TTA Standard

정보통신단체표준(영문표준)

TTAE.OT-xx.xxxx-Part2 제정일: 2018년 xx월 xx일

구조화된 위협 정보 표현 규격(STIX^{TM}) 버전 2.0 - 제2부: STIX 객체

Structured Threat Information eXpression(STIX^{TM}) Version 2.0 - Part2: STIX Objects

한국정보통신기술협회
Telecommunications Technology Association

표준초안 검토 위원회　사이버보안 프로젝트그룹(PG503)

표준안 심의 위원회　정보보호 기술위원회(TC5)

	성명	소 속	직위	위원회 및 직위	표준번호
표준(과제) 제안	김종현	한국전자통신연구원	책임연구원	사이버보안 프로젝트 그룹 위원	미정
	박성민	한국인터넷진흥원	선임연구원	위원	미정
	김낙현	한국인터넷진흥원	선임연구원	위원	미정
	이철호	국가보안기술 연구소	책임연구원	위원	미정
표준 초안 작성자	김종현	한국전자통신연구원	책임연구원	사이버보안 프로젝트 그룹 위원	미정
	박성민	한국인터넷진흥원	선임연구원	위원	미정
	김낙현	한국인터넷진흥원	선임연구원	위원	미정
	이철호	국가보안기술 연구소	책임연구원	위원	미정
	염흥열	순천향대학교	교수	위원	미정
	김익균	한국전자통신연구원	책임연구원	사이버보안 프로젝트 그룹 위원	미정
사무국 담당	박수정	TTA	책임연구원	-	

본 표준 발간 이전에 접수된 지식재산권 확약서 정보는 본 표준의 ‘부록(지식재산권 확약서 정보)’에 명시하고 있으며, 이후 접수된 지식재산권 확약서는 TTA 웹사이트에서 확인할 수 있습니다.
본 표준과 관련하여 접수된 확약서 외의 지식재산권이 존재할 수 있습니다.

발행인 : 한국정보통신기술협회 회장
발행처 : 한국정보통신기술협회
13591, 경기도 성남시 분당구 분당로 47
Tel : 031-724-0114, Fax : 031-724-0109
발행일 : 2018.xx

서 문

1 표준의 목적

이 표준은 STIX 도메인 객체(SDO, STIX Domain Objects) 및 보유 속성(Properties), 그리고 STIX 관계(Relationships)에 대한 정의와 설명을 포함한다. STIX 객체는 사이버 위협 인텔리전스(CTI, Cyber Threat Intelligence)의 일관되고 직관적인 공유를 위한 표준이다.

2 주요 내용 요약

이 표준은 STIX 도메인 객체(SDO, STIX Domain Objects)의 집합을 정의하며, 각 SDO는 일반적으로 CTI에서 널리 사용되는 개념에 해당한다. SDO의 구성요소(공격패턴, 캠페인, 대응방법, 아이덴티티, 침해지표, 침투 집합, 악성코드, 관측 데이터, 리포트, 위협 행위자, 도구, 취약점)와 STIX 관계(Relationships)를 사용하여 객체간의 폭 넓고 다양한 CTI를 생성하고 공유할 수 있다.

3 인용 표준과의 비교

3.1 인용 표준과의 관련성

이 표준은 인용표준(STIX^{TM} Version 2.0. Part2: STIX Objects)을 영문 그대로 완전히 수용하는 표준이다.

3.2 인용 표준과 본 표준의 비교표

TTAE.xx-xx.xxxx	STIX^{TM} Version 2.0. Part2: STIX Objects	비고
1. 소개	1. Introduction	동일
2. STIX 도메인 객체	2. STIX Domain Objects	동일
3. STIX 관계 객체	3. STIX Relationship Objects	동일
4. 적합성	4. Conformance	동일
부속서 A. 용어 사전	Appendix A. Glossary	동일
부속서 B. 감사의 글	Appendix B. Acknowledgements	동일
부속서 C. 개정이력	Appendix C. Revision History	동일

Preface

1 Purpose

The purpose of this standard is to include definitions and descriptions of STIX Domain Objects(SDO), Retention Properties, and STIX Relationships. STIX Objects provides a consistent and intuitive sharing of Cyber Threat Intelligence (CTI).

2 Summary

This standard defines the set of STIX Domain Objects, each of which corresponds to a unique concept commonly represented in CTI. Using SDO(Attack Pattern, Campaign, Course of Action, Identity, Indicator, Intrusion Set, Malware, Observed data, Report, Threat Actor, Tool, Vulnerability) and STIX relationships as building blocks, individuals can create and share broad and comprehensive cyber threat intelligence.

3 Relationship to Reference Standards

3.1 The relationship of international standards

The standard is fully equivalent to STIX^{TM} Version 2.0. Part2: STIX Objects.

3.2 Differences between International standards(recommendation) and this standard

TTAE.xx-xx.xxxx	STIX^{TM} Version 2.0. Part2: STIX Objects	Remarks
1. Introduction	1. Introduction	Equals
2. STIX Domain Objects	2. STIX Domain Objects	Equals
3. STIX Relationship Objects	3. STIX Relationship Objects	Equals
4. Conformance	4. Conformance	Equals
Appendix A. Glossary	Appendix A. Glossary	Equals
Appendix B. Acknowledgements	Appendix B. Acknowledgements	Equals
Appendix C. Revision History	Appendix C. Revision History	Equals

목 차

STIX™ Version 2.0. Part 2: STIX Objects

Committee Specification 01

19 July 2017

Specification URIs

This version:

http://docs.oasis-open.org/cti/stix/v2.0/cs01/part2-stix-objects/stix-v2.0-cs01-part2-stix-objects.docx (Authoritative)

http://docs.oasis-open.org/cti/stix/v2.0/cs01/part2-stix-objects/stix-v2.0-cs01-part2-stix-objects.html

http://docs.oasis-open.org/cti/stix/v2.0/cs01/part2-stix-objects/stix-v2.0-cs01-part2-stix-objects.pdf

Previous version:

http://docs.oasis-open.org/cti/stix/v2.0/csprd02/part2-stix-objects/stix-v2.0-csprd02-part2-stix-objects.docx (Authoritative)

http://docs.oasis-open.org/cti/stix/v2.0/csprd02/part2-stix-objects/stix-v2.0-csprd02-part2-stix-objects.html

http://docs.oasis-open.org/cti/stix/v2.0/csprd02/part2-stix-objects/stix-v2.0-csprd02-part2-stix-objects.pdf

Latest version:

http://docs.oasis-open.org/cti/stix/v2.0/stix-v2.0-part2-stix-objects.docx (Authoritative)

http://docs.oasis-open.org/cti/stix/v2.0/stix-v2.0-part2-stix-objects.html

http://docs.oasis-open.org/cti/stix/v2.0/stix-v2.0-part2-stix-objects.pdf

Technical Committee:

OASIS Cyber Threat Intelligence (CTI) TC

Chair:

Richard Struse (Richard.Struse@HQ.DHS.GOV), DHS Office of Cybersecurity and Communications (CS&C)

Editors:

Rich Piazza (rpiazza@mitre.org), MITRE Corporation

John Wunder (jwunder@mitre.org), MITRE Corporation

Bret Jordan (bret_jordan@symantec.com), Symantec Corp.

Additional artifacts:

This prose specification is one component of a Work Product that also includes:

1. *STIX™ Version 2.0. Part 1: STIX Core Concepts*. http://docs.oasis-open.org/cti/stix/v2.0/cs01/part1-stix-core/stix-v2.0-cs01-part1-stix-core.html.
2. (this document) *STIX™ Version 2.0. Part 2: STIX Objects*. http://docs.oasis-open.org/cti/stix/v2.0/cs01/part2-stix-objects/stix-v2.0-cs01-part2-stix-objects.html.
3. *STIX™ Version 2.0. Part 3: Cyber Observable Core Concepts*. http://docs.oasis-open.org/cti/stix/v2.0/cs01/part3-cyber-observable-core/stix-v2.0-cs01-part3-cyber-observable-core.html.
4. *STIX™ Version 2.0. Part 4: Cyber Observable Objects*. http://docs.oasis-open.org/cti/stix/v2.0/cs01/part4-cyber-observable-objects/stix-v2.0-cs01-part4-cyber-observable-objects.html.
5. *STIX™ Version 2.0. Part 5: STIX Patterning*. http://docs.oasis-open.org/cti/stix/v2.0/cs01/part5-stix-patterning/stix-v2.0-cs01-part5-stix-patterning.html.

Related work:

This specification replaces or supersedes:

6. *STIX™ Version 1.2.1. Part 1: Overview*. Edited by Sean Barnum, Desiree Beck, Aharon Chernin, and Rich Piazza. Latest version: http://docs.oasis-open.org/cti/stix/v1.2.1/stix-v1.2.1-part1-overview.html.

7. *CybOX™ Version 2.1.1. Part 01: Overview.* Edited by Trey Darley, Ivan Kirillov, Rich Piazza, and Desiree Beck. Latest version: http://docs.oasis-open.org/cti/cybox/v2.1.1/cybox-v2.1.1-part01-overview.html.

This specification is related to:

- *TAXII™ Version 2.0.* Edited by John Wunder, Mark Davidson, and Bret Jordan. Latest version: http://docs.oasis-open.org/cti/taxii/v2.0/taxii-v2.0.html.

Abstract:

Structured Threat Information Expression (STIX™) is a language for expressing cyber threat and observable information. This document defines the set of domain objects and relationship objects that STIX uses to represent cyber threat information.

Status:

This document was last revised or approved by the OASIS Cyber Threat Intelligence (CTI) TC on the above date. The level of approval is also listed above. Check the “Latest version” location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=cti#technical.

TC members should send comments on this specification to the TC’s email list. Others should send comments to the TC’s public comment list, after subscribing to it by following the instructions at the “Send A Comment” button on the TC’s web page at https://www.oasis-open.org/committees/cti/.

This Committee Specification is provided under the Non-Assertion Mode of the OASIS IPR Policy, the mode chosen when the Technical Committee was established. For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the TC’s web page (https://www.oasis-open.org/committees/cti/ipr.php).

Note that any machine-readable content (Computer Language Definitions) declared Normative for this Work Product is provided in separate plain text files. In the event of a discrepancy between any such plain text file and display content in the Work Product's prose narrative document(s), the content in the separate plain text file prevails.

Citation format:

When referencing this specification the following citation format should be used:

[STIX-v2.0-Pt2-Objects]

STIX™ Version 2.0. Part 2: STIX Objects. Edited by Rich Piazza, John Wunder, and Bret Jordan. 19 July 2017. OASIS Committee Specification 01. http://docs.oasis-open.org/cti/stix/v2.0/cs01/part2-stix-objects/stix-v2.0-cs01-part2-stix-objects.htmll. Latest version: http://docs.oasis-open.org/cti/stix/v2.0/stix-v2.0-part2-stix-objects.html.

Notices

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The name "OASIS" is a trademark of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see https://www.oasis-open.org/policies-guidelines/trademark for above guidance.

STIX™, CYBOX™, AND TAXII™ (STANDARD OR STANDARDS) AND THEIR COMPONENT PARTS ARE PROVIDED "AS IS" WITHOUT ANY WARRANTY OF ANY KIND, EITHER EXPRESSED, IMPLIED, OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTY THAT THESE STANDARDS OR ANY OF THEIR COMPONENT PARTS WILL CONFORM TO SPECIFICATIONS, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR FREEDOM FROM INFRINGEMENT, ANY WARRANTY THAT THE STANDARDS OR THEIR COMPONENT PARTS WILL BE ERROR FREE, OR ANY WARRANTY THAT THE DOCUMENTATION, IF PROVIDED, WILL CONFORM TO THE STANDARDS OR THEIR COMPONENT PARTS. IN NO EVENT SHALL THE

UNITED STATES GOVERNMENT OR ITS CONTRACTORS OR SUBCONTRACTORS BE LIABLE FOR ANY DAMAGES, INCLUDING, BUT NOT LIMITED TO, DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES, ARISING OUT OF, RESULTING FROM, OR IN ANY WAY CONNECTED WITH THESE STANDARDS OR THEIR COMPONENT PARTS OR ANY PROVIDED DOCUMENTATION, WHETHER OR NOT BASED UPON WARRANTY, CONTRACT, TORT, OR OTHERWISE, WHETHER OR NOT INJURY WAS SUSTAINED BY PERSONS OR PROPERTY OR OTHERWISE, AND WHETHER OR NOT LOSS WAS SUSTAINED FROM, OR AROSE OUT OF THE RESULTS OF, OR USE OF, THE STANDARDS, THEIR COMPONENT PARTS, AND ANY PROVIDED DOCUMENTATION. THE UNITED STATES GOVERNMENT DISCLAIMS ALL WARRANTIES AND LIABILITIES REGARDING THE STANDARDS OR THEIR COMPONENT PARTS ATTRIBUTABLE TO ANY THIRD PARTY, IF PRESENT IN THE STANDARDS OR THEIR COMPONENT PARTS AND DISTRIBUTES IT OR THEM "AS IS."

Table of Contents

1 Introduction

Structured Threat Information Expression (STIX™) is a language and serialization format used to exchange cyber threat intelligence (CTI). STIX enables organizations to share CTI with one another in a consistent and machine readable manner, allowing security communities to better understand what computer-based attacks they are most likely to see and to anticipate and/or respond to those attacks faster and more effectively. STIX is designed to improve many different capabilities, such as collaborative threat analysis, automated threat exchange, automated detection and response, and more.

