STIX, the Vulnerability object is primarily used to represent identifiers in external libraries or to link to external definitions. It is used as a linkage to the asset management and compliance process.

1.16.1. Properties

Common Properties		
`type`, `id`, `created_by_ref`, `labels`, `version`, `created`, `modified`, `revoked`, `version_comment`, `external_references`, `confidence`, `object_markings_refs`, `granular_markings`		
Attack Pattern Specific Properties		
`name`, `description`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this field MUST be `vulnerability`
`external_references` (optional)	`list` of type `external-reference`	A list of external references which refer to non-STIX information. This field **MAY** be used to provide one or more Vulnerability identifiers, such as a CVE ID [TODO: add reference]. When specifying a CVE ID, the **`source`** field of the external reference **MUST** be set to `cve` and the **`external_id`** field **MUST** be the exact CVE identifier.
`name` (required)	`string`	The name used to identify the Vulnerability.
`description` (optional)	`string`	A description that provides more details and context about the Vulnerability, potentially including its purpose and its key characteristics.

1.16.2. Relationships

These are the relationships explicitly defined between the Vulnerability object and other objects. The first section lists the embedded relationships by property name along with their corresponding target. The rest of the table identifies the relationships that can be made from this object by way of the Relationship Object. The reverse relationships (relationships "to" this object) are included as a convenience. For their definitions, please see the objects for which they represent a "from" relationship.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships			
`created_by_ref`		`source`	
`object_markings_refs`		`marking-definition`	
Common Relationships			
`duplicate-of`, `related-to`			
Source	**Name**	**Target**	**Description**
Reverse Relationships			
`attack-pattern`, `malware`	`exploits`	`vulnerability`	See forward relationship for definition.
`campaign`, `intrusion-set`, `threat-actor`	`targets`	`vulnerability`	See forward relationship for definition.
`course-of-action`	`mitigates`	`vulnerability`	See forward relationship for definition.

1.16.3. Examples

```
{
  "type": "vulnerability",
  "id": "vulnerability--0c7b5b88-8ff7-4a4d-aa9d-feb398cd0061",
```

```
  "version": 1,
  "created": "2016-05-12T08:17:27.000000Z",
  "modified": "2016-05-12T08:17:27.000000Z",
  "name": "CVE-2016-1234"
  "external_references": [
    {
      "source": "cve",
      "id": "CVE-2016-1234"
    }
  ]
}
```

2. Relationship Objects

<TODO> In STIX there are three types of relationships:

- Embedded relationships that represent "facts", such as **`created_by_ref`**;
- Named relationships that use the generic Relationships Object (covered in section 2.1, below); and
- Relationships that use a specific relationship object, like the Sighting object (covered in Section 2.2, below).

2.1. Relationship

Type Name: `relationship`

This object is used to link together other SDOs, such as Indicator, Observed Data, and Threat Actor in order to describe how those SDOs are related to each other. If other SDOs are considered “nodes” or “vertices” in the graph, the relationship object represents “edges”.

STIX defines many named relationships to link together SDOs. These named relationships are contained in the "Relationships" table under each SDO definition. Named relationships **SHOULD** be used whenever possible to ensure consistency. An example of a named relationship is that an `indicator` `indicates` a `campaign`.

STIX also allows relationships from any SDO to any SDO that have not been defined in the specification. These relationships **MAY** use the `related-to` relationship name or **MAY** use a custom relationship name. Custom relationship names **SHOULD** be all lowercase (where lowercase is defined by the locality conventions) and **SHOULD** use dashes instead of spaces or underscores. As an example, a user might want to link `observed-data` directly to a `tool`. They can do so using `related-to` to say that the Observed Data is related to the Tool but not

describe how, or they could use `has-executable` (a custom name they determined) to indicate more detail.

Note that some relationships in STIX may seem like "shortcuts". For example, an Indicator doesn't really detect a Campaign: it detects activity (Attack Patterns, Malware, etc.) that are often used by that campaign. While some analysts might want all of the source data and think that shortcuts are "wrong", in many cases it's helpful to provide just the key points (shortcuts) and leave out the low-level details. In other cases, the low-level analysis may not be sharable while the high-level analysis is. For these reasons, relationships that might appear to be "shortcuts" are included in STIX.

2.1.1. Named Relationships Summary

Source	Name	Target	Source	Name	Target
attack-pattern	exploits	vulnerability	indicator	detects	attack-pattern
attack-pattern	targets	victim-target	indicator	detects	malware
attack-pattern	uses	malware	indicator	detects	tool
attack-pattern	uses	tool	indicator	indicates	campaign
campaign	attributed-to	intrusion-set	indicator	indicates	intrusion-set
campaign	attributed-to	threat-actor	indicator	indicates	threat-actor
campaign	targets	victim-target	intrusion-set	attributed-to	threat-actor
campaign	targets	vulnerability	intrusion-set	targets	victim-target
campaign	uses	attack-pattern	intrusion-set	targets	vulnerability
campaign	uses	malware	intrusion-set	uses	attack-pattern
campaign	uses	tool	intrusion-set	uses	malware
course-of-action	mitigates	attack-pattern	intrusion-set	uses	tool
course-of-action	mitigates	incident	malware	exploits	vulnerability
course-of-action	mitigates	malware	malware	targets	victim-target
course-of-action	mitigates	tool	malware	variant-of	malware
course-of-action	mitigates	vulnerability	threat-actor	targets	victim-target
incident	attributed-to	attack-pattern	threat-actor	targets	vulnerability

incident	attributed-to	campaign	threat-actor	uses	attack-pattern
incident	attributed-to	intrusion-set	threat-actor	uses	malware
incident	attributed-to	malware	threat-actor	uses	tool
incident	attributed-to	threat-actor			
incident	exploits	victim-target			
incident	targets	victim-target			
incident	uses	course-of-action			

2.1.2. Properties

Common Properties		
type, id, created_by_ref, labels, version, created, modified, revoked, version_comment, external_references, confidence, object_markings_refs, granular_markings		
Relationship Specific Properties		
name, description, source_ref, target_ref		
Property Name	**Type**	**Description**
type (required)	string	The value of this field **MUST** be relationship
name (required)	string	The name used to identify the Relationship. This value **SHOULD** be an exact value listed in the relationships for the source and target SDO, but **MAY** be any string.
description (optional)	string	A description that provides more details and context about this object, potentially including its purpose and its key characteristics.
source_ref (required)	identifier	The **id** of the source (from) object series.
target_ref (required)	identifier	The **id** of the target (to) object series.

2.1.3. Relationships

These are no relationships explicitly defined between the Relationship object and other objects, other than those defined as common relationships. The first section lists the embedded relationships by property name along with their corresponding target.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
`created_by_ref`	`source`
`object_markings_refs`	`marking-definition`
Common Relationships	
`duplicate-of`, `related-to`	

2.2. Sighting

Type Name: `sighting`

Sighting is a special type of Relationship (an SRO) that represent events of security interest, often detected via indicators or analytics, and are used to communicate that the indicator or analytic was "sighted". It can be tied to how it was discovered (indicator, analytic, or even a description of human analysis) and to what was actually seen (set of Observed Data). It could also be tied to other objects to indicate that, for example, a campaign was spotted. Sightings differ from Observed Data in that you can tie a Sighting to something that triggered it with additional context regarding when it was seen and how many times it was seen, whereas the Observed Data is simply the CybOX wrapper in STIX.

This object will be particularly useful in the context of threat intelligence sharing within trust circles because it gives analysts from different organizations the opportunity to acknowledge that a particular phenomenon was “seen” in multiple places. It adds an SRO that can be used to crowdsource CTI and thereby quantify the phenomenon. It can also be used to support judgements about confidence ratings.

2.2.1. Properties

Common Properties		
type, id, created_by_ref, labels, version, created, modified, revoked, version_comment, external_references, confidence, object_markings_refs, granular_markings		
Sighting Specific Properties		
first_seen, first_seen_precision, last_seen, last_seen_precision, count, sighting_of_ref, observed_data_refs, where_sighted_refs, summary		
Property Name	**Type**	**Description**
type (required)	`string`	The value of this field **MUST** be `sighting`
first_seen (required)	`timestamp`	The time that this sighting was first seen.
first_seen_precision (optional)	`timestamp-precision`	The precision of the **first_seen** timestamp.
last_seen (required)	`timestamp`	The last time this sighting was seen. For single point in time sighting, this should match the **first_seen** time. If the count equals 1, then the `first_seen` and `last_seen` **MUST** be equal.
last_seen_precision (optional)	`timestamp-precision`	The precision of the **last_seen** timestamp.
count (optional)	`integer`	This is an integer between 0 and 999,999,999 inclusive and represents the number of times the object was sighted. Both `observed-data` and `sighting` have count fields. The count fields of the sighting and any `observed-data` instances that are reference should

		be interpreted independently of each other (the counts are not multiplicative or additive). In other words, a Sighting with a count of 14 means that the sighting was seen 14 times, even if it links to an `observed-data` with a count of 10 and another with a count of 2. Counts on the referenced `observed-data` may add up to the count on the sighting, but may not.
`sighting_of_ref` (required)	`identifier`	An ID reference to the object that has been sighted. For example, if this sighting is a sighting of an Indicator, that indicator's ID.
`observed_data_refs` (optional)	`list` of type `identifier`	A list of ID references to the Observed Data that were seen. This is used when for example you have an indicator watch list with hundreds of IPs and you need to sight a single IP address.
`where_sighted_refs` (optional)	`list` of type `identifier`	The ID of the Victim Target objects of the entities that saw the sighting. Omitting the **`where_sighted_refs`** field does not imply that the sighting was seen by the object creator. To indicate that the sighting was seen by the object creator, the object creator's ID **MUST** be listed in **`where_sighted_refs`**.
`summary` (optional)	`boolean`	Whether the data should be considered primary source data (and therefore considered for counts) or summary data (in which case it may overlap or summarize primary source or other summary data). Default value is `false`.

2.2.2. Relationships

These are no relationships explicitly defined between the Sighting object and other objects, other than those defined as common relationships. The first section lists the embedded relationships by property name along with their corresponding target.

Relationships are not restricted to those listed below. Relationships can be created between any objects using the `related-to` relationship name or, as with open vocabularies, user-defined names.

Embedded Relationships	
created_by_ref	`source`
object_markings_refs	`marking-definition`
sighting_of_ref	`identifier`
observed_data_refs	`identifier`
where_sighted_refs	`identifier`
Common Relationships	
`duplicate-of`, `related-to`	

2.2.3. Examples

Sighting of Indicator, without Observed Data

```
{
  "type": "sighting",
  "id": "sighting--ee20065d-2555-424f-ad9e-0f8428623c75",
  "created": "2016-04-06T20:08:31Z",
  "modified": "2016-04-06T20:08:31Z",
  "version": 1,
  "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "sighting_of_ref": "indicator--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f"
}
```

Sighting of Indicator, with Observed Data (what exactly was seen) and where it was seen

```
[
  {
    "type": "sighting",
    "id": "sighting--ee20065d-2555-424f-ad9e-0f8428623c75",
```

```
    "created": "2016-04-06T20:08:31Z",
    "modified": "2016-04-06T20:08:31Z",
    "version": 1,
    "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
    "sighting_of_ref": "indicator--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
    "observed_data_refs": [ "observed-data--b67d30ff-02ac-498a-92f9-32f845f448cf" ],
    "where_sighted_refs": [ "source--b67d30ff-02ac-498a-92f9-32f845f448ff" ],
    "first_sighted": "2015-12-21T19:00:00Z",
    "last_sighted": "2015-12-21T19:00:00Z",
    "count": 50
  },
  {
    "type": "observed-data",
    "id": "observed-data--b67d30ff-02ac-498a-92f9-32f845f448cf",
    "created": "2016-04-06T19:58:16Z",
    "modified": "2016-04-06T19:58:16Z",
    "version": 1,
    "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
    "start": "2015-12-21T19:00:00Z",
    "stop": "2016-04-06T19:58:16Z",
    "count": 50,
    "cybox": {
      "objects": [
        {
          "type": "file-object",
          "file_name": "malware.exe",
          "hashes": {
            "md5": "3773a88f65a5e780c8dff9cdc3a056f3",
            "sha1": "cac35ec206d868b7d7cb0b55f31d9425b075082b"
          }
        }
      }
    ]
  }
]
```

3. Metadata Objects

3.1. STIX Bundle

Type Name: `bundle`

A `bundle` is a collection of STIX Objects grouped together to enable them to be exchanged across interoperable systems. A bundle is not a standard STIX object itself and is only used to group STIX objects for exchange. It can be thought of as an envelope, enabling the delivery of the objects it contains. It does not have any of the Common Properties other than the **`type`** and **`id`** fields. The `bundle` object should not be treated as a persistent object.