In response to lessons learned in implementing previous versions, STIX has been significantly redesigned and, as a result, omits some of the objects and properties defined in STIX 1.2.1 (see *STIX™ Version 1.2.1 Part 1: Overview*). The objects chosen for inclusion in STIX 2.0 represent a minimally viable product (MVP) that fulfills basic consumer and producer requirements for CTI sharing. Objects and properties not included in STIX 2.0, but deemed necessary by the community, will be included in future releases.

This document (STIX Objects) uses the concepts introduced in *STIX™ Version 2.0. Part 1: STIX Core Concepts* to define STIX Domain Objects and STIX Relationship Objects.

1.0 IPR Policy

This Committee Specification is provided under the Non-Assertion Mode of the OASIS IPR Policy, the mode chosen when the Technical Committee was established. For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the TC's web page (https://www.oasis-open.org/committees/cti/ipr.php).

1.1 Terminology

The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this document are to be interpreted as described in [RFC2119].

All text is normative except for examples and any text marked non-normative.

1.2 Normative References

[RFC2119] Bradner, S., ""Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, http://www.rfc-editor.org/info/rfc2119

1.3 Non-Normative References

[CAPEC] Common Attack Pattern Enumeration and Classification (CAPEC). (2014, Nov. 7). The MITRE Corporation. [Online]. Available: http://capec.mitre.org.

[CVE] Common Vulnerabilities and Exposures (CVE). The MITRE Corporation. [Online]. Available: http://cve.mitre.org.

1.4 Naming Requirements

1.4.1 Property Names and String Literals

In the JSON serialization all property names and string literals **MUST** be exactly the same, including case, as the names listed in the property tables in this specification. For example, the SDO common property **`created_by_ref`** must result in the JSON key name "created_by_ref". Properties marked required in the property tables **MUST** be present in the JSON serialization.

1.4.2 Reserved Names

Reserved property names are marked with a type called `RESERVED` and a description text of “RESERVED FOR FUTURE USE”. Any property name that is marked as `RESERVED` **MUST NOT** be present in STIX content conforming to this version of the specification.

1.5 Document Conventions

1.5.1 Naming Conventions

All type names, property names and literals are in lowercase, except when referencing canonical names defined in another standard (e.g. literal values from an IANA registry). Words in property names are separated with an underscore (_), while words in type names and string enumerations are separated with a hyphen (-). All type names, property names, object names, and vocabulary terms are between three and 250 characters long.

1.5.2 Font Colors and Style

The following color, font and font style conventions are used in this document:

- The `Consolas` font is used for all type names, property names and literals.
  - type names are in red with a light red background – `threat-actor`
  - property names are in bold style – **`created_at`**
  - literals (values) are in blue with a blue background – `malicious-activity`
  - All relationship types are string literals, therefore they will also appear in blue with a blue background – `related-to`
- In an object's property table, if a common property is being redefined in some way, then the background is dark grey.
- All examples in this document are expressed in JSON. They are in `Consolas` 9-point font, with straight quotes, black text and a `light grey background`, and 2-space indentation.
- Parts of the example may be omitted for conciseness and clarity. These omitted parts are denoted with the ellipses (...).
- The term “hyphen” is used throughout this document to refer to the ASCII hyphen or minus character, which in Unicode is “hyphen-minus”, U+002D.

2 STIX Domain Objects

This specification defines the set of STIX Domain Objects (SDOs), each of which corresponds to a unique concept commonly represented in CTI. Using SDOs and STIX relationships as building blocks, individuals can create and share broad and comprehensive cyber threat intelligence.

Property information, relationship information, and examples are provided for each SDO defined below. 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. Forward relationships (i.e., relationships *from* the SDO to other SDOs) are fully defined, while reverse relationships (i.e., relationships *to* the SDO from other SDOs) are duplicated for convenience.

Some SDOs are similar and can be grouped together into categories. Attack Pattern, Malware, and Tool can all be considered types of tactics, techniques, and procedures (TTPs): they describe behaviors and resources that attackers use to carry out their attacks. Similarly, Campaign, Intrusion Set, and Threat Actor all describe information about why adversaries carry out attacks and how they organize themselves.

2.1 Attack Pattern

Type Name: `attack-pattern`

Attack Patterns are a type of TTP that describe ways that adversaries attempt to compromise targets. Attack Patterns are used to help categorize attacks, generalize specific attacks to the patterns that they follow, and provide detailed information about how attacks are performed. An example of an attack pattern is "spear phishing": a common type of attack where an attacker sends a carefully crafted e-mail message to a party with the intent of getting them to click a link or open an attachment to deliver malware. Attack Patterns can also be more specific; spear phishing as practiced by a particular threat actor (e.g., they might generally say that the target won a contest) can also be an Attack Pattern.

The Attack Pattern SDO contains textual descriptions of the pattern along with references to externally-defined taxonomies of attacks such as CAPEC [CAPEC]. Relationships from Attack Pattern can be used to relate it to what it targets (Vulnerabilities and Identities) and which tools and malware use it (Tool and Malware).

2.1.1 Properties

Common Properties		
`type`, `id`, `created_by_ref`, `created`, `modified`, `revoked`, `labels`, `external_references`, `object_marking_refs`, `granular_markings`		
Attack Pattern Specific Properties		
`name`, `description`, `kill_chain_phases`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of property 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 property **MAY** be used to provide one or more Attack Pattern identifiers, such as a CAPEC ID. When specifying a CAPEC

		ID, the **`source_name`** property of the external reference **MUST** be set to `capec` and the **`external_id`** property **MUST** be formatted as `CAPEC-[id]`.
`name` (required)	`string`	A 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.

2.1.2 Relationships

These are the relationships explicitly defined between the Attack Pattern object and other SDOs. 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 the Attack Pattern object by way of the Relationship object. The reverse relationships (relationships "to" the Attack Pattern 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. Relationship objects can be created between any SDOs using the `related-to` relationship type or, as with open vocabularies, user-defined names.

Embedded Relationships			
`created_by_ref`	`identifier` (of type `identity`)		
`object_marking_refs`	`identifier` (of type `marking-definition`)		
Common Relationships			
`duplicate-of`, `derived-from`, `related-to`			
Source	**Relationship Type**	**Target**	**Description**
`attack-pattern`	`targets`	`identity`, `vulnerability`	This Relationship describes that this Attack Pattern typically targets the type of victims or vulnerability represented by the related Identity or Vulnerability object. For example, a `targets` Relationship linking an Attack Pattern for SQL injection to an Identity object representing domain administrators means that the form of SQL injection characterized by the Attack Pattern targets domain administrators in order to achieve its objectives.

			Another example is a Relationship linking an Attack Pattern for SQL injection to a Vulnerability in blogging software means that the particular SQL injection attack exploits that vulnerability.
`attack-pattern`	`uses`	`malware`, `tool`	This Relationship describes that the related Malware or Tool is used to perform the behavior identified in the Attack Pattern. For example, a `uses` Relationship linking an Attack Pattern for a distributed denial of service (DDoS) to a Tool for Low Orbit Ion Cannon (LOIC) indicates that the tool can be used to perform those DDoS attacks.
Reverse Relationships			
`indicator`	`indicates`	`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.

Examples

A generic attack pattern for spear phishing, referencing CAPEC

```
{
  "type": "attack-pattern",
  "id": "attack-pattern--0c7b5b88-8ff7-4a4d-aa9d-feb398cd0061",
  "created": "2016-05-12T08:17:27.000Z",
  "modified": "2016-05-12T08:17:27.000Z",
  "name": "Spear Phishing",
  "description": "...",
  "external_references": [
  {
  "source_name": "capec",
  "external_id": "CAPEC-163"
  }
  ]
}
```

A specific attack pattern for a particular form of spear phishing, referencing CAPEC

```
[
  {
  "type": "attack-pattern",
  "id": "attack-pattern--7e33a43e-e34b-40ec-89da-36c9bb2cacd5",
  "created": "2016-05-12T08:17:27.000Z",
  "modified": "2016-05-12T08:17:27.000Z",
  "name": "Spear Phishing as Practiced by Adversary X",
  "description": "A particular form of spear phishing where the attacker claims that the target had won a contest, including personal details, to get them to click on a link.",
  "external_references": [
  {
    "source_name": "capec",
    "id": "CAPEC-163"
  }
  ]
  },
  {
  "type": "relationship",
  "id": "relationship--57b56a43-b8b0-4cba-9deb-34e3e1faed9e",
  "created": "2016-05-12T08:17:27.000Z",
  "modified": "2016-05-12T08:17:27.000Z",
  "relationship_type": "uses",
  "source_ref": "intrusion-set--0c7e22ad-b099-4dc3-b0df-2ea3f49ae2e6",
  "target_ref": "attack-pattern--7e33a43e-e34b-40ec-89da-36c9bb2cacd5"
  },
  {
  "type": "intrusion-set",
  "id": "intrusion-set--0c7e22ad-b099-4dc3-b0df-2ea3f49ae2e6",
  "created": "2016-05-12T08:17:27.000Z",
  "modified": "2016-05-12T08:17:27.000Z",
  "name": "Adversary X"
  }
]
```

2.2 Campaign

Type Name: `campaign`

A Campaign is a grouping of adversarial behaviors that describes a set of malicious activities or attacks (sometimes called waves) that occur over a period of time against a specific set of targets. Campaigns usually have well defined objectives and may be part of an Intrusion Set.

Campaigns are often attributed to an intrusion set and threat actors. The threat actors may reuse known infrastructure from the intrusion set or may set up new infrastructure specific for conducting that campaign.

Campaigns can be characterized by their objectives and the incidents they cause, people or resources they target, and the resources (infrastructure, intelligence, Malware, Tools, etc.) they use.

For example, a Campaign could be used to describe a crime syndicate's attack using a specific variant of malware and new C2 servers against the executives of ACME Bank during the summer of 2016 in order to gain secret information about an upcoming merger with another bank.

2.2.1 Properties

Common Properties		
`type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings`		
Campaign Specific Properties		
`name, description, aliases, first_seen, last_seen, objective`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this property **MUST** be `campaign`.
`name` (required)	`string`	A name used to identify the Campaign.
`description` (optional)	`string`	A description that provides more details and context about the Campaign, 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 this Campaign was first seen. This property is a summary property of data from sightings and other data that may or may not be available in STIX. If new sightings are received that are earlier than the first seen timestamp, the object may be updated to account for the new data.
`last_seen` (optional)	`timestamp`	The time that this Campaign was last seen. This property is a summary property of data from sightings and other data that may or may not be available in STIX. If new sightings are received that are later than the last seen timestamp, the object may be updated to account for the new data.
`objective` (optional)	`string`	This property defines the Campaign’s primary goal, objective, desired outcome, or intended effect — what the Threat Actor hopes to accomplish with this Campaign.

2.2.2 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 the Campaign object by way of the Relationship object. The reverse relationships (relationships "to" the Campaign 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 type or, as with open vocabularies, user-defined names.

Embedded Relationships			
`created_by_ref`		`identifier` (of type `identity`)	
`object_marking_refs`		`identifier` (of type `marking-definition`)	
Common Relationships			
`duplicate-of`, `derived-from`, `related-to`			
Source	**Relationship Type**	**Target**	**Description**
`campaign`	`attributed-to`	`intrusion-set`, `threat-actor`	This Relationship describes that the Intrusion Set or Threat Actor that is involved in carrying out the Campaign. For example, an `attributed-to` Relationship from the Glass Gazelle Campaign to the Urban Fowl Threat Actor means that the actor carried out or was involved in some of the activity described by the Campaign.
`campaign`	`targets`	`identity`, `vulnerability`	This Relationship describes that the Campaign uses exploits of the related Vulnerability or targets the type of victims described by the related Identity. For example, a `targets` Relationship from the Glass Gazelle Campaign to a Vulnerability in a blogging platform indicates that attacks performed as part of Glass Gazelle often exploit that Vulnerability. Similarly, a `targets` Relationship from the Glass Gazelle Campaign to a Identity describing the energy sector in the United States means that the Campaign typically carries out attacks against targets in that sector.
`campaign`	`uses`	`attack-pattern`, `malware`, `tool`	This Relationship describes that attacks carried out as part of the Campaign typically use the related Attack Pattern, Malware, or Tool. For example, a `uses` Relationship from the Glass Gazelle Campaign to the xInject Malware indicates that xInject is often used during attacks attributed to that Campaign.

Reverse Relationships			
`indicator`	`indicates`	`campaign`	See forward relationship for definition.

Examples

```
{
  "type": "campaign",
  "id": "campaign--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:03:00.000Z",
  "modified": "2016-04-06T20:03:00.000Z",
  "name": "Green Group Attacks Against Finance",
  "description": "Campaign by Green Group against a series of targets in the financial services sector."
}
```

2.3 Course of Action

Type Name: `course-of-action`

Note: The Course of Action object in STIX 2.0 is a stub. It is included to support basic use cases (such as sharing prose courses of action) but does not support the ability to represent automated courses of action or contain properties to represent metadata about courses of action. Future STIX 2 releases will expand it to include these capabilities.