3.1.1. Properties

Property Name	Type	Description
type (required)	`string`	Indicates that this object is a STIX Bundle. The value of this field **MUST** be `bundle`
id (required)	`identifier`	An identifier for this bundle. The **id** field for the bundle is designed to help tools that may need it for processing, but tools are not required to store or track it. Consuming tools should not rely on the presence of this field.
spec_version (required)	`spec-version-enum`	The version of the STIX specification used to represent the content in this bundle. This enables non-TAXII transports or other transports without their own content identification mechanisms to know the version of STIX content.
attack_patterns (optional)	`list` of type `attack-pattern`	Specifies a set of one or more Attack Patterns.
campaigns (optional)	`list` of type `campaign`	Specifies a set of one or more Campaigns.
courses_of_action (optional)	`list` of type `course-of-action`	Specifies a set of one or more Courses of Action that could be taken in regard to one of more cyber threats.
incidents (optional)	`list` of type `incidents`	Specifies a set of one or more cyber threat Incidents.
indicators (optional)	`list` of type `indicator`	Specifies a set of one or more cyber threat Indicators.
intrusion_sets (optional)	`list` of type `intrusion-set`	Specifies a set of one or more cyber threat Intrusion Sets.
malware (optional)	`list` of type `malware`	Specifies a set of one or more Malware TTPs.

marking_definitions (optional)	list of type marking-definition	Specifies a set of one or more Marking Definitions.
observed_data (optional)	list of type observed-data	Specifies a set of one or more piece of Observed Data.
relationships (optional)	list of type relationship	Specifies a set of one or more relationships between SDOs.
reports (optional)	list of type report	Specifies a set of one or more reports.
sightings (optional)	list of type sighting	Specifies a set of one or more sightings.
sources (optional)	list of type source	Specifies a set of one or more individual or organizational sources
threat_actors (optional)	list of type threat-actor	Specifies a set of one or more Threat Actors.
tools (optional)	list of type tool	Specifies a set of one or more Tools.
victim_targets (optional)	list of type victim-target	Specifies a set of one or more Victim Targets.
vulnerabilities (optional)	list of type vulnerability	Specifies a set of one or more Vulnerability.
custom_objects (optional	list of type custom-object	Specifies a list of one or more custom objects.

3.1.2. Relationships

This object is not a SDO and **MUST NOT** have any relationships to it or from it.

3.1.3. Examples

```
{
  "type": "bundle",
  "spec_version": "2.0”,
  "indicators": [
    {
      "type": "indicator",
      "id": "indicator--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
      "created_by_ref": "source--f431f809-377b-45e0-aa1c-6a4751cae5ff",
      "created": "2016-04-29T14:09:00.123456Z",
```

```
      "version": 1,
      "modified": "2016-04-29T14:09:00.123456Z",
      "object_marking_refs": ["marking-definition--089a6ecb-cc15-43cc-9494-767639779123"],
      "name": "Poison Ivy Malware",
      "description": "This file is part of Poison Ivy",
      "pattern": "file-object.hashes.md5 = '3773a88f65a5e780c8dff9cdc3a056f3'"
    }
  ],
  "marking_definitions": [
    {
      "type": "marking-definition",
      "id": "marking-definition--089a6ecb-cc15-43cc-9494-767639779123",
      "created": "2016-02-19T09:11:01Z",
      "modified": "2016-02-19T09:11:01Z",
      "version": 1",
      "definition": {
        "type": "tlp",
        "tlp": "green"
      }
    }
  ]
}
```

4. Vocabularies

STIX supports three types of vocabularies:

- **Open vocabularies** (indicated throughout this specification as ‘ov’), which provide a listing of common and industry accepted terms as a guide to the user but do not limit the user to that defined list; and
- **Closed vocabularies**, (indicated throughout this specification as ‘cv’), which provide a discrete listing of the terms to be used to further describe that property or relationship object, and provide extensions in a separate field.
- **Enumerations**, (indicated throughout this specification as 'enum'), which provide a hardcoded list of terms defined by the specification that cannot be expanded.

The following sections provide object-specific listings for each of the vocabularies referenced in the object description sections.

4.1. Attack Motivation

Type Name: `attack-motivation-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Campaign
- Intrusion Set
- Threat Actor

Knowing a threat Threat Actors' or Campaign's motivation narrows which targets that actor or campaign may focus on. For example, crime syndicates, who have a strong profit motive, will generally only take assets they can easily convert to cash regardless of the discoverability of their actions, while Threat Actors seeking notoriety will ignore attacks on non-visible assets that will not bring them attention.

Understanding the Threat Actors' or Campaign's intent helps defenders focus their often-limited defense resources on the most likely attack scenarios for any particular asset.

Motivation shapes the intensity and the persistence of an attack. Threat Actors and Campaigns usually act in a manner that reflects their underlying emotion or situation, and this informs defenders of the manner of attack. For example, a spy motivated by nationalism (ideology) likely has the patience to achieve long-term goals and work quietly for years, whereas a cyber-vandal out for kicks can create an intense and attention-grabbing attack but quickly loses interest and moves on. Understanding these differences allows defenders to implement controls tailored to each type of attack for greatest efficiency.

Motivation helps in describing threat and risk scenarios in less technical terms. Analysts must eventually convey all Threat Actor analysis to others in their organization who can act to help mitigate the risks. Describing Attack Motivation tells a fuller, more relatable story to colleagues of all security levels. Communicating risks in a more understandable fashion obviously leads to faster implementation of more effective defenses.

This section including vocabulary items and their descriptions is based on and contains copied text from the Threat Agent Motivations publication from Intel Corp in Feb 2015[3]

Vocabulary Summary	
`accidental`, `coercion`, `dominance`, `ideology`, `notoriety`, `organizational-gain`, `personal-gain`, `personal-satisfaction`, `revenge`, `unpredictable`	
Vocabulary Value	**Description**
`accidental`	Benevolent or harmless intent but with actions that inadvertently cause harm.

[3] Intel Corp Threat Agent Motivations Feb 2015

	This element generally describes the non-hostile Threat Actor, such as a well-meaning, dedicated employee who through distraction or poor training unintentionally causes harm to his or her organization. A common occurrence is an employee who was quickly assigned additional duties to cover for laid-off employees but has not yet received proper training. With a heavy workload and lacking a full understanding of the tasks, the employee is bound to make mistakes, unwittingly and possibly without even knowing a mistake has occurred.
`coercion`	Forced to act illegally on behalf of another. Unlike the other Motivations, a coerced person does not act for personal gain, but out of fear of incurring a loss. These individuals have been forced through intimidation or blackmail to act for someone else's benefit and are conducting acts they probably would not normally do and that may even directly conflict with their own self-interests. In most cases, a coerced person is just as much a victim as the attack target. Coercion can effectively force a person to commit very harmful, possibly violent actions, if the threat against him or her is severe enough. Coercion can also subvert employees often considered above reproach, such as executives or those who undergo regular security checks. There are probably fewer total threat Threat Actors driven by Coercion than by the other Motivations, but it can be a motivator for almost any kind of threat and must be considered when analyzing and planning for risks.
`dominance`	Attempting to assert superiority over another. Dominance can take many forms at many scales, for example, physically intimidating a coworker, threatening to expose sensitive data of a corporation, or amassing an army along a border. But in all cases Threat Actors use whatever power they have to bully others into submission. Threat Actors seeking dominance may also steal information assets to create power and build toward a goal of dominance. Collection can include compromising items such as sensitive intellectual property, personal information, business data, product data, and information on operational aspects such as networks and supply chains. Access to these items allows an attacker to leverage them or their vulnerabilities during an

	attack. For example, to prepare for a cyber attack during a future national conflict, a government spy may steal software bug reports from a network device manufacturer, which detail the device's vulnerabilities and enable cyberattacks. Ideology and Dominance may both be present in some state-sponsored Threat Actors, but Dominance can occur with or without Ideology. Crime Syndicates often act to establish dominance in extreme acts of bullying but not in support or in conjunction with some higher objective—that is, Ideology. Vandalism and hacking are also included under Dominance, because cyber vandals typically seek dominance over others through bullying. Other factors, such as Notoriety, may also be present to some degree in these Threat Actors, but Dominance is the primary motivator.
`ideology`	A passion to express a set of ideas, beliefs, and values that shapes and drives harmful acts. Threat Actor who act for ideological reasons (e.g. political, religious, human rights, environmental, etc.) are not usually motivated primarily by the desire for profit, they are acting on their own sense of morality, justice, or political loyalty. Ideological motivation can arise independent of any prior interaction with the target. For example, an activist group may sabotage a company's equipment because they believe the company is harming the environment even though the activists may have never actually used any of the company's products. Because ideologies vary, so do the types of threat posed by individuals or organizations with this motivation. The threat may come in the form of a direct attack, such as sabotage, theft, or exposure of sensitive information. It may also happen indirectly, such as an employee who improperly uses company computers to participate in a cyberattack against an organization the employee believes to be oppressive. If traced back, the attacked organization could launch a counterattack or bring legal action against the unsuspecting "attacking" company.
`notoriety`	Seeking to become well known for harmful activity. Threat Actors motivated by Notoriety are often seeking either personal validation or respect within a community. The actions used to achieve even unreasonable notoriety may be

	quite well reasoned and strategic. Similar to vandalism, the individual or group may seek to cause damage for its own sake, but staying covert is not a priority—quite the opposite, in fact. To garner the respect of their target audience, the actions that those seeking notoriety take are not tempered by a need for secrecy and therefore can be extreme in scope and damage.
`organizational-gain`	Seeking an advantage over a competitor's organization. The prospect of increased profit or other gains through an unfairly obtained competitive advantage has always been a powerful incentive, and the temptation to cheat will always be too strong for some to resist. Through theft of information such as intellectual property, business processes, or supply chain agreements, a competitor bypasses the lengthy and expensive process of developing it themselves, accelerating its position in a market or capability. The inappropriate acquisition or misuse of information, even seemingly esoteric data such as employee demographics, could be used to gain a competitive edge. Information theft is not the only option used to get ahead. A competitor could also choose sabotage, lawsuits, or other non-theft means to undermine a competing organization to gain an advantage. Organizational Gain includes military objectives as well. A military organization can use stolen information to advance its own technology while also enabling careful study of their target's capabilities and vulnerabilities. In some cases, individuals with similar objectives may work collaboratively to advance their own personal gain but do so under voluntary organizational rules, such as military mercenaries or hacktivist collectives. In these cases, the organization motivation reflects the individual's' motivation.
`personal-gain`	Improve one's own financial status. A selfish desire for personal gain motivates many crimes. This element describes individuals who steal money in some way or conduct activities that will net them money, such as hacking in exchange for a paycheck. These individuals are most likely indifferent to the damage caused by their actions, but apart from stealing, usually do not go out of their way to harm their target.

	This motivation is different from Organizational Gain in the timeliness of the threat. Usually, the individual stealing assets wants to make a quick profit by selling them, rather than invest the time and expertise needed to craft a package for sale like an organization might create. Financial fraud is a result of this motivation, as is physical theft of valuable items. Intellectual property theft for sale is also a result of this motivation, but in the special case of espionage, Ideology may also play a significant part in an individual's motivation. An individual Threat Actor may be seeking personal gain, but this does not mean that the Threat Actor always acts alone. Many criminal groups, organized or not, are often made up of individuals banded together solely to maximize their own personal profits. In addition to greed, the need to steal can stem from other factors, such as pressing medical or addiction debts, poverty, coercion, disgruntlement, or mental impairment. These issues can easily lead an otherwise honest individual to commit illegal acts. Personal Financial Gain can also apply to individuals working for an organizational Threat Actor, such as a Competitor or Organized Crime. While the organization seeks an advantage for its collective goals, the individuals working for that organization may be driven by more personal reasons that may have little to do the organization's objectives. Often this personal motivation is simply the personal financial gain that results from supporting the organization, such as a paycheck or a cut of the spoils. (See the Personal Motivation modifier).
`personal-satisfaction`	Fulfilling an emotional self-interest. Some people may cause harm when they act not to support a financial or ideological objective but to satisfy a strictly internal, personal interest. This personal interest can be expressed in many ways, such as intrusive curiosity or thrill seeking, like children who break into a building just for the excitement of going where they are not allowed. More harmful possibilities include a healthcare worker who inappropriately reads the medical records of celebrities to see what treatment they are receiving or a hacker who attacks a website primarily because he or she enjoys the lawlessness of the act. Most "crimes of passion"—those caused by love, anger, fear, and so on— also fall under Personal Satisfaction.

	Threat Actors driven by Personal Satisfaction may incidentally receive some other gain from their actions, such as a profit, but their primary motivation is to gratify a personal, emotional need. This personal interest does not preclude people from banding together with other like-minded individuals toward a mutual, but not necessarily organizational, objective.
`revenge`	A desire to avenge perceived wrongs through harm. Most people go through stages of dissatisfaction with their employer or with a company they have done business with, but usually the situation resolves without illegal behavior. When the grievance (real or perceived) is severe and the situation escalates, a disgruntled person can seek revengeful and harmful retaliation. Unlike Ideology, Disgruntlement implies there is a history of some direct interaction with the target organization. A disgruntled Threat Actor seeking Revenge can include employees or former employees, all of whom may have extensive knowledge that the actors can leverage when conducting attacks. Often a disgruntled individual acts alone but may join an organization, whether a competitor, group of similar individuals, or criminal organization, if the individual believes that doing so will enable him or her to better harm the source of his or her anger. Predicting the actions of a disgruntled person or group is very difficult, because the action can take many forms including sabotage, violence, theft, fraud, or embarrassing individuals or the organization.
`unpredictable`	Acting without identifiable reason or purpose and creating unpredictable events. It may seem that since Unpredictable represents the actions one cannot anticipate, there is no need to include it—everyone knows life has many surprises. However, explicitly recognizing the potential for unpredictability in an environment and comprehending it in planning is essential for effective risk management. We include Unpredictable here to enable and support the discipline of planning for the unexpected. In its application as a Threat Actor Motivation, Unpredictable is not a miscellaneous or default category. It does not include

	acts such as a sudden DDOS attack on a company website or a new type of email phishing campaign. Those events may have occurred unexpectedly and the methods may have been novel, but a reasonable person could easily anticipate those kinds of events would occur at some point. In this taxonomy, Unpredictable means a truly random and likely bizarre event, which seems to have no logical purpose to the victims. In many cases Unpredictable acts will be the actions of a mentally ill person, such as the near-assassination of U.S. President Ronald Reagan in 1981 by a man who acted not for political reasons but because he believed his act would attract the love of a movie actress he had never met. Unpredictable acts can also come from competent Threat Actor with a new or unanticipated purpose. For example, in 2012 a small anarchist group began shooting at scientists employed at various nanotech companies. (Several people were injured but no one was killed.) The anarchists targeted them because they believed the scientists were working on implantable microcircuits that governments could use to monitor the thoughts of its citizens. For our purposes, it is not the anarchists' misguided conclusions that make them Unpredictable in an analysis, but in acting so violently against a target no one expected. To them, their conclusions were perfectly rational and their actions reasonable for the situation. To the security managers of those companies—who probably had anticipated and prepared for physical threats to the CEO and other corporate officers—the extreme ambush attacks on presumably lesser targets were not practically foreseeable and so would fall into the Unpredictable motivation.