A Course of Action is an action taken either to prevent an attack or to respond to an attack that is in progress. It may describe technical, automatable responses (applying patches, reconfiguring firewalls) but can also describe higher level actions like employee training or policy changes. For example, a course of action to mitigate a vulnerability could describe applying the patch that fixes it.

The Course of Action SDO contains a textual description of the action; a reserved **`action`** property also serves as placeholder for future inclusion of machine automatable courses of action. Relationships from the Course of Action can be used to link it to the Vulnerabilities or behaviors (Tool, Malware, Attack Pattern) that it mitigates.

2.3.1 Properties

Common Properties		
`type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings`		
Course of Action Specific Properties		
`name, description, action`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this property **MUST** be `course-of-action`.
`name` (required)	`string`	A name used to identify the Course of Action.

description (optional)	string	A description that provides more details and context about the Course of Action, potentially including its purpose and its key characteristics.
action (reserved)	RESERVED	RESERVED – To capture structured/automated courses of action.

2.3.2 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 the Course of Action object by way of the Relationship object. The reverse relationships (relationships "to" the Course of Action 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 type or, as with open vocabularies, user-defined names.

Embedded Relationships			
created_by_ref		identifier (of type identity)	
object_marking_refs		identifier (of type marking-definition)	
Common Relationships			
duplicate-of, derived-from, related-to			
Source	**Relationship Type**	**Target**	**Description**
course-of-action	mitigates	attack-pattern, malware, tool, vulnerability	This Relationship describes that the Course of Action can mitigate the related Attack Pattern, Malware, Vulnerability, or Tool. For example, a mitigates Relationship from a Course of Action object to a Malware object indicates that the course of action mitigates the impact of that malware.
Reverse Relationships			
—	—	—	—

Examples

```
[
  {
  "type": "course-of-action",
  "id": "course-of-action--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:03:48.000Z",
  "modified": "2016-04-06T20:03:48.000Z",
  "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 to the existing UDP 1434 filter ..."
  },
  {
  "type": "relationship",
  "id": "relationship--44298a74-ba52-4f0c-87a3-1824e67d7fad",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:07:10.000Z",
  "modified": "2016-04-06T20:07:10.000Z",
  "relationship_type": "mitigates",
  "source_ref": "course-of-action--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "target_ref": "malware--31b940d4-6f7f-459a-80ea-9c1f17b5891b"
  },
  {
  "type": "malware",
  "id": "malware--31b940d4-6f7f-459a-80ea-9c1f17b5891b",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:07:09.000Z",
  "modified": "2016-04-06T20:07:09.000Z",
  "name": "Poison Ivy"
  }
]
```

2.4 Identity

Type Name: `identity`

Identities can represent actual individuals, organizations, or groups (e.g., ACME, Inc.) as well as classes of individuals, organizations, or groups (e.g., the finance sector).

The Identity SDO can capture basic identifying information, contact information, and the sectors that the Identity belongs to. Identity is used in STIX to represent, among other things, targets of attacks, information sources, object creators, and threat actor identities.

2.4.1 Properties

Common Properties
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings

Identity Specific Properties		
name, description, identity_class, sectors, contact_information		
Property Name	**Type**	**Description**
type (required)	string	The value of this property **MUST** be identity.
labels (optional)	list of type string	The list of roles that this Identity performs (e.g., CEO, Domain Administrators, Doctors, Hospital, or Retailer). No open vocabulary is yet defined for this property.
name (required)	string	The name of this Identity. When referring to a specific entity (e.g., an individual or organization), this property **SHOULD** contain the canonical name of the specific entity.
description (optional)	string	A description that provides more details and context about the Identity, potentially including its purpose and its key characteristics.
identity_class (required)	open-vocab	The type of entity that this Identity describes, e.g., an individual or organization. This is an open vocabulary and the values **SHOULD** come from the identity-class-ov vocabulary.
sectors (optional)	list of type open-vocab	The list of industry sectors that this Identity belongs to. 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 Identity. No format for this information is currently defined by this specification.

2.4.2 Relationships

There is an embedded relationship to Identity in all STIX Objects called **created_by_ref** that is inherited from the Common Properties. This property links each object with the Identity of the organization or individual that created the object.

These are the relationships explicitly defined between the Identity 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 the Identity object by way of the Relationship object. None are defined for the Identity object. The reverse relationships (relationships "to" the Identity 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 type or, as with open vocabularies, user-defined names.

Embedded Relationships			
created_by_ref		identifier (of type identity)	
object_marking_refs		identifier (of type marking-definition)	
Common Relationships			
duplicate-of, derived-from, related-to			
Source	**Relationship Type**	**Target**	**Description**
—	—	—	—
Reverse Relationships			
attack-pattern, campaign, intrusion-set, malware, threat-actor, tool	targets	identity	See forward relationship for definition.
threat-actor	attributed-to, impersonates	identity	See forward relationship for definition.

Examples

An Identity for an individual named John Smith

```
{
  "type": "identity",
  "id": "identity--023d105b-752e-4e3c-941c-7d3f3cb15e9e",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:03:00.000Z",
  "modified": "2016-04-06T20:03:00.000Z",
  "name": "John Smith",
  "identity_class": "individual"
}
```

An Identity for a company named ACME Widget, Inc.

```
{
  "type": "identity",
  "id": "identity--e5f1b90a-d9b6-40ab-81a9-8a29df4b6b65",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:03:00.000Z",
  "modified": "2016-04-06T20:03:00.000Z",
  "name": "ACME Widget, Inc.",
  "identity_class": "organization"
}
```

2.5 Indicator

Type Name: `indicator`

Indicators contain a pattern that can be used to detect suspicious or malicious cyber activity. For example, an Indicator may be used to represent a set of malicious domains and use the STIX Patterning Language (*STIX™ Version 2.0. Part 5: STIX Patterning*) to specify these domains.

The Indicator SDO contains a simple textual description, the Kill Chain Phases that it detects behavior in, a time window for when the Indicator is valid or useful, and a required **`pattern`** property to capture a structured detection pattern. Conforming STIX implementations **MUST** support the STIX Patterning Language as defined in *STIX™ Version 2.0. Part 5: STIX Patterning*. While each structured pattern language has different syntax and potentially different semantics, in general an Indicator is considered to have "matched" (or been "sighted") when the conditions specified in the structured pattern are satisfied in whatever context they are evaluated in.

Relationships from the Indicator can describe the malicious or suspicious behavior that it directly detects (Malware, Tool, and Attack Pattern) as well as the Campaigns, Intrusion Sets, and Threat Actors that it might indicate the presence of.

2.5.1 Properties

Common Properties
`type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings`
Indicator Specific Properties
`name, description, pattern, valid_from, valid_until, kill_chain_phases`

Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this property **MUST** be `indicator`.
`labels` (required)	`list` of type `open-vocab`	This property 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`	A name used to identify the Indicator.
`description` (optional)	`string`	A description that provides more details and context about the Indicator, potentially including its purpose and its key characteristics.
`pattern` (required)	`string`	The detection pattern for this

		Indicator is a STIX Pattern as specified in *STIX™ Version 2.0. Part 5: STIX Patterning*.
valid_from (required)	timestamp	The time from which this Indicator should be considered valuable intelligence.
valid_until (optional)	timestamp	The time at which this Indicator should no longer be considered valuable intelligence. If the **valid_until** property is omitted, then there is no constraint on the latest time for which the Indicator should be used.
kill_chain_phases (optional)	list of type kill-chain-phase	The kill chain phase(s) to which this Indicator corresponds.

2.5.2 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 the Indicator object by way of the Relationship object. The reverse relationships (relationships "to" the Indicator 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 type or, as with open vocabularies, user-defined names.

Embedded Relationships			
created_by_ref	identifier (of type identity)		
object_marking_refs	identifier (of type marking-definition)		
Common Relationships			
duplicate-of, derived-from, related-to			
Source	**Relationship Type**	**Target**	**Description**
indicator	indicates	attack-pattern, campaign, intrusion-set, malware, threat-actor, tool	This Relationship describes that the Indicator can detect evidence of the related Campaign, Intrusion Set, or Threat Actor. This evidence may not be direct: for example, the Indicator may detect secondary evidence of the Campaign, such as malware or behavior commonly used by that Campaign. For example, an indicates Relationship from an Indicator to a Campaign object representing Glass Gazelle means that the Indicator is capable of detecting evidence of Glass Gazelle, such as command and

			control IPs commonly used by that Campaign.
Reverse Relationships			
—	—	—	—

Examples

Indicator itself, with context

```
[
  {
  "type": "indicator",
  "id": "indicator--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:03:48.000Z",
  "modified": "2016-04-06T20:03:48.000Z",
  "labels": ["malicious-activity"],
  "name": "Poison Ivy Malware",
  "description": "This file is part of Poison Ivy",
  "pattern": "[ file:hashes.'SHA-256' =
'4bac27393bdd9777ce02453256c5577cd02275510b2227f473d03f533924f877' ]",
  "valid_from": "2016-01-01T00:00:00Z"
  },
  {
  "type": "relationship",
  "id": "relationship--44298a74-ba52-4f0c-87a3-1824e67d7fad",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:06:37.000Z",
  "modified": "2016-04-06T20:06:37.000Z",
  "relationship_type": "indicates",
  "source_ref": "indicator--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "target_ref": "malware--31b940d4-6f7f-459a-80ea-9c1f17b5891b"
  },
  {
  "type": "malware",
  "id": "malware--31b940d4-6f7f-459a-80ea-9c1f17b5891b",
  "created": "2016-04-06T20:07:09.000Z",
  "modified": "2016-04-06T20:07:09.000Z",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "name": "Poison Ivy"
  }
]
```

2.6 Intrusion Set

Type Name: `intrusion-set`

An Intrusion Set is a grouped set of adversarial behaviors and resources with common properties that is believed to be orchestrated by a single organization. An Intrusion Set may capture multiple Campaigns or other activities that are all tied together by shared attributes indicating a common known or unknown Threat Actor. New activity can be attributed to an Intrusion Set even if the Threat Actors behind the attack are not known. Threat Actors can move from supporting one Intrusion Set to supporting another, or they may support multiple Intrusion Sets.

Where a Campaign is a set of attacks over a period of time against a specific set of targets to achieve some objective, an Intrusion Set is the entire attack package and may be used over a very long period of time in multiple Campaigns to achieve potentially multiple purposes.

While sometimes an Intrusion Set is not active, 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 and may be able to only attribute it back to a nation state or perhaps back to an organization within that nation state.

2.6.1 Properties

Common Properties		
`type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings`		
Intrusion Set Specific Properties		
`name, description, aliases, first_seen, last_seen, goals, resource_level, primary_motivation, secondary_motivations`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this property **MUST** be `intrusion-set`.
`name` (required)	`string`	A name used to identify this Intrusion Set.
`description` (optional)	`string`	A description that provides more details and context about the Intrusion Set, 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. This property is a summary property of data from sightings and other data that may or may not be available in STIX. If new sightings are received that are earlier than the first seen timestamp, the object may be updated to account for the new data.

last_seen (optional)	timestamp	The time that this Intrusion Set was last seen. This property is a summary property of data from sightings and other data that may or may not be available in STIX. If new sightings are received that are later than the last seen timestamp, the object may be updated to account for the new data.
goals (optional)	list of type string	The high level goals of this Intrusion Set, namely, *what* are they trying to do. For example, they may be motivated by personal gain, but their goal is to steal credit card numbers. To do this, they may execute specific Campaigns that have detailed objectives like compromising point of sale systems at a large retailer. Another example: to gain information about latest merger and IPO information from ACME Bank.
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 motivation is *why* the Intrusion Set wishes to achieve the goal (what they are trying to achieve). For example, an Intrusion Set with a goal to disrupt the finance sector in a country might be motivated by ideological hatred of capitalism. 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. These motivations can exist as an equal or near-equal cause to the primary motivation. However, it does not replace or necessarily magnify the primary motivation, but it might indicate additional context. This is an open vocabulary and values **SHOULD** come from the attack-motivation-ov vocabulary.

2.6.2 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 the Intrusion Set object by way of the Relationship object. The reverse relationships (relationships "to" the Intrusion Set 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 type or, as with open vocabularies, user-defined names.

Embedded Relationships	
`created_by_ref`	`identifier` (of type `identity`)
`object_marking_refs`	`identifier` (of type `marking-definition`)

Common Relationships
`duplicate-of`, `derived-from`, `related-to`

Source	Relationship Type	Target	Description
`intrusion-set`	`attributed-to`	`threat-actor`	This Relationship describes that the related Threat Actor is involved in carrying out the Intrusion Set. For example, an `attributed-to` Relationship from the Red Orca Intrusion Set to the Urban Fowl Threat Actor means that the actor carried out or was involved in some of the activity described by the Intrusion Set.
`intrusion-set`	`targets`	`identity`, `vulnerability`	This Relationship describes that the Intrusion Set uses exploits of the related Vulnerability or targets the type of victims described by the related Identity. For example, a `targets` Relationship from the Red Orca Intrusion Set to a Vulnerability in a blogging platform indicates that attacks performed as part of Red Orca often exploit that Vulnerability. Similarly, a `targets` Relationship from the Red Orca Intrusion Set to an Identity describing the energy sector in the United States means that the Intrusion Set typically carries out attacks against targets in that sector.
`intrusion-set`	`uses`	`attack-pattern`, `malware`, `tool`	This Relationship describes that attacks carried out as part of the Intrusion Set typically use the related Attack Pattern, Malware, or Tool. For example, a `uses` Relationship

			from the Red Orca Intrusion Set to the xInject Malware indicates that xInject is often used during attacks attributed to that Intrusion Set.
Reverse Relationships			
`campaign`	`attributed-to`	`intrusion-set`	See forward relationship for definition.
`indicator`	`indicates`	`intrusion-set`	See forward relationship for definition.