4.2. Attack Objective

Type Name: `attack-objective-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Campaign
- Intrusion Set
- Threat Actor

Attack Objective is an open vocabulary that captures the objectives of a particular attacker or series of attacks (campaign or intrusion set). The vocabulary is divided into two sections: the high-level objectives come from the Intel Threat Agent Library publication and describe the broad objectives that the attacker wants to achieve. The specific means capture the mechanisms by which those objectives are achieved. For example, an attacker interested in damage might choose to cause damage to an organization by damaging their brand.

This vocabulary captures both types of objectives to simplify usage and avoid divergence in practice.

This section including vocabulary items and their descriptions is based on and contains copied text from the Threat Agent Library publication from Intel Corp in Sept 2007[4]

Vocabulary Summary	
`acquisition-theft`, `business-advantage`, `damage`, `embarrassment`, `technical-advantage`	
Vocabulary Value	**Description**
High-Level Objectives	*The broad objectives that the attacker wishes to achieve.*
`acquisition-theft`	Illicit acquisition of valuable assets for resale or extortion in a way that preserves the assets' integrity but may incidentally damage other items in the process.
`business-advantage`	Increased ability to compete in a market with a given set of products. The goal is to acquire business processes or assets.
`damage`	Injury to personnel, physical or electronic assets, or intellectual property.
`embarrassment`	Public portrayal in an unflattering light, causing to lost influence, credibility, competitiveness, or stock value.
`technical-advantage`	Illicit improvement of a specific product or production capability. The primary target is to acquire production processes or assets rather than a business process.
Specific Means / Goals	*The specific means by which the attacker attempts to achieve the above high-level objectives.*

[4] Intel Corp Threat Agent Library Sept 2007

account-takeover	Obtain control over an account (financial, etc)
brand-damage	Cause some brand damage on the target
credential-theft	Theft of credentials in bulk
degradation-of-service	Reducing the level of services provided by the target
extortion	Force the payment of some sort to prevent the attacker from taking some action.
harassment	Pressure or intimidate the target
identity-theft	Theft of the target's identity
intellectual-property-theft	Theft of intellectual property

4.3. Attack Resource Level

Type Name: attack-resource-level-ov

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Campaign
- Intrusion Set
- Threat Actor

Attack Resource Level is an open vocabulary that captures the general level of resources that a threat actor, intrusion set, or campaign might have access to. It ranges from individual, a person acting alone, to government, the resources of a national government.

This section including vocabulary items and their descriptions is based on and contains copied text from the Threat Agent Library publication from Intel Corp in Sept 2007[5]

Vocabulary Summary	
individual, club, contest, team, organization, government	
Vocabulary Value	**Description**

[5] Intel Corp Threat Agent Library Sept 2007

`individual`	Resources limited to the average individual; Threat Actor acts independently. Minimum Sophistication level: None.
`club`	Members interact on a social and volunteer basis, often with little personal interest in the specific target. An example might be a core group of unrelated activists who regularly exchange tips on a particular blog. Group persists long term. Minimum Sophistication level: Novice.
`contest`	A short-lived and perhaps anonymous interaction that concludes when the participants have achieved a single goal. For example, people who break into systems just for thrills or prestige may hold a contest to see who can break into a specific target first. It also includes announced "operations" to achieve a specific goal, such as the original "OpIsrael" call for volunteers to disrupt all Israel internet functions for a day. Minimum Sophistication level: Practitioner.
`team`	A formally organized group with a leader, typically motivated by a specific goal and organized around that goal. Group persists long term and typically operates within a single geography. Minimum Sophistication level: Practitioner.
`organization`	Larger and better resourced than a Team; typically a company or crime syndicate. Usually operates in multiple geographies and persists long term. Minimum Sophistication level: Expert.
`government`	Controls public assets and functions within a jurisdiction; very well resourced and persists long term. Minimum Sophistication level: Expert.

4.4. Attack Sophistication Level

Type Name: `attack-sophistication-level-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Threat Actor

The threat actor sophistication vocabulary captures the skill level of a threat actor. It ranges from "none", which describes a complete novice, to "innovator", which describes an actor who is able to create their own types of attacks and discover 0-days. This vocabulary is separate from resource level: an innovative, highly-skilled threat actor may have access to very few resources while a practitioner-level actor might have the resources of an organized crime ring.

This section including vocabulary items and their descriptions is based on and contains copied text from the Threat Agent Library publication from Intel Corp in Sept 2007[6]

Vocabulary Summary	
`none`, `novice`, `practitioner`, `expert`, `innovator`	
Vocabulary Value	**Description**
`none`	Has average intelligence and ability and can easily carry out random acts of disruption or destruction, but has no expertise or training in the specific methods necessary for a targeted attack.
`novice`	Can copy and use existing techniques. Example: Untrained Employee. Demonstrates a nascent capability. A novice has basic computer skills and likely requires the assistance of a Practitioner or higher to engage in hacking activity. He uses existing and frequently well known and easy-to-find techniques and programs or scripts to search for and exploit weaknesses in other computers on the Internet and lacks the ability to conduct his own reconnaissance and targeting research..
`practitioner`	Has a demonstrated, albeit low, capability. A practitioner possesses low sophistication capability. He does not have the ability to identify or exploit known vulnerabilities without the use of automated tools. He is proficient in the basic uses of publicly available hacking tools, but is unable to write or alter such programs on his own.
`expert`	Expert in technology and attack methods, and can both apply existing attacks and create new ones to greatest advantage. Example: Legal Adversary. Demonstrates advanced capability. An actor possessing expert capability has the ability to modify existing programs or codes but does not have the capability to script sophisticated programs from scratch. The expert has a working knowledge of networks, operating systems, and possibly even defensive techniques and will typically exhibit some operational security.
`innovator`	Demonstrates sophisticated capability. An innovator has the ability to create and script unique programs and codes targeting virtually

[6] Intel Corp Threat Agent Library Sept 2007

	any form of technology. At this level, this actor has a deep knowledge of networks, operating systems, programming languages, firmware, and infrastructure topologies and will demonstrate operational security when conducting his activities. Innovators are largely responsible for the discovery of 0-day vulnerabilities and the development of new attack techniques.