Examples

```
{
  "type": "intrusion-set",
  "id": "intrusion-set--4e78f46f-a023-4e5f-bc24-71b3ca22ec29",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:03:48.000Z",
  "modified": "2016-04-06T20:03:48.000Z",
  "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"],
  "goals": ["acquisition-theft", "harassment", "damage"]
}
```

2.7 Malware

Type Name: `malware`

Note: The Malware object in STIX 2.0 is a stub. It is included to support basic use cases but is likely not useful for actual malware analysis or for including even simple malware instance data. Future versions of STIX 2 will expand it to include these capabilities.

Malware is a type of TTP that is also known as malicious code and malicious software, and 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 are usually designed to perform these nefarious functions in such a way that users are unaware of them, at least initially.[1]

The Malware SDO characterizes, identifies, and categorizes malware samples and families via a text **`description`** property. This provides detailed information about how the malware works and what it does. Relationships from Malware can capture what the malware targets (Vulnerability and Identity) and link it to another Malware SDO that it is a variant of.

2.7.1 Properties

Common Properties
`type, id, created_by_ref, created, modified, revoked, labels, external_references,`

[1] NIST SP 800-83. http://csrc.nist.gov/publications/nistpubs/800-83/SP800-83.pdf.

`object_marking_refs, granular_markings`		
Malware Specific Properties		
`name, description, kill_chain_phases`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this property **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-label-ov` vocabulary.
`name` (required)	`string`	A name used to identify the Malware sample.
`description` (optional)	`string`	A description that provides more details and context about the Malware, 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 can be used.

2.7.2 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 the Malware object by way of the Relationship object. The reverse relationships (relationships "to" the Malware 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 type or, as with open vocabularies, user-defined names.

Embedded Relationships			
`created_by_ref`		`identifier` (of type `identity`)	
`object_marking_refs`		`identifier` (of type `marking-definition`)	
Common Relationships			
`duplicate-of`, `derived-from`, `related-to`			
Source	**Relationship Type**	**Target**	**Description**
`malware`	`targets`	`identity`, `vulnerability`	This Relationship documents that this Malware is being used to target this Identity or exploit the Vulnerability.

			For example, a targets Relationship linking a Malware representing a downloader to a Vulnerability for CVE-2016-0001 means that the malware exploits that vulnerability. Similarly, a targets Relationship linking a Malware representing a downloader to an Identity representing the energy sector means that downloader is typically used against targets in the energy sector.
malware	uses	tool	This Relationship documents that this Malware uses the related tool to perform its functions.
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			
indicator	indicates	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.

Examples

```
{
  "type": "malware",
  "id": "malware--0c7b5b88-8ff7-4a4d-aa9d-feb398cd0061",
  "created": "2016-05-12T08:17:27.000Z",
  "modified": "2016-05-12T08:17:27.000Z",
  "name": "Cryptolocker",
  "description": "...",
  "labels": ["ransomware"]
}
```

2.8 Observed Data

Type Name: observed-data

Observed Data conveys information that was observed on systems and networks using the Cyber Observable specification defined in parts 3 and 4 of this specification. For example, Observed Data can capture the observation of an IP address, a network connection, a file, or a registry key. Observed Data is not an intelligence assertion, it is simply information: this file was seen, without any context for what it means.

Observed Data captures both a single observation of a single entity (file, network connection) as well as the aggregation of multiple observations of an entity. When the **number_observed** property is 1 the Observed Data is of a single entity. When the **number_observed** property is greater than 1, the observed data consists of several instances of an entity collected over the time window specified by the **first_observed** and **last_observed** properties. When used to collect aggregate data, it is likely that some fields in the Cyber Observable Object (e.g., timestamp fields) will be omitted because they would differ for each of the individual observations.

Observed Data may be used by itself (without relationships) to convey raw data collected from network and host-based detection tools. A firewall could emit a single Observed Data instance containing a single Network Traffic object for each connection it sees. The firewall could also aggregate data and instead send out an Observed Data instance every ten minutes with an IP address and an appropriate **number_observed** value to indicate the number of times that IP address was observed in that window.

Observed Data may also be related to other SDOs to represent raw data that is relevant to those objects. The Sighting object, which captures the sighting of an Indicator, Malware, or other SDO, uses Observed Data to represent the raw information that led to the creation of the Sighting (e.g., what was actually seen that suggested that a particular instance of malware was active).

2.8.1 Properties

Common Properties		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
Observed Data Specific Properties		
first_observed, last_observed, number_observed, objects		
Property Name	**Type**	**Description**
type (required)	string	The value of this property **MUST** be observed-data.
first_observed (required)	timestamp	The beginning of the time window during which the data was observed.
last_observed (required)	timestamp	The end of the time window during which the data was observed.
number_observed (required)	integer	The number of times the data represented in the **objects** property was observed. This **MUST** be an integer between 1 and 999,999,999 inclusive. If the **number_observed** property is greater than 1, the data contained in the **objects** property was observed multiple times. In these cases, object creators **MAY** omit properties of the Cyber Observable object (such as timestamps) that are specific to a single instance of that observed data.

`objects` (required)	`observable-objects`	A dictionary of Cyber Observable Objects representing the observation. The dictionary **MUST** contain at least one object. The `observable-objects` type is defined in *STIX™ Version 2.0. Part 3: Cyber Observable Core Concepts*. The Cyber Observable content **MAY** include multiple objects if those objects are related as part of a single observation. Multiple objects not related to each other via Cyber Observable Relationships **MUST NOT** be contained within the same Observed Data instance. For example, a Network Traffic object and two IPv4 Address objects related via the **`src_ref`** and **`dst_ref`** properties can be contained in the same Observed Data because they are all related and used to characterize that single entity. Two unrelated IPv4 address objects that just happened to be observed at the same time, however, must be represented in separate Observed Data instances.

2.8.2 Relationships

There 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.

In addition to the relationships created using the generic Relationship object, Observed Data is also a direct target of the Sighting SRO. Sightings represent a relationship between some intelligence entity that was seen (e.g., an Indicator or Malware instance), where it was seen, and what evidence was actually seen. The evidence (or raw data) in that relationship is captured as Observed Data.

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

<table>
<tr><th colspan="4">Embedded Relationships</th></tr>
<tr><td colspan="2"><b><code>created_by_ref</code></b></td><td colspan="2"><code>identifier</code> (of type <code>identity</code>)</td></tr>
<tr><td colspan="2"><b><code>object_marking_refs</code></b></td><td colspan="2"><code>identifier</code> (of type <code>marking-definition</code>)</td></tr>
<tr><th colspan="4">Common Relationships</th></tr>
<tr><td colspan="4"><code>duplicate-of</code>, <code>derived-from</code>, <code>related-to</code></td></tr>
<tr><th>Source</th><th>Name</th><th>Target</th><th>Description</th></tr>
<tr><td>—</td><td>—</td><td>—</td><td>—</td></tr>
</table>

Examples

Observed Data of a File object

```
{
  "type": "observed-data",
  "id": "observed-data--b67d30ff-02ac-498a-92f9-32f845f448cf",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T19:58:16.000Z",
  "modified": "2016-04-06T19:58:16.000Z",
  "first_observed": "2015-12-21T19:00:00Z",
  "last_observed": "2015-12-21T19:00:00Z",
  "number_observed": 50,
  "objects": {
  "0": {
  "type": "file",
  ...
  }
  }
}
```

2.9 Report

Type Name: `report`

Reports are collections of threat intelligence focused on one or more topics, such as a description of a threat actor, malware, or attack technique, including context and related details. They are used to group related threat intelligence together so that it can be published as a comprehensive cyber threat story.

The Report SDO contains a list of references to SDOs and SROs (the CTI objects included in the report) along with a textual description and the name of the report.

For example, a threat report produced by ACME Defense Corp. discussing the Glass Gazelle campaign should be represented using Report. The Report itself would contain the narrative of the report while the Campaign SDO and any related SDOs (e.g., Indicators for the Campaign, Malware it uses, and the associated Relationships) would be referenced in the report contents.

2.9.1 Properties

<table>
<tr><th colspan="3">Common Properties</th></tr>
<tr><td colspan="3"><b>type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings</b></td></tr>
<tr><th colspan="3">Report Specific Properties</th></tr>
<tr><td colspan="3"><b>name, description, published, object_refs</b></td></tr>
<tr><th>Property Name</th><th>Type</th><th>Description</th></tr>
</table>

type (required)	string	The value of this property **MUST** be report.
labels (required)	list of type open-vocab	This property 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	A name used to identify the Report.
description (optional)	string	A description that provides more details and context about the Report, 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. The publication date (public release, legal release, etc.) may be different than the date the report was created or shared internally (the date in the **created** property).
object_refs (required)	list of type identifier	Specifies the STIX Objects that are referred to by this Report.

2.9.2 Relationships

There 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	identifier (of type identity)
object_marking_refs	identifier (of type marking-definition)
object_refs	list of type identifier (of STIX Object or marking-definition type)
Common Relationships	
duplicate-of, derived-from, related-to	

Examples

A standalone Report; the consumer may or may not already have access to the referenced STIX

Objects.

```
{
  "type": "report",
  "id": "report--84e4d88f-44ea-4bcd-bbf3-b2c1c320bcb3",
  "created_by_ref": "identity--a463ffb3-1bd9-4d94-b02d-74e4f1658283",
  "created": "2015-12-21T19:59:11.000Z",
  "modified": "2015-12-21T19:59:11.000Z",
  "name": "The Black Vine Cyberespionage Group",
  "description": "A simple report with an indicator and campaign",
  "published": "2016-01-20T17:00:00.000Z",
  "labels": ["campaign"],
  "object_refs": [
  "indicator--26ffb872-1dd9-446e-b6f5-d58527e5b5d2",
  "campaign--83422c77-904c-4dc1-aff5-5c38f3a2c55c",
  "relationship--f82356ae-fe6c-437c-9c24-6b64314ae68a"
  ]
}
```

A Bundle with a Report and the STIX Objects that are referred to by the Report

```
{
  "type": "bundle",
  "id": "bundle--44af6c39-c09b-49c5-9de2-394224b04982",
  "objects": [
  {
  "type": "identity",
  "id": "identity--a463ffb3-1bd9-4d94-b02d-74e4f1658283",
  ...,
  "name": "Acme Cybersecurity Solutions"
  },
  {
  "type": "report",
  "id": "report--84e4d88f-44ea-4bcd-bbf3-b2c1c320bcbd",
  "created_by_ref": "identity--a463ffb3-1bd9-4d94-b02d-74e4f1658283",
  "created": "2015-12-21T19:59:11.000Z",
  "modified": "2016-05-21T19:59:11.000Z",
  "name": "The Black Vine Cyberespionage Group",
  "description": "A simple report with an indicator and campaign",
  "published": "2016-01-201T17:00:00Z",
  "labels": ["campaign"],
  "object_refs": [
    "indicator--26ffb872-1dd9-446e-b6f5-d58527e5b5d2",
    "campaign--83422c77-904c-4dc1-aff5-5c38f3a2c55c",
    "relationship--f82356ae-fe6c-437c-9c24-6b64314ae68a"
  ]
  },
  {
  "type": "indicator",
  "id": "indicator--26ffb872-1dd9-446e-b6f5-d58527e5b5d2",
  "created": "2015-12-21T19:59:17.000Z",
  "modified": "2016-05-21T19:59:17.000Z",
  "name": "Some indicator",
  "labels": ["malicious-activity"],
  "pattern": "[ file:hashes.MD5 = '3773a88f65a5e780c8dff9cdc3a056f3' ]",
  "valid_from": "2015-12-21T19:59:17Z",
  "created_by_ref": "identity--a463ffb3-1bd9-4d94-b02d-74e4f1658283"
  },
  {
  "type": "campaign",
  "id": "campaign--83422c77-904c-4dc1-aff5-5c38f3a2c55c",
  "created_by_ref": "identity--a463ffb3-1bd9-4d94-b02d-74e4f1658283",
  "created": "2015-12-21T19:59:17.000Z",
  "modified": "2016-05-21T19:59:17.000Z",
  "name": "Some Campaign"
  },
  {
  "type": "relationship",
  "id": "relationship--f82356ae-fe6c-437c-9c24-6b64314ae68a",
  "created_by_ref": "identity--a463ffb3-1bd9-4d94-b02d-74e4f1658283",
  "created": "2015-12-21T19:59:17.000Z",
  "modified": "2015-12-21T19:59:17.000Z",
  "source_ref": "indicator--26ffb872-1dd9-446e-b6f5-d58527e5b5d2",
  "target_ref": "campaign--26ffb872-1dd9-446e-b6f5-d58527e5b5d2",
  "relationship_type": "indicates"
  }
  ]
}
```

2.10 Threat Actor

Type Name: `threat-actor`

Threat Actors are actual individuals, groups, or organizations believed to be operating with malicious intent. A Threat Actor is not an Intrusion Set but may support or be affiliated with various Intrusion Sets, groups, or organizations over time.