4.5. Course of Action Label

Type Name: `course-of-action-label-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Course of Action

Course of Action Label is an open vocabulary used to label Courses of Action. The labels describe the general type of action that is being represented, such as redirection (for example to a honeynet), internal blocking (for example at the host level), and external blocking (for example at an external firewall). It also includes higher-level courses of action such as policy changes, diplomatic actions, and user training.

Vocabulary Summary	
`perimeter-blocking`, `internal-blocking`, `redirection`, `hardening`, `patching`, `eradication`, `rebuilding`, `training`, `monitoring`, `physical-access-restrictions`, `logical-access-restrictions`, `public-disclosure`, `diplomatic-actions`, `policy-actions`	
Vocabulary Value	**Description**
`perimeter-blocking`	Perimeter-based blocking of traffic from a compromised source.
`internal-blocking`	Host-based blocking of traffic from an internal compromised source.
`redirection`	Re-routing of suspicious or known malicious traffic away from the intended target to an area where the threat can be more safely observed and analyzed.

`hardening`	Securing a system by reducing its surface of unnecessary software, usernames or logins, and running services.
`patching`	A specific form of hardening, patching involves applying a code fix directly to the software with the vulnerability.
`eradication`	Identifying, locating, and eliminating malware from the network.
`rebuilding`	Re-installing a computing resource from a known safe source in order to ensure that the malware is no longer present on the previously compromised resource.
`training`	Training users and administrators on how to identify and mitigate this type of threat.
`monitoring`	Setting up network or host-based sensors to detected the presence of this threat.
`physical-access-restrictions`	Activities associated with restricting physical access to computing resources.
`logical-access-restrictions`	Activities associated with restricting logical access to computing resources.
`public-disclosure`	Informing the public of the existence and characteristics of the threat or threat actor to influence positive change in adversary behavior.
`diplomatic-actions`	Engaging in communications and relationship building with threat actors to influence positive changes in behavior.
`policy-actions`	Modifications to policy that reduce the attack surface or infection vectors of malware.

4.6. Identity Classification

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Threat Actor
- Source
- Victim Target

This vocabulary describes the classification of an identity: whether it describes an organization, group, individual, or class.

Vocabulary Summary	
individual, group, organization, class, unknown	
Vocabulary Value	**Description**
individual	A single person.
group	An informal collection of people, without formal governance, such as a distributed hacker group.
organization	A formal organization of people, with governance, such as a company or country.
class	A class of entities, such as all hospitals or all Europeans.
unknown	It is unknown whether the classification is individual, group, organization, or class.

4.7. Incident Label

Type Name: `incident-label-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Incident

Incident labels is a controlled vocabulary to categorize incidents. Items are not mutually exclusive: an incident can be both a compromise of an asset and a compromise of information.

Vocabulary Summary	
benign, denial-of-service, improper-usage, compromise-asset, compromise-information, insider-breach, malicious-code, probing-scanning, unauthorized-access, investigating	
Vocabulary Value	**Description**

`benign`	An incident that is that is the result of an exercise, testing or a false alarm
`denial-of-service`	An attack that successfully prevents or impairs the normal authorized functionality of networks, systems or applications by exhausting resources. This activity includes being the victim or participating in the DoS
`improper-usage`	A person violates acceptable computing use policies.
`compromise-asset`	An incident that results in the compromise of an asset, such as a host, network device, application or account.
`compromise-information`	An incident that results in the disclosure, corruption or destruction of sensitive information or intellectual property.
`insider-breach`	An incident caused by a threat actor associated with the organization which was the target of the incident.
`malicious-code`	Installation of malicious software (e.g., virus, worm, Trojan horse, or other code-based malicious entity) that infects an operating system or application.
`probing-scanning`	An incident that includes any activity that seeks to access or identify a computer, open ports, protocols, service, or any combination for later exploit. This activity does not directly result in a compromise or denial of service
`unauthorized-access`	An incident in which a Threat Actor gains logical or physical access without permission to a network, system, application, data, or other resource.
`investigating`	Unconfirmed incidents that are potentially malicious or anomalous activity deemed by the reporting entity to warrant further review.

4.8. Indicator Label

Type Name: `indicator-label-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Indicator

Indicator labels is an open vocabulary used to categorize Indicators. It is intended to be high-level to promote consistent practices. Indicator labels should not be used to capture information that can be better captured via related Malware or Attack Pattern objects. It is better to link an Indicator to a Malware object describing Poison Ivy rather than simply labeling it with "poison-ivy".

Vocabulary Summary	
`anomalous-activity`, `anonymization`, `benign`, `compromised`, `malicious-activity`, `attribution`	
Vocabulary Value	**Description**
`anomalous-activity`	An Indicator with this label describes unexpected, or unusual activity that may not necessarily be malicious or indicate compromise. This type of activity may include reconnaissance like behavior such as port scans or version identification, network behaviour anomalies, and asset and/or user behavioural anomalies.
`anonymization`	An Indicator with this label describes suspected anonymization tools or infrastructure (proxy, TOR, VPN, etc.).
`benign`	An Indicator with this label describes activity that is not suspicious or malicious in and of itself, but when combined with other activity may indicate suspicious or malicious behavior.
`compromised`	An Indicator with this label describes assets that are suspected to be compromised.
`malicious-activity`	An Indicator with this label describes patterns of suspected malicious objects and/or activity.
`attribution`	An Indicator with this label describes patterns of behavior that indicate attribution to a particular threat actor or campaign.

4.9. Industry Sector

Type Name: `industry-sector-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Source
- Threat Actor
- Victim Target

Industry sector is an open vocabulary that describes industrial and commercial sectors. It is intended to be holistic: it has been derived from several other lists and is not limited to "critical infrastructure" sectors.

Vocabulary Summary	
`agriculture`, `aerospace`, `automotive`, `communications`, `construction`, `defence`, `education`, `energy`, `engineering`, `entertainment`, `financial-services`, `government-national`, `government-regional`, `government-local`, `government-public-services`, `healthcare`, `hospitality-leisure`, `infrastructure`, `insurance`, `manufacturing`, `mining`, `non-profit`, `pharmaceuticals`, `retail`, `technology`, `telecommunications`, `transportation`, `utilities`	
Vocabulary Value	**Description**
`agriculture`	
`aerospace`	
`automotive`	
`communications`	
`construction`	
`defence`	
`education`	
`energy`	
`engineering`	
`entertainment`	
`financial-services`	
`government-national`	
`government-regional`	

government-local	
government-public-services	
healthcare	
hospitality-leisure	
infrastructure	
insurance	
manufacturing	
mining	
non-profit	
pharmaceuticals	
retail	
technology	
telecommunications	
transportation	
utilities	

4.10. Malware Label

Type Name: `malware-label-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Malware

Malware label is an open vocabulary that represents different types and functions of malware. Malware labels are not mutually exclusive: a malware instance can be both spyware and a screen capture tool.