Threat Actors leverage their resources, and possibly the resources of an Intrusion Set, to conduct attacks and run Campaigns against targets.

Threat Actors can be characterized by their motives, capabilities, goals, sophistication level, past activities, resources they have access to, and their role in the organization.

2.10.1 Properties

<table>
<tr><th colspan="3">Common Properties</th></tr>
<tr><td colspan="3"><b>type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings</b></td></tr>
<tr><th colspan="3">Threat Actor Specific Properties</th></tr>
<tr><td colspan="3"><b>name, description, aliases, roles, goals, sophistication, resource_level, primary_motivation, secondary_motivations, personal_motivations</b></td></tr>
<tr><th>Property Name</th><th>Type</th><th>Description</th></tr>
<tr><td><b>type</b> (required)</td><td>string</td><td>The value of this property <b>MUST</b> be
threat-actor.</td></tr>
<tr><td><b>labels</b> (required)</td><td>list of type open-vocab</td><td>This property specifies the type of Threat Actor.
This is an open vocabulary and values <b>SHOULD</b> come from the threat-actor-label-ov vocabulary.</td></tr>
<tr><td><b>name</b> (required)</td><td>string</td><td>A name used to identify this Threat Actor or Threat Actor group.</td></tr>
<tr><td><b>description</b> (optional)</td><td>string</td><td>A description that provides more details and context about the Threat Actor, potentially including its purpose and its key characteristics.</td></tr>
<tr><td><b>aliases</b> (optional)</td><td>list of type string</td><td>A list of other names that this Threat Actor is believed to use.</td></tr>
<tr><td><b>roles</b> (optional)</td><td>list of type open-vocab</td><td>A list of roles the Threat Actor plays.
This is an open vocabulary and the values <b>SHOULD</b> come from the threat-actor-role-ov vocabulary.</td></tr>
<tr><td><b>goals</b> (optional)</td><td>list of type string</td><td>The high level goals of this Threat Actor, namely, <i>what</i> are they trying to do. For example, they may be motivated by personal gain, but their goal is to steal credit card numbers. To do this, they may execute specific Campaigns that have detailed objectives like compromising point of sale systems at a large retailer.</td></tr>
<tr><td><b>sophistication</b> (optional)</td><td>open-vocab</td><td>The skill, specific knowledge, special training,</td></tr>
</table>

		or expertise a Threat Actor must have to perform the attack. This is an open vocabulary and values **SHOULD** come from the `threat-actor-sophistication-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`** property — 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 motivation is *why* the Threat Actor wishes to achieve the goal (what they are trying to achieve). For example, a Threat Actor with a goal to disrupt the finance sector in a country might be motivated by ideological hatred of capitalism. 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. These motivations can exist as an equal or near-equal cause to the primary motivation. However, it does not replace or necessarily magnify the primary motivation, but it might indicate additional context. 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 goals. For example, an individual analyst may join a Data Miner corporation because his or her skills may 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.

2.10.2 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 the Threat Actor object by way of the Relationship object. The reverse relationships (relationships "to" the Threat Actor 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 type or, as with open vocabularies, user-defined names.

Embedded Relationships			
`created_by_ref`	`identifier` (of type `identity`)		
`object_marking_refs`	`identifier` (of type `marking-definition`)		
Common Relationships			
`duplicate-of`, `derived-from`, `related-to`			
Source	**Relationship Type**	**Target**	**Description**
`threat-actor`	`attributed-to`	`identity`	This Relationship describes that the Threat Actor's real identity is the related Identity. For example, an `attributed-to` Relationship from the jay-sm17h Threat Actor to the John Smith Identity means that the actor known as jay-sm17h is John Smith.
`threat-actor`	`impersonates`	`identity`	This Relationship describes that the Threat Actor impersonates the related Identity. For example, an `impersonates` Relationship from the gh0st Threat Actor to the ACME Corp. Identity means that the actor known as gh0st impersonates ACME Corp.
`threat-actor`	`targets`	`identity`, `vulnerability`	This Relationship describes that the Threat Actor uses exploits of the related Vulnerability or targets the type of victims described by the related Identity. For example, a `targets` Relationship from the jay-sm17h Threat Actor to a Vulnerability in a blogging platform indicates that attacks performed by John Smith

			often exploit that Vulnerability. Similarly, a targets Relationship from the jay-sm17h Threat Actor to an Identity describing the energy sector in the United States means that John Smith often carries out attacks against targets in that sector.
threat-actor	uses	attack-pattern, malware, tool	This Relationship describes that attacks carried out as part of the Threat Actor typically use the related Attack Pattern, Malware, or Tool. For example, a uses Relationship from the jay-sm17h Threat Actor to the xInject Malware indicates that xInject is often used by John Smith.
Reverse Relationships			
campaign, intrusion-set	attributed-to	threat-actor	See forward relationship for definition.
indicator	indicates	threat-actor	See forward relationship for definition.

Examples

```
{
  "type": "threat-actor",
  "id": "threat-actor--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:03:48.000Z",
  "modified": "2016-04-06T20:03:48.000Z",
  "labels": [ "crime-syndicate"],
  "name": "Evil Org",
  "description": "The Evil Org threat actor group",
  "aliases": ["Syndicate 1", "Evil Syndicate 99"],
  "roles": "director",
  "goals": ["Steal bank money", "Steal credit cards"],
  "sophistication": "advanced",
  "resource_level": "team",
  "primary_motivation": "organizational-gain"
}
```

2.11 Tool

Type Name: tool

Tools are legitimate software that can be used by threat actors to perform attacks. Knowing how and when threat actors use such tools can be important for understanding how campaigns are executed. Unlike malware, 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. Remote access tools (e.g., RDP) and network scanning tools (e.g., Nmap) are examples of Tools that may be used by a Threat Actor during an attack.

The Tool SDO characterizes 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. It contains properties to name and describe the tool, a list of Kill Chain Phases the tool can be used to carry out, and the version of the tool.

This SDO **MUST NOT** be used to characterize malware. Further, Tool **MUST NOT** be used to characterise tools used as part of a course of action in response to an attack. Tools used during response activities can be included directly as part of a Course of Action SDO.

2.11.1 Properties

Common Properties		
`type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings`		
Tool Specific Properties		
`name, description, kill_chain_phases, tool_version`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this property **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 the Tool, 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 Tool can be used.
`tool_version` (optional)	`string`	The version identifier associated with the Tool.

2.11.2 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 the Tool object by way of the Relationship object. The reverse relationships (relationships "to" the Tool 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 type or, as with open vocabularies, user-defined names.

Embedded Relationships			
created_by_ref		identifier (of type identity)	
object_marking_refs		identifier (of type marking-definition)	
Common Relationships			
duplicate-of, derived-from, related-to			
Source	**Relationship Type**	**Target**	**Description**
tool	targets	identity, vulnerability	This Relationship documents that this Tool is being used to target this Identity or exploit the Vulnerability. For example, a targets Relationship linking an exploit Tool to a Vulnerability for CVE-2016-0001 means that the tool exploits that vulnerability. Similarly, a targets Relationship linking a DDoS Tool to an Identity representing the energy sector means that Tool is typically used against targets in the energy sector.
Reverse Relationships			
indicator	indicates	tool	See forward relationship for definition
course-of-action	mitigates	tool	See forward relationship for definition
attack-pattern, campaign, intrusion-set, malware, threat-actor	uses	tool	See forward relationship for definition

Examples

```
{
  "type": "tool",
  "id": "tool--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:03:48.000Z",
  "modified": "2016-04-06T20:03:48.000Z",
  "labels": [ "remote-access"],
  "name": "VNC"
}
```

2.12 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" [CVE]. For example, if a piece of malware exploits CVE-2015-12345, a Malware object could be linked to a Vulnerability object that references CVE-2015-12345.

The Vulnerability SDO is primarily used to link to external definitions of vulnerabilities or to describe 0-day vulnerabilities that do not yet have an external definition. Typically, other SDOs assert relationships to Vulnerability objects when a specific vulnerability is targeted and exploited as part of malicious cyber activity. As such, Vulnerability objects can be used as a linkage to the asset management and compliance process.

2.12.1 Properties

Common Properties		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
Vulnerability Specific Properties		
name, description		
Property Name	**Type**	**Description**
type (required)	`string`	The value of this property MUST be `vulnerability`.
external_references (optional)	`list` of type `external-reference`	A list of external references which refer to non-STIX information. This property **MAY** be used to provide one or more Vulnerability identifiers, such as a CVE ID [CVE]. When specifying a CVE ID, the **source_name** property of the external reference **MUST** be set to `cve` and the **external_id** property **MUST** be the exact CVE identifier.
name (required)	`string`	A 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.

2.12.2 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 the Vulnerability object by way of the Relationship object. None are defined for the Vulnerability object. The reverse relationships (relationships "to" the Vulnerability 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 type or, as with open vocabularies, user-defined names.

<table>
<tr><th colspan="4">Embedded Relationships</th></tr>
<tr><td colspan="2"><b><code>created_by_ref</code></b></td><td colspan="2"><code>identifier</code> (of type <code>identity</code>)</td></tr>
<tr><td colspan="2"><b><code>object_marking_refs</code></b></td><td colspan="2"><code>identifier</code> (of type <code>marking-definition</code>)</td></tr>
<tr><th colspan="4">Common Relationships</th></tr>
<tr><td colspan="4"><code>duplicate-of</code>, <code>derived-from</code>, <code>related-to</code></td></tr>
<tr><th>Source</th><th>Relationship Type</th><th>Target</th><th>Description</th></tr>
<tr><td>—</td><td>—</td><td>—</td><td>—</td></tr>
<tr><th colspan="4">Reverse Relationships</th></tr>
<tr><td><code>attack-pattern</code>, <code>campaign</code>, <code>intrusion-set</code>, <code>malware</code>, <code>threat-actor</code>, <code>tool</code></td><td><code>targets</code></td><td><code>vulnerability</code></td><td>See forward relationship for definition.</td></tr>
<tr><td><code>course-of-action</code></td><td><code>mitigates</code></td><td><code>vulnerability</code></td><td>See forward relationship for definition.</td></tr>
</table>

Examples

```
{
  "type": "vulnerability",
  "id": "vulnerability--0c7b5b88-8ff7-4a4d-aa9d-feb398cd0061",
  "created": "2016-05-12T08:17:27.000Z",
  "modified": "2016-05-12T08:17:27.000Z",
  "name": "CVE-2016-1234",
  "external_references": [
  {
  "source_name": "cve",
  "external_id": "CVE-2016-1234"
  }
  ]
}
```

3 STIX Relationship Objects

STIX Relationship Objects (SROs) represent types of relationships used to describe CTI. The generic Relationship SRO is used to describe many varied types of relationships, while the specific Sighting SRO contains additional properties to represent Sighting relationships.

Property information, relationship information, and examples are provided for each SRO defined below. Property information includes common properties as well as properties that are specific to each SRO. Because SROs cannot be the source or target of other SROs, relationship information is included but only to describe embedded relationships (e.g., **created_by_ref**).

3.1 Relationship

Type Name: relationship

The Relationship object is used to link together two SDOs in order to describe how they are related to each other. If SDOs are considered "nodes" or "vertices" in the graph, the Relationship Objects (SROs) represent "edges".

STIX defines many relationship types to link together SDOs. These relationships are contained in the "Relationships" table under each SDO definition. Relationship types defined in the specification **SHOULD** be used to ensure consistency. An example of a specification-defined relationship is that an indicator indicates a campaign. That relationship type is listed in the Relationships section of the Indicator SDO definition.

STIX also allows relationships from any SDO to any SDO that have not been defined in this specification. These relationships **MAY** use the related-to relationship type or **MAY** use a custom relationship type. As an example, a user might want to link malware directly to a tool. They can do so using related-to to say that the Malware is related to the Tool but not describe how, or they could use delivered-by (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 misleading, 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 known or sharable, while the high-level analysis is. For these reasons, relationships that might appear to be "shortcuts" are not excluded from STIX.

3.1.1 Specification-Defined Relationships Summary

This relationship summary table is provided as a convenience. If there is a discrepancy between this table and the relationships defined with each of the SDOs, then the relationships defined with the SDOs **MUST** be viewed as authoritative.