Vocabulary Summary	

`adware`, `backdoor`, `bot`, `ddos`, `dropper`, `exploit-kit`, `keylogger`, `ransomware`, `remote-access-trojan`, `resource-exploitation`, `rogue-antivirus`, `rootkit`, `screen-capture`, `spyware`, `worm`	
Vocabulary Value	**Description**
`adware`	Any software that is funded by advertising. Adware may also gather sensitive user information from a system.
`backdoor`	A malicious program that allows an attacker to perform actions on a remote system, such as transferring files, acquiring passwords, or executing arbitrary commands [TODO: Ref NIST).
`bot`	A program that resides on an infected system, communicating with and forming part of a botnet. The bot may be implanted by a worm or Trojan, which opens a backdoor. The bot then monitors the backdoor for further instructions.
`ddos`	A tool used to perform a distributed denial of service attack.
`dropper`	A type of trojan that deposits an enclosed payload (generally, other malware) onto the target computer.
`exploit-kit`	A software toolkit to target common vulnerabilities.
`keylogger`	A type of malware that surreptitiously monitors keystrokes and either records them for later retrieval or sends them back to a central collection point.
`ransomware`	A type of malware that encrypts files on a victim's system, demanding payment of ransom in return for the access codes required to unlock files.
`remote-access-trojan`	A remote access Trojan program or RAT, is a Trojan horse capable of controlling a machine through commands issued by a remote attacker.
`resource-exploitation`	A type of malware that steals a system's resources (e.g., CPU cycles), such as a bitcoin miner.
`rogue-antivirus`	A fake security product that demands money to clean phony infections.
`rootkit`	A type of malware that hides its files or processes from normal

	methods of monitoring in order to conceal its presence and activities. Rootkits can operate at a number of levels, from the application level - simply replacing or adjusting the settings of system software to prevent the display of certain information - through hooking certain functions or inserting modules or drivers into the operating system kernel, to the deeper level of firmware or virtualization rook kits, which are activated before the operating system and thus even harder to detect while the system is running.
`screen-capture`	A type of malware used to capture images from the target systems screen, used for exfiltration and command and control.
`spyware`	Software that gathers information on a user's system without their knowledge and sends it to another party. Spyware is generally used to track activities for the purpose of delivering advertising.
`worm`	A self-replicating, self-contained program that usually executes itself without user intervention.

4.11. Pattern Language

Type Name: `pattern-lang-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Indicator

Pattern Language is an open vocabulary that describes the different types of pattern languages that can be used in a STIX Indicator.

Vocabulary Summary	
`cybox`, `snort`, `yara`	
Vocabulary Value	**Description**
`cybox`	CybOX Patterning v1.0 [TODO Ref]
`snort`	Snort pattern (any version)

`yara`	Yara pattern (any version)

4.12. Report Label

Type Name: `report-label-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Report

Report Label is an open vocabulary to describe the primary purpose or subject of a report. For example, a report that contains malware and indicators for that malware should have a report intent of `malware-report` to capture that the malware is the primary purpose. Report labels are not mutually exclusive: a Report can be both a malware report and a tool report.

Vocabulary Summary	
`threat-report`, `attack-pattern-report`, `campaign-report`, `indicator-report`, `malware-report`, `observed-data-report`, `threat-actor-report`, `tool-report`, `victim-target-report`, `vulnerability-report`	
Vocabulary Value	**Description**
`threat-report`	Report subject is a broad characterization of a threat across multiple facets.
`attack-pattern-report`	Report subject is a characterization of one or more attack patterns and related information.
`campaign-report`	Report subject is a characterization of one or more campaigns and related information.
`indicator-report`	Report subject is a characterization of one or more indicators and related information.
`malware-report`	Report subject is a characterization of one or more malware instances and related information.

observed-data-report	Report subject is a characterization of observed data and related information.
threat-actor-report	Report subject is a characterization of one or more threat actors and related information.
tool-report	Report subject is a characterization of one or more tools and related information.
victim-target-report	Report subject is a characterization of one or more victim targets and related information.
vulnerability-report	Report subject is a characterization of one or more vulnerabilities and related information.

4.13. Threat Actor Label

Type Name: `threat-actor-label-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Threat Actor

Threat actor labels is an open vocabulary used to describe types of threat actors. For example, some threat actors are competitors who try to steal information, while others are activists who act in support of a social or political cause. Actor labels are not mutually exclusive: a threat actor can be both a disgruntled employee and a spy.

This section including vocabulary items and their descriptions is based on and contains copied text from the Threat Agent Library publication from Intel Corp in Sept 2007[7]

Vocabulary Summary	
activist, competitor, crime-syndicate, cyber-warrior, employee-accidental, employee-disgruntled, sensationalist, spy, terrorist, thief	
Vocabulary Value	**Description**

[7] Intel Corp Threat Agent Library Sept 2007

`activist`	Highly motivated, potentially destructive supporter of a social or political cause. Activist actions directed towards an organization are often intended to protest and influence the organization's decisions pertaining to issues such as facility placement, trade and business dealings, or labor or environmental impacts. Their attacks are usually intended to either disrupt the ability produce product or services or damage the company's image. The activist may act entirely online, or may extend their operations into the cyber realm in addition to physical attacks. Activists are primarily motivated by ideology, which can drive extensive and persistent attacks. This category includes actors sometimes referred to as anarchists, cyber vandals, extremists, and hacktivists in addition to what are traditionally known as activists. It does not include terrorists, as activist attacks can be severe but generally do not intend the personal injury and loss of life sought by terrorists.
`competitor`	An organization which rivals another in the economic marketplace and competes for the same market share. The goal of a competitor is to gain an advantage in business with respect to the rival organization it targets. It usually does this by copying intellectual property, trade secrets, acquisition strategies, or other technical or business data from a rival organization with the intention of using the data to bolster its own assets and market position. Highly aggressive competitors may also use disruptive or damaging attacks to slow or block a rival's progress. "Competitor" can include vendors and partners, but in this context does not include military adversaries (see the Cyber-Warrior and Spy descriptors). The primary motivation for a competitor taking hostile actions is organizational gain.
`crime-syndicate`	An enterprise organized to conduct significant, large-scale criminal activity for profit. Crime syndicates, also known as organized crime, are generally large, well-resourced groups that operate to create profit from all types of crime. Their activities can

	be seriously harmful and even extreme in impact, and they may use any combination of physical and cyber techniques to both execute attacks and protect their organization. They are almost entirely motivated by organizational gain to create profit, including cases where they have hired out to political or nationalistic interests to attack on their clients' behalf. However, they can also act from dominance in establishing local political or social power or in opposing rival syndicates. As the name implies a crime syndicate is generally a larger, formal organization. Those with similar criminal objectives but working independently or in very small groups generally belong in the Thief category.
`cyber-warrior`	Member of an organization that engages in cyber activities to support active military objectives. Cyber warriors usually work for organizations affiliated with the military forces of a nation state and work at the direction of that state's government and military leadership, but may work for a private organization. A cyberwarrior typically has access to significant support, resources, training, and tools and is capable of designing and executing very sophisticated and effective campaigns. Using these capabilities, the cyberwarrior's role is to support the organization in active conflicts, either physical or political. Their motivation is primarily dominance, but other motivations such as ideology may come into play. As in all military organizations, intelligence gathered through espionage is essential to their conflict success and that espionage is often carried out by the same organization. Although affiliated with the cyberwarrior, the espionage role is properly called "Spy," even though the individual may actually work in a cyber-war unit and may even take on the cyberwarrior role during conflicts. "Cyberwarrior" refers only to individuals engaged in active conflicts, including conflicts of the "cold war" type. In cases where the espionage and cyber-war organization are the same, that relationship should be noted in the [TODO affiliation construct???].
`employee-accidental`	A non-hostile employee who unintentionally exposes the organization to harm.