Source	Type	Target	Source	Type	Target
attack-pattern	targets	vulnerability	intrusion-set	attributed-to	threat-actor
attack-pattern	targets	identity	intrusion-set	targets	identity
attack-pattern	uses	malware	intrusion-set	targets	vulnerability
attack-pattern	uses	tool	intrusion-set	uses	attack-pattern
campaign	attributed-to	intrusion-set	intrusion-set	uses	malware
campaign	attributed-to	threat-actor	intrusion-set	uses	tool

campaign	targets	identity	malware	targets	identity
campaign	targets	vulnerability	malware	targets	vulnerability
campaign	uses	attack-pattern	malware	uses	tool
campaign	uses	malware	malware	variant-of	malware
campaign	uses	tool	threat-actor	attributed-to	identity
course-of-action	mitigates	attack-pattern	threat-actor	impersonates	identity
course-of-action	mitigates	malware	threat-actor	targets	identity
course-of-action	mitigates	tool	threat-actor	targets	vulnerability
course-of-action	mitigates	vulnerability	threat-actor	uses	attack-pattern
indicator	indicates	attack-pattern	threat-actor	uses	malware
indicator	indicates	campaign	threat-actor	uses	tool
indicator	indicates	intrusion-set	tool	targets	identity
indicator	indicates	malware	tool	targets	vulnerability
indicator	indicates	threat-actor			
indicator	indicates	tool			

3.1.2 Properties

<table>
<tr><th colspan="3">Common Properties</th></tr>
<tr><td colspan="3"><b>type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings</b></td></tr>
<tr><th colspan="3">Relationship Specific Properties</th></tr>
<tr><td colspan="3"><b>relationship_type, description, source_ref, target_ref</b></td></tr>
<tr><th>Property Name</th><th>Type</th><th>Description</th></tr>
<tr><td><b>type</b> (required)</td><td>string</td><td>The value of this property <b>MUST</b> be relationship.</td></tr>
<tr><td><b>relationship_type</b> (required)</td><td>string</td><td>The name used to identify the type of Relationship. This value <b>SHOULD</b> be an exact value listed in the relationships for the source and target SDO, but <b>MAY</b> be any string. The value of this property <b>MUST</b> be in ASCII and is limited to characters a–z (lowercase ASCII), 0–9, and hyphen (-).</td></tr>
<tr><td><b>description</b> (optional)</td><td>string</td><td>A description that provides more details and context about the Relationship, potentially including its purpose and its</td></tr>
</table>

		key characteristics.
`source_ref` (required)	`identifier`	The **`id`** of the source (from) object. The value **MUST** be an ID reference to an SDO (i.e., it cannot point to an SRO, Bundle, or Marking Definition).
`target_ref` (required)	`identifier`	The **`id`** of the target (to) object. The value **MUST** be an ID reference to an SDO (i.e., it cannot point to an SRO, Bundle, or Marking Definition).

3.1.3 Relationships

There are no relationships between the Relationship object and other objects, other than the embedded relationships listed below by property name along with their corresponding target.

Embedded Relationships	
`created_by_ref`	`identifier` (of type `identity`)
`object_marking_refs`	`identifier` (of type `marking-definition`)

3.2 Sighting

Type Name: `sighting`

A Sighting denotes the belief that something in CTI (e.g., an indicator, malware, tool, threat actor, etc.) was seen. Sightings are used to track who and what are being targeted, how attacks are carried out, and to track trends in attack behavior.

The Sighting relationship object is a special type of SRO; it is a relationship that contains extra properties not present on the generic Relationship object. These extra properties are included to represent data specific to sighting relationships (e.g., **`count`**, representing how many times something was seen), but for other purposes a Sighting can be thought of as a Relationship with a name of "sighting-of". Sighting is captured as a relationship because you cannot have a sighting unless you have something that has been sighted. Sighting does not make sense without the relationship to what was sighted.

Sighting relationships relate three aspects of the sighting:

- What was sighted, such as the Indicator, Malware, Campaign, or other SDO (**`sighting_of_ref`**)
- Who sighted it and/or where it was sighted, represented as an Identity (**`where_sighted_refs`**) and
- What was actually seen on systems and networks, represented as Observed Data (**`observed_data_refs`**)

What was sighted is required; a sighting does not make sense unless you say what you saw. Who sighted it, where it was sighted, and what was actually seen are optional. In many cases it is not necessary to provide that level of detail in order to provide value.

Sightings are used whenever any SDO has been "seen". In some cases, the object creator wishes to convey very little information about the sighting; the details might be sensitive, but the fact that they saw a malware instance or threat actor could still be very useful. In other cases, providing the details may be helpful or even necessary; saying exactly which of the 1000 IP addresses in an indicator were sighted is helpful when tracking which of those IPs is still malicious.

Sighting is distinct from Observed Data in that Sighting is an intelligence assertion ("I saw this threat actor") while Observed Data is simply information ("I saw this file"). When you combine them by including the linked Observed Data (**`observed_data_refs`**) from a Sighting, you can say "I saw this file, and that makes me think I saw this threat actor". Although **confidence** is currently reserved, notionally confidence would be added to Sighting (the intelligence relationship) but not to Observed Data (the raw information).

3.2.1 Properties

Common Properties		
`type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings`		
Sighting Specific Properties		
`first_seen, last_seen, count, sighting_of_ref, observed_data_refs, where_sighted_refs, summary`		
Property Name	**Type**	**Description**
`type` (required)	`string`	The value of this property **MUST** be `sighting`.
`first_seen` (optional)	`timestamp`	The beginning of the time window during which the SDO referenced by the `sighting_of_ref` property was sighted.
`last_seen` (optional)	`timestamp`	The end of the time window during which the SDO referenced by the `sighting_of_ref` property was sighted.
`count` (optional)	`integer`	This **MUST** be an integer between 0 and 999,999,999 inclusive and represents the number of times the SDO referenced by the `sighting_of_ref` property was sighted. Observed Data has a similar property called **`number_observed`**, which refers to the number of times the data was observed. These counts refer to different concepts and are distinct. For example, a single sighting of a DDoS bot might have many millions of observations of the network traffic that it generates. Thus, the Sighting **`count`** would be 1 (the bot was observed once) but the Observed Data **`number_observed`** would be much higher. As another example, a sighting with a count of 0 can be used to express that an indicator was not seen at all.
`sighting_of_ref` (required)	`identifier`	An ID reference to the SDO that was sighted (e.g., Indicator or Malware).

		For example, if this is a Sighting of an Indicator, that Indicator's ID would be the value of this property. This property **MUST** reference only an SDO or a Custom Object.
`observed_data_refs` (optional)	`list` of type `identifier`	A list of ID references to the Observed Data objects that contain the raw cyber data for this Sighting. For example, a Sighting of an Indicator with an IP address could include the Observed Data for the network connection that the Indicator was used to detect. This property **MUST** reference only Observed Data SDOs.
`where_sighted_refs` (optional)	`list` of type `identifier`	A list of ID references to the Identity (victim) objects of the entities that saw the sighting. Omitting the **`where_sighted_refs`** property does not imply that the sighting was seen by the object creator. To indicate that the sighting was seen by the object creator, an Identity representing the object creator should be listed in **`where_sighted_refs`**. This property **MUST** reference only Identity SDOs.
`summary` (optional)	`boolean`	The **`summary`** property indicates whether the Sighting should be considered summary data. Summary data is an aggregation of previous Sightings reports and should not be considered primary source data. Default value is `false`.

3.2.2 Relationships

There are no relationships between the Sighting object and other objects, other than the embedded relationships listed below by property name along with their corresponding target.

Embedded Relationships	
`created_by_ref`	`identifier` (of type `identity`)
`object_marking_refs`	`identifier` (of type `marking-definition`)
`sighting_of_ref`	`identifier` (of type any STIX Object type)
`observed_data_refs`	`list` of type `identifier` (of type `observed-data`)
`where_sighted_refs`	`list` of type `identifier` (of type `identity`)

Examples

Sighting of Indicator, without Observed Data

```
{
  "type": "sighting",
  "id": "sighting--ee20065d-2555-424f-ad9e-0f8428623c75",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:08:31.000Z",
  "modified": "2016-04-06T20:08:31.000Z",
  "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_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:08:31.000Z",
  "modified": "2016-04-06T20:08:31.000Z",
  "first_seen": "2015-12-21T19:00:00Z",
  "last_seen": "2015-12-21T19:00:00Z",
  "count": 50,
  "sighting_of_ref": "indicator--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "observed_data_refs": ["observed-data--b67d30ff-02ac-498a-92f9-32f845f448cf"],
  "where_sighted_refs": ["identity--b67d30ff-02ac-498a-92f9-32f845f448ff"]
  },
  {
  "type": "observed-data",
  "id": "observed-data--b67d30ff-02ac-498a-92f9-32f845f448cf",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T19:58:16.000Z",
  "modified": "2016-04-06T19:58:16.000Z",
  "start": "2015-12-21T19:00:00Z",
  "stop": "2016-04-06T19:58:16Z",
  "count": 50,
  "objects": {
  "0": {
    "type": "file",
    ...
  }
  }
  }
]
```

4 Conformance

4.1 Object Producers

A "STIX 2.0 Producer" that creates an object from section 2 (STIX Domain Objects) or section 3 (STIX Relationship Objects) is a "Producer" of that object. Object producers **MUST** conform to all normative requirements in the section for that object.

For example, a "STIX 2.0 Producer" that can produce Indicators is an "Indicator Producer". That producer has to conform to all normative requirements in section 2.5, Indicator.

4.2 Object Consumers

A "STIX 2.0 Consumer" that receives an object from section 2 (STIX Domain Objects) or section 3 (STIX Relationship Objects) is a "Consumer" of that object. Object consumers **MUST** conform to all normative requirements in the section for that object.

For example, a "STIX 2.0 Consumer" that can receive Campaigns is a "Campaign Consumer". That consumer has to conform to all normative requirements in section 2.2, Campaign.

Appendix A. Glossary

CAPEC - Common Attack Pattern Enumeration and Classification

Consumer - Any entity that receives STIX content

CTI - Cyber Threat Intelligence

Embedded Relationship - A link (an "edge" in a graph) between one STIX Object and another represented as a property on one object containing the ID of another object

Entity - Anything that has a separately identifiable existence (e.g., organization, person, group, etc.)

IEP - FIRST (Forum of Incident Response and Security Teams) Information Exchange Policy

Instance - A single occurrence of a STIX object version

MTI - Mandatory To Implement

MVP - Minimally Viable Product

Object Creator - The entity that created or updated a STIX object (see section 3.3 of *STIX™ Version 2.0. Part 1: STIX Core Concepts*).

Object Representation - An instance of an object version that is serialized as STIX

Producer - Any entity that distributes STIX content, including object creators as well as those passing along existing content

SDO - STIX Domain Object (a "node" in a graph)

SRO - STIX Relationship Object (one mechanism to represent an "edge" in a graph)

STIX - Structured Threat Information Expression

STIX Content - STIX documents, including STIX Objects, STIX Objects grouped as bundles, etc.

STIX Object - A STIX Domain Object (SDO) or STIX Relationship Object (SRO)

STIX Relationship - A link (an "edge" in a graph) between two STIX Objects represented by either an SRO or an embedded relationship

TAXII - An application layer protocol for the communication of cyber threat information

TLP - Traffic Light Protocol

TTP - Tactic, technique, or procedure; behaviors and resources that attackers use to carry out their attacks

Appendix B. Acknowledgments

STIX Subcommittee Chairs:

Sarah Kelley, Center for Internet Security (CIS)

John Wunder, MITRE Corporation

Cyber Observable Subcommittee Chairs:

Trey Darley, Kingfisher Operations, sprl

Ivan Kirillov, MITRE Corporation

Special Thanks:

Substantial contributions to this specification from the following individuals are gratefully acknowledged:

Sarah Kelley, Center for Internet Security (CIS)

Terry MacDonald, Cosive

Jane Ginn, Cyber Threat Intelligence Network, Inc. (CTIN)

Richard Struse, DHS Office of Cybersecurity and Communications

Iain Brown, GDS

Jason Keirstead, IBM

Tim Casey, Intel

Trey Darley, Kingfisher Operations, sprl

Allan Thomson, LookingGlass Cyber

Greg Back, MITRE Corporation

Ivan Kirillov, MITRE Corporation

Jon Baker, MITRE Corporation

John Wunder, MITRE Corporation

Sean Barnum, MITRE Corporation

Richard Piazza, MITRE Corporation

Christian Hunt, New Context Services, Inc.

John-Mark Gurney, New Context Services, Inc.

Aharon Chernin, Perch

Dave Cridland, Surevine

Bret Jordan, Symantec Corp.

Participants:

The following individuals were members of the OASIS CTI Technical Committee during the creation of this specification and their contributions are gratefully acknowledged:

David Crawford, Aetna

Marcos Orallo, Airbus Group SAS
Roman Fiedler, AIT Austrian Institute of Technology
Florian Skopik, AIT Austrian Institute of Technology
Russell Spitler, AlienVault
Ryan Clough, Anomali
Nicholas Hayden, Anomali
Wei Huang, Anomali
Angela Nichols, Anomali
Hugh Njemanze, Anomali
Katie Pelusi, Anomali
Dean Thompson, Australia and New Zealand Banking Group (ANZ Bank)
Alexander Foley, Bank of America
Sounil Yu, Bank of America
Vicky Laurens, Bank of Montreal
Humphrey Christian, Bay Dynamics
Ryan Stolte, Bay Dynamics
Alexandre Dulaunoy, CIRCL
Andras Iklody, CIRCL
Rapha'l Vinot, CIRCL
Sarah Kelley, CIS
Syam Appala, Cisco Systems
Ted Bedwell, Cisco Systems
David McGrew, Cisco Systems
Mark-David McLaughlin, Cisco Systems
Pavan Reddy, Cisco Systems
Omar Santos, Cisco Systems
Jyoti Verma, Cisco Systems
Doug DePeppe, Cyber Threat Intelligence Network, Inc. (CTIN)
Jane Ginn, Cyber Threat Intelligence Network, Inc. (CTIN)
Ben Othman, Cyber Threat Intelligence Network, Inc. (CTIN)
Jeff Odom, Dell
Sreejith Padmajadevi, Dell
Ravi Sharda, Dell
Will Urbanski, Dell
Sean Sobieraj, DHS Office of Cybersecurity and Communications (CS&C)
Richard Struse, DHS Office of Cybersecurity and Communications (CS&C)
Marlon Taylor, DHS Office of Cybersecurity and Communications (CS&C)
Jens Aabol, Difi-Agency for Public Management and eGovernment
Wouter Bolsterlee, EclecticIQ
Marko Dragoljevic, EclecticIQ
Oliver Gheorghe, EclecticIQ
Joep Gommers, EclecticIQ

Sergey Polzunov, EclecticIQ
Rutger Prins, EclecticIQ
Andrei S”rghi, EclecticIQ
Raymon van der Velde, EclecticIQ
Ben Sooter, Electric Power Research Institute (EPRI)
Chris Ricard, Financial Services Information Sharing and Analysis Center (FS-ISAC)
Phillip Boles, FireEye, Inc.
Prasad Gaikwad, FireEye, Inc.
Rajeev Jha, FireEye, Inc.
Anuj Kumar, FireEye, Inc.
Shyamal Pandya, FireEye, Inc.
Paul Patrick, FireEye, Inc.
Scott Shreve, FireEye, Inc.
Jon Warren, FireEye, Inc.
Remko Weterings, FireEye, Inc.
Gavin Chow, Fortinet Inc.
Steve Fossen, Fortinet Inc.
Kenichi Terashita, Fortinet Inc.
Ryusuke Masuoka, Fujitsu Limited
Daisuke Murabayashi, Fujitsu Limited
Derek Northrope, Fujitsu Limited
Jonathan Algar, GDS
Iain Brown, GDS
Adam Cooper, GDS
Mike McLellan, GDS
Tyrone Nembhard, GDS
Chris O'Brien, GDS
James Penman, GDS
Howard Staple, GDS
Chris Taylor, GDS
Laurie Thomson, GDS
Alastair Treharne, GDS
Julian White, GDS
Bethany Yates, GDS
Robert van Engelen, Genivia
Eric Burger, Georgetown University
Allison Miller, Google Inc.
Mark Risher, Google Inc.
Yoshihide Kawada, Hitachi, Ltd.
Jun Nakanishi, Hitachi, Ltd.
Kazuo Noguchi, Hitachi, Ltd.
Akihito Sawada, Hitachi, Ltd.

Yutaka Takami, Hitachi, Ltd.

Masato Terada, Hitachi, Ltd.

Peter Allor, IBM

Eldan Ben-Haim, IBM

Allen Hadden, IBM

Sandra Hernandez, IBM

Jason Keirstead, IBM

John Morris, IBM

Laura Rusu, IBM

Ron Williams, IBM

Paul Martini, iboss, Inc.

Jerome Athias, Individual

Peter Brown, Individual

Joerg Eschweiler, Individual

Stefan Hagen, Individual

Elysa Jones, Individual

Sanjiv Kalkar, Individual

Terry MacDonald, Individual

Alex Pinto, Individual

Tim Casey, Intel Corporation

Kent Landfield, Intel Corporation

Karin Marr, Johns Hopkins University Applied Physics Laboratory

Julie Modlin, Johns Hopkins University Applied Physics Laboratory

Mark Moss, Johns Hopkins University Applied Physics Laboratory

Mark Munoz, Johns Hopkins University Applied Physics Laboratory

Nathan Reller, Johns Hopkins University Applied Physics Laboratory

Pamela Smith, Johns Hopkins University Applied Physics Laboratory

David Laurance, JPMorgan Chase Bank, N.A.

Russell Culpepper, Kaiser Permanente

Beth Pumo, Kaiser Permanente

Michael Slavick, Kaiser Permanente

Trey Darley, Kingfisher Operations, sprl

Gus Creedon, Logistics Management Institute

Wesley Brown, LookingGlass

Jamison Day, LookingGlass

Kinshuk Pahare, LookingGlass

Allan Thomson, LookingGlass

Ian Truslove, LookingGlass

Chris Wood, LookingGlass

Greg Back, Mitre Corporation

Jonathan Baker, Mitre Corporation

Sean Barnum, Mitre Corporation

Desiree Beck, Mitre Corporation
Michael Chisholm, Mitre Corporation
Nicole Gong, Mitre Corporation
Ivan Kirillov, Mitre Corporation
Michael Kouremetis, Mitre Corporation
Chris Lenk, Mitre Corporation
Richard Piazza, Mitre Corporation
Larry Rodrigues, Mitre Corporation
Jon Salwen, Mitre Corporation
Charles Schmidt, Mitre Corporation
Alex Tweed, Mitre Corporation
Emmanuelle Vargas-Gonzalez, Mitre Corporation
John Wunder, Mitre Corporation
James Cabral, MTG Management Consultants, LLC.
Scott Algeier, National Council of ISACs (NCI)
Denise Anderson, National Council of ISACs (NCI)
Josh Poster, National Council of ISACs (NCI)
Mike Boyle, National Security Agency
Joe Brule, National Security Agency
Jessica Fitzgerald-McKay, National Security Agency
David Kemp, National Security Agency
Shaun McCullough, National Security Agency
John Anderson, NC4
Michael Butt, NC4
Mark Davidson, NC4
Daniel Dye, NC4
Angelo Mendonca, NC4
Michael Pepin, NC4
Natalie Suarez, NC4
Benjamin Yates, NC4
Daichi Hasumi, NEC Corporation
Takahiro Kakumaru, NEC Corporation
Lauri Korts-P_rn, NEC Corporation
John-Mark Gurney, New Context Services, Inc.
Christian Hunt, New Context Services, Inc.
Daniel Riedel, New Context Services, Inc.
Andrew Storms, New Context Services, Inc.
Stephen Banghart, NIST
David Darnell, North American Energy Standards Board
Cory Casanave, Object Management Group
Aharon Chernin, Perch
Dave Eilken, Perch

Sourabh Satish, Phantom
Josh Larkins, PhishMe Inc.
John Tolbert, Queralt Inc.
Ted Julian, Resilient Systems, Inc..
Igor Baikalov, Securonix
Joseph Brand, Semper Fortis Solutions
Duncan Sparrell, sFractal Consulting LLC
Thomas Schreck, Siemens AG
Rob Roel, Southern California Edison
Dave Cridland, Surevine Ltd.
Bret Jordan, Symantec Corp.
Curtis Kostrosky, Symantec Corp.
Juha Haaga, Synopsys
Masood Nasir, TELUS
Greg Reaume, TELUS
Alan Steer, TELUS
Crystal Hayes, The Boeing Company
Wade Baker, ThreatConnect, Inc.
Cole Iliff, ThreatConnect, Inc.
Andrew Pendergast, ThreatConnect, Inc.
Ben Schmoker, ThreatConnect, Inc.
Jason Spies, ThreatConnect, Inc.
Ryan Trost, ThreatQuotient, Inc.
Patrick Coughlin, TruSTAR Technology
Chris Roblee, TruSTAR Technology
Mark Angel, U.S. Bank
Brian Fay, U.S. Bank
Joseph Frazier, U.S. Bank
Mark Heidrick, U.S. Bank
Mona Magathan, U.S. Bank
Yevgen Sautin, U.S. Bank
Richard Shok, U.S. Bank
James Bohling, US Department of Defense (DoD)
Eoghan Casey, US Department of Defense (DoD)
Gary Katz, US Department of Defense (DoD)
Jeffrey Mates, US Department of Defense (DoD)
Evette Maynard-Noel, US Department of Homeland Security
Robert Coderre, VeriSign
Kyle Maxwell, VeriSign
Eric Osterweil, VeriSign
Patrick Maroney, Wapack Labs LLC
Anthony Rutkowski, Yanna Technologies LLC

Appendix C. Revision History

Revision	Date	Editor	Changes Made
01	2017-01-20	Bret Jordan, John Wunder, Rich Piazza, Ivan Kirillov, Trey Darley	Initial Version
02	2017-04-24	Bret Jordan, John Wunder, Rich Piazza, Ivan Kirillov, Trey Darley	Changes made from first public review

부 록 Ⅰ-1

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

지식재산권 확약서 정보

Ⅰ-1.1 지식재산권 확약서(1)

- 해당 사항 없음

Ⅰ-1.2 지식재산권 확약서(2)

- 해당 사항 없음

※ 상기 기재된 지식재산권 확약서 이외에도 본 표준이 발간된 후 접수된 확약서가 있을 수 있으니, TTA 웹사이트에서 확인하시기 바랍니다.

부 록 Ⅰ-2

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

시험인증 관련 사항

Ⅰ-2.1 시험인증 대상 여부

- 해당 사항 없음

Ⅰ-2.2 시험표준 제정 현황

- 해당 사항 없음

부 록 Ⅰ-3

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

본 표준의 연계(family) 표준

Ⅰ-3.1 TTA, 구조화된 위협 정보 표현 규격(STIX^{TM}) 버전 2.0 - 제1부: STIX 핵심 개념

STIX의 핵심 개념을 정의하는 문서로 공통 데이터 형식, STIX 객체, 데이터 표시 등에 대한 설명을 제공

Ⅰ-3.2 TTA, 구조화된 위협 정보 표현 규격(STIX^{TM}) 버전 2.0 - 제3부: STIX 사이버 관찰 코어 개념

STIX의 Observable 핵심 개념을 정의하는 문서로 관측 가능한 객체를 구성하는 필드와 필드에 대한 설명을 제공

Ⅰ-3.3 TTA, 구조화된 위협 정보 표현 규격(STIX^{TM}) 버전 2.0 - 제4부: STIX 사이버 관찰 객체

STIX의 관측 가능한 객체 집합을 정의하는 문서로 관측 가능한 객체의 구성요소와 구성요소에 대한 설명을 제공

Ⅰ-3.4 TTA, 구조화된 위협 정보 표현 규격(STIX^{TM}) 버전 2.0 - 제5부: STIX 패터닝

STIX의 인디케이터 지원 패턴을 정의하는 문서로 인디케이터 지원 패턴을 구성하는 필드와 필드에 대한 설명을 제공

부 록 Ⅰ-4

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

참고 문헌

[1] NIST SP 800-83, “Guide to Malware Incident Prevention and Handling”, November 2011., http://csrc.nist.gov/publications/nistpubs/800-83/SP800-83.pdf.

[2] CAPEC, “Common Attack Pattern Enumeration and Classification”, November 2014., The MITRE Corporation., http://capec.mitre.org.

[3] CVE, “Common Vulnerabilities and Exposures”, 1999, The MITRE Corporation., http://cve.mitre.org.

[4] RFC 2119, Bradner, S., “"Key words for use in RFCs to Indicate Requirement Levels", BCP 14, DOI 10.17487/RFC2119, March 1997, http://www.rfc-editor.org/info/rfc2119

부록 Ⅰ-5

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

영문표준 해설서

Ⅰ-5.1 개요

STIX(Structured Threat Information Expression)는 사이버 위협 인텔리전스(CTI)를 교환하는데 사용하는 언어이고 직렬화 규격이다. STIX를 이용하여 조직들은 CTI를 일관성 있고 자동화된 해석이 가능한 방식으로 서로 공유할 수 있으므로, 보안 커뮤니티는 예상 가능한 사이버 공격을 보다 잘 이해하고 효과적으로 예측, 대응할 수 있다.

Ⅰ-5.2 STIX 도메인 객체

이 표준은 STIX 도메인 객체(SDO, STIX Domain Object)의 집합을 정의하며, 각 SDO는 일반적으로 CTI로 표현되는 고유한 개념에 해당한다. SDO와 STIX 관계를 구성요소로 사용하여 광범위하고 포괄적인 CTI를 만들고 공유할 수 있다.

Ⅰ-5.2.1 공격 패턴(Attack Pattern)

공격 패턴은 악의적 사용자가 대상에게 공격을 시도하는 방법을 설명하는 TTP의 한 형식이다. 공격 패턴은 공격을 분류하고, 특정 공격을 해당 공격이 따르는 패턴으로 일반화하고, 공격이 수행되는 방법에 관한 자세한 정보를 제공하기 위해 사용된다. 공격 패턴은 CAPEC(Common Attack Pattern Enumeration and Classification)처럼 외부에서 정의된 공격 분류법에 대한 참조와 함께 패턴에 대한 텍스트 설명을 포함하고 있다. 공격 패턴의 관계를 사용하면 해당 패턴이 표적으로 삼는 대상과 사용하는 도구 및 멀웨어의 종류를 관련시킬 수 있다.

<표 Ⅰ-5.2-1> 공격패턴 속성

공통 속성		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
공격 패턴 고유 속성		
name, description, kill_chain_phases		
속성 이름	형식	설명
type(필수)	string	이 속성의 값은 attack-pattern이어야 한다.
external_references (선택사항)	list of type external-reference	STIX가 아닌 정보를 참조하는 외부 참조 목록. 이 속성은 CAPEC ID 같은 하나 이상의 공격 패턴 ID를 제공하는 데 사용할 수 있다.
name(필수)	string	공격 패턴을 식별하는 데 사용되는 이름.
description(선택사항)	string	공격 패턴에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.
kill_chain_phases (선택사항)	list of type kill-chain-phase	이 공격 패턴이 사용되는 킬 체인 단계의 목록.