	“Employee” in this context includes any worker extended internal trust, such as regular employees, contractors, consultants, and temporary workers. Every employee occasionally makes mistakes, sometimes serious ones. Some risk factors that increase the likelihood of a security-relevant mistake include poor or incomplete training, fatigue, overwork, and distraction. For instance, a new hire may not yet have the knowledge to precisely follow confidentiality protocols, or an experienced worker may be distressed about a relative's illness and forget an important step in a sandbox configuration procedure. In any case, the employee is well-intentioned, and the mistakes are unintentional and possibly even unnoticed by the employee.
`employee-disgruntled`	Current or former employee with intent to harm the organization in retaliation for perceived wrongs. “Employee” in this context includes any worker extended internal trust, such as regular employees, contractors, consultants, and temporary workers. When the grievances of a disgruntled employee (real or perceived) is severe and the situation escalates, he or she a can seek revengeful and harmful retaliation. Disgruntled threat actors can include both employees and former employees, who may have extensive knowledge that can be leveraged when conducting attacks. Often a disgruntled employee acts alone but may join an organization, whether group of similar individuals, a competitor, or criminal organization, if the individual believes that doing so will enable greater harm to the source of his or her anger. A disgruntled person can use cyber or physical means to take any number of actions including sabotage, violence, theft, fraud, espionage, or embarrassing individuals or the organization.
`sensationalist`	Seeks to cause embarrassment and brand damage by exposing sensitive information in a manner designed to cause a public relations crisis. A Sensationalist may be an individual or small group of people motivated primarily by a need for notoriety. Unlike the Activist, the Sensationalist generally has no political goal, and is not using bad PR to influence the target to

	change its behavior or business practices. The embarrassment of the target is the end in itself, along with the “15 minutes of fame” that the scandal may bring to the Sensationalists themselves. Any disruption or damage to the target's infrastructure is only important insofar as it adds to negative public perception.
`spy`	Secretly collects the sensitive information of another for use, dissemination, or sale. While in the broad sense spying, i.e., espionage, is a form of theft, it is recognized as special case and is usually treated far more severely than simple thievery. Many spies are part of a well-resourced intelligence organization and are capable of very sophisticated clandestine operations. However, insiders such as employees or consultants can be just as effective and damaging, even when their activities are largely opportunistic and not part of an overall campaign. This includes employees who leak information they believe is evidence of wrongdoing, or opportunistically taking information when they leave the organization. In this context, a Spy is one who collects sensitive information for the benefit of any economic, industrial, or military espionage objective, in other words the domain or end user is not considered in defining the Spy. There can be any number of motivations for spying depending on the individual or organizations involved.
`terrorist`	Uses extreme violence to advance a social or political agenda as well as monetary crimes to support its activities. “Terrorist” does not have a universally accepted definition and usually depends on regional and situational aspects for identification. In this context it refers to individuals who target noncombatants with extreme violence to send a message of fear far beyond the actual events. They may act independently or as part of a terrorist organization. While terrorist violence requires physical action that action can be generated through cyber means, such as by sabotaging critical infrastructure or facility safety systems via cyber manipulation. Terrorist organizations must typically raise much of their operating budget through criminal activity, which is increasingly occurring online. Terrorists are also

	often adept at using and covertly manipulating social media for both recruitment and impact. The primary motivation for terrorist activity, both violent and monetary, is ideology, which can drive extensive and persistent attacks. Dominance, disgruntlement, and organizational gain are often also present as motivators.
`thief`	Individual who steals items of value for personal financial gain. A Thief opportunistically attacks wherever it looks like there is easy profit to be made, whether it be from a large company or from another individual. Many kinds of resources can be stolen especially money or other financial assets such as credit card numbers, but also valuables, hardware, business or personal data, intellectual property, or anything else that can be easily sold. Also considered theft are various avenues of extortion, such as ransomware. Theft can be as simple as pocketing an unattended smartphone, and as sophisticated as hacking into a large organization to steal thousands of identities to sell on the black market. Unlike a Spy, who also steals and sells information but for organizational gain, the Thief's goal is simple personal financial gain. As defined here, “Thief” refers to those acting individually or in very small or informal groups. For sophisticated, organized criminal activity, see the Crime Syndicate descriptor.

4.14. Threat Actor Role

Type Name: `threat-actor-role-ov`

This vocabulary is currently used in the following SDO(s) in the STIX model:

- Threat Actor

Threat actor roles is an open vocabulary that is used to describe the different roles that a threat actor can play. For example, some threat actors author malware or operate botnets while other actors actually carry out attacks directly.

Threat actor roles are not mutually exclusive: an actor can be both a financial backer for attacks and also direct attacks.

Vocabulary Summary	
`agent`, `director`, `financial-backer`, `infrastructure-operator`, `malware-author`, `perpetrator`	
Vocabulary Value	**Description**
`agent`	Threat actor is an independent agent.
`director`	Threat actor directs the activities and goals of attacks.
`financial-backer`	Threat actor funds attacks.
`infrastructure-operator`	Threat actor provides attack infrastructure (botnet providers, etc.)
`malware-author`	Threat actor authors malware or other malicious tools.
`perpetrator`	Threat actor performs actual attacks.

4.15. Tool Label

Type Name: `tool-label-ov`

Tool labels describe the categories of tools that can be used to perform attacks.

Vocabulary Summary	
`denial-of-service`, `exploitation`, `network-capture`, `password-cracking`, `remote-access`, `vulnerability-scanning`	
Vocabulary Value	**Description**
`denial-of-service`	A tool used to perform denial of service attacks or distributed denial of service attacks, such as Low Orbit Ion Cannon (LOIC) and DHCPig.
`exploitation`	A tool used to exploit software and systems, such as sqlmap and metasploit.

`network-capture`	Tools used to capture network traffic, such as Wireshark and Kismet.
`password-cracking`	Tools used to crack password databases, either locally or remotely, such as John the Ripper and NCrack.
`remote-access`	Tools used to access machines remotely, such as VNC and Remote Desktop.
`vulnerability-scanning`	Tools used to scan systems and networks for vulnerabilities, such as NMAP.