Ⅰ-5.2.2 캠페인(Campaign)

캠페인은 특정 목표군을 대상으로 일정 기간 동안 발생하는 일단의 악의적 활동 또는 공격을 설명하는 악의적 동작의 그룹이다. 캠페인은 잘 정의된 목표를 가지고 있으며 대개 침입 단체의 일부일 수 있다. 캠페인은 흔히 침입 단체(Intrusion Set)과 위협 행위자로 인해 이루어진다. 위협 행위자는 침입 단체(Intrusion Set)의 알려진 인프라를 다시 사용할 수도 있고 해당 캠페인을 수행하기 위한 새로운 전용 인프라를 설치할 수도 있다. 캠페인은 그 목표와 해당 캠페인이 야기하는 침해사고, 목표로 하는 사람이나 리소스 및 사용하는 리소스(인프라, 인텔리전스, 멀웨어, 도구 등)에 따라 특성이 결정될 수 있다.

<표 Ⅰ-5.2-2> 캠페인 속성

공통 속성		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
캠페인 고유 속성		
name, description, aliases, first_seen, last_seen, objective		
속성 이름	형식	설명
type(필수)	string	이 속성의 값은 campaign이어야 한다.
name(필수)	string	캠페인을 식별하는 데 사용되는 이름.
description(선택사항)	string	캠페인에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.
aliases(선택사항)	list of type string	이 캠페인을 식별하는 데 사용되는 대체 이름.
first_seen(선택사항)	timestamp	이 캠페인이 처음 발견된 시간.
last_seen(선택사항)	timestamp	이 캠페인이 마지막 발견된 시간.
objective(선택사항)	string	이 속성은 캠페인의 주 목적, 목표, 원하는 결과 또는 의도한 효과, 즉 위협 행위자가 이 캠페인을 통해 달성하고자 하는 것을 정의한다.

Ⅰ-5.2.3 조치(Course of Action)

참고: STIX 2.0의 조치 객체는 스텁(stub)이다. 기본적인 사용 사례(평범한 조치의 공유 등)를 지원하기 위해 포함되었지만 자동화된 조치를 표시하는 기능을 지원하거나 조치에 관한 메타데이터를 표현하는 속성을 포함하지 않는다. 향후 STIX 2 릴리스는 이러한 기능을 포함하도록 확장될 것이다.

조치는 공격을 예방하거나 진행 중인 공격에 대응하기 위해 실행한 작업이다. 기술적이고 자동화가 가능한 대응(패치 적용, 방화벽 재구성)을 설명할 수 있지만 직원 훈련이나 정책 변경 같은 더 높은 수준의 작업을 설명할 수도 있다. 예를 들어 취약점을 완화하기 위한 조치는 해당 취약점을 해결하는 패치의 적용을 설명할 수 있다.

<표 I-5.2-3> 조치 속성

<table>
<tr><th colspan="3">공통 속성</th></tr>
<tr><td colspan="3">type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings</td></tr>
<tr><th colspan="3">조치 고유 속성</th></tr>
<tr><td colspan="3">name, description, action</td></tr>
<tr><th>속성 이름</th><th>형식</th><th>설명</th></tr>
<tr><td>type(필수)</td><td>string</td><td>이 속성의 값은 course-of-action이어야 한다.</td></tr>
<tr><td>name(필수)</td><td>string</td><td>조치를 식별하는 데 사용되는 이름.</td></tr>
<tr><td>description(선택사항)</td><td>string</td><td>조치에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.</td></tr>
<tr><td>action(예약)</td><td>RESERVED</td><td>예약(구조화/자동화 조치를 포착하기 위한 속성)</td></tr>
</table>

I -5.2.4 ID(Identity)

ID는 실제 개인, 조직 또는 그룹(예: ACME, Inc.)은 물론 개인, 조직 또는 그룹의 부류(예: 재무 부문)도 표현할 수 있다. ID는 기본적인 식별 정보, 연락처 정보 및 ID가 속한 부문을 포착할 수 있다. ID는 STIX에서 특히 공격 목표, 정보 출처, 객체 작성자 및 위협 행위자 ID를 표현하는 데 사용된다.

<표 I-5.2-4> ID 속성

<table>
<tr><th colspan="3">공통 속성</th></tr>
<tr><td colspan="3">type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings</td></tr>
<tr><th colspan="3">ID 고유 속성</th></tr>
<tr><td colspan="3">name, description, identity_class, sectors, contact_information</td></tr>
<tr><th>속성 이름</th><th>형식</th><th>설명</th></tr>
<tr><td>type(필수)</td><td>string</td><td>이 속성의 값은 identity이어야 한다.</td></tr>
<tr><td>labels(선택사항)</td><td>string 형식의 list</td><td>이 ID가 수행하는 역할(예: CEO, 도메인 관리자, 의사, 병원 또는 소매점)의 목록. 이 속성에 대한 개방형 어휘는 아직 정의되어 있지 않다.</td></tr>
<tr><td>name(필수)</td><td>string</td><td>이 ID의 이름. 특정 엔터티(예: 개인 또는 조직)를 참조할 때 이 속성은 해당 특정 엔터티의 정식 이름을 포함하는 것이 바람직하다.</td></tr>
<tr><td>description(선택사항)</td><td>string</td><td>ID에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.</td></tr>
<tr><td>identity_class(필수)</td><td>open-vocab</td><td>이 ID가 설명하는 형식(예: 개인 또는 조직).
이는 개방형 어휘이며 값은 identity-class-ov 어휘에서 가져오는 것이 바람직하다.</td></tr>
<tr><td>sectors(선택사항)</td><td>open-vocab의 list</td><td>이 ID가 속한 산업 부문의 목록.
이는 개방형 어휘이며 값은 industry-sector-ov 어휘에서 가져오는 것이 바람직하다.</td></tr>
<tr><td>contact_information (선택사항)</td><td>string</td><td>이 ID에 대한 연락처 정보(이메일, 전화번호 등). 이 사양에서 이 정보에 대한 형식은 현재 정의하지 않고 있다.</td></tr>
</table>

Ⅰ-5.2.5 인디케이터(Indicator)

인디케이터는 수상한 또는 악의적 사이버 활동을 검색하는데 사용할 수 있는 패턴을 포함하고 있다. 예를 들어 인디케이터를 사용하여 일단의 악의적 도메인을 표현하고 STIX 패턴화 언어(STIX™ Version 2.0. Part 5: STIX Patterning)를 사용하여 이러한 도메인을 지정할 수 있다.

인디케이터는 단순한 텍스트 설명, 동작을 검색하는 킬 체인 단계, 인디케이터가 유효하거나 유용한 시간 범위 및 구조화 검색 패턴을 포착하는 데 필요한 pattern 속성을 포함하고 있다. STIX 구현을 준수한다면 STIX™ Version 2.0. Part 5: STIX Patterning에 정의된 STIX 패턴화 언어를 지원해야 한다. 각 구조화 패턴 언어는 서로 다른 구문과 잠재적으로 서로 다른 어의를 가지고 있지만, 일반적으로 인디케이터는 해당 인디케이터를 평가하는 컨텍스트에서 구조화 패턴에 지정한 조건이 충족될 때 "일치했다"(또는 "발견되었다")고 간주된다. 인디케이터에서 시작하는 관계는 직접 검색하는 악의적 또는 수상한 동작(멀웨어, 도구 및 공격 패턴)은 물론 그 존재를 나타낼 수 있는 캠페인, 침입 단체 및 위협 행위자도 설명할 수 있다.

<표 Ⅰ-5.2-5> 인디케이터 속성

공통 속성		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
인디케이터 고유 속성		
name, description, pattern, valid_from, valid_until, kill_chain_phases		
속성 이름	형식	설명
type(필수)	string	이 속성의 값은 인디케이터여야 한다.
labels(필수)	open-vocab의 list	이 속성은 인디케이터의 형식을 지정하는 개방형 어휘이다. 이는 개방형 어휘이며 값은 indicator-label-ov 어휘에서 가져오는 것이 바람직하다.
name(선택사항)	string	인디케이터를 식별하는 데 사용되는 이름.
description(선택사항)	string	인디케이터에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.
pattern(필수)	string	이 인디케이터에 대한 검색 패턴은 STIX™ Version 2.0. Part 5: STIX Patterning에 지정된 STIX 패턴이다.
valid_from(필수)	timestamp	이 인디케이터를 중요 인텔리전스로 간주하는 것이 바람직한 시간.
valid_until(선택사항)	timestamp	이 인디케이터가 더 이상 중요 인텔리전스로 간주되지 않는 시간. valid_until 속성이 생략된 경우 인디케이터를 사용해야 하는 마지막 시간에 대한 구속조건은 없다.
kill_chain_phases(선택사항)	kill-chain-phase 형식의 list	이 인디케이터가 해당하는 킬 체인 단계.

I -5.2.6 침입 단체(Intrusion Set)

침입 단체는 단일 조직이 지휘한다고 여겨지는 공통 속성을 가진 악의적 동작과 리소스의 그룹화된 집합체이다. 침입 단체는 공통의 알려진 또는 알려지지 않은 위협 행위자를 나타내는 공유 속성에 의해 서로 결합되는 여러 캠페인 또는 다른 활동을 포함할 수 있다. 공격의 이면에 존재하는 위협 행위자를 알지 못하는 경우에도 새로운 활동은 침입 단체에 포함될 수 있다. 위협 행위자는 하나의 지원 침입 단체에서 다른 침입 단체로 이동하거나 복수의 침입 단체를 지원할 수 있다. 캠페인이 어떤 목표를 달성하기 위해 특정 목표군을 기준으로 일정 기간 동안 이루어지는 공격의 집합체인 경우, 침입 단체는 전체 공격 패키지이며 복수의 목적을 달성하기 위해 복수의 캠페인에서 매우 장기간에 걸쳐 사용될 수 있다.

<표 I-5.2-6> 침입단체 속성

공통 속성		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
침입 단체 고유 속성		
name, description, aliases, first_seen, last_seen, goals, resource_level, primary_motivation, secondary_motivations		
속성 이름	형식	설명
type(필수)	string	이 속성의 값은 intrusion-set이어야 한다.
name(필수)	string	이 침입 단체를 식별하는 데 사용되는 이름.
description(선택사항)	string	침입 단체에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.
aliases(선택사항)	list of type string	이 침입 단체를 식별하는 데 사용되는 대체 이름.
first_seen(선택사항)	timestamp	이 침입 단체가 처음 발견된 시간.
last_seen(선택사항)	timestamp	이 침입 단체가 마지막 발견된 시간.
goals(선택사항)	string 형식의 list	이 침입 단체의 고급 목적, 즉 수행하려고 시도하는 목적.
resource_level(선택사항)	open-vocab	이 침입 단체가 일반적으로 일하고, 그에 따라 공격에 사용하기 위해 이 침입 단체에 사용할 수 있는 리소스를 결정하는 조직 수준을 정의한다. 이는 개방형 어휘이며 값은 attack-resource-level-ov 어휘에서 가져오는 것이 바람직하다.
primary_motivation (선택사항)	open-vocab	이 침입 단체의 이면에 존재하는 주된 이유, 동기 또는 목적. 동기는 침입 단체가 목표를 달성하려고 하는 이유(달성하고자 하는 목표의 내용)이다. 이는 개방형 어휘이며 값은 attack-motivation-ov 어휘에서 가져오는 것이 바람직하다.
secondary_motivations (선택 사항)	open-vocab의 list	이 침입 단체의 이면에 존재하는 2차적 이유, 동기 또는 목적. 이러한 동기는 주된 동기와 같거나 거의 같은 원인으로 존재할 수 있다. 이는 개방형 어휘이며 값은 attack-motivation-ov 어휘에서 가져오는 것이 바람직하다.

Ⅰ-5.2.7 멀웨어(Malware)

참고: STIX 2.0의 멀웨어 객체는 스텁(stub)이다. 기본적인 사용 사례를 지원하기 위해 포함되었지만 실제 멀웨어 분석, 또는 심지어 단순한 멀웨어 인스턴스 데이터를 포함하는 데에도 유용하지 않을 가능성이 있다. STIX 2의 미래 버전은 이러한 기능을 포함하도록 확장될 것이다.

멀웨어는 description 속성을 통해 멀웨어 샘플과 계열을 특성화하고 식별 및 분류한다. 이렇게 해서 멀웨어가 작동하는 원리와 그것이 수행하는 일에 관한 자세한 정보를 제공한다. 멀웨어에서 시작하는 관계는 멀웨어가 목표로 삼는 대상을 포착하고 그것을 해당 멀웨어의 변형인 다른 멀웨어 객체에 연결할 수 있다.

<표 Ⅰ-5.2-7> 멀웨어 속성

공통 속성		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
멀웨어 고유 속성		
name, description, kill_chain_phases		
속성 이름	형식	설명
type(필수)	string	이 속성의 값은 malware이어야 한다.
labels(필수)	open-vocab의 list	설명하는 멀웨어의 형식. 이는 개방형 어휘이며 값은 malware-label-ov 어휘에서 가져오는 것이 바람직하다.
name(필수)	string	멀웨어 샘플을 식별하는 데 사용되는 이름.
description(선택사항)	string	멀웨어에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.
kill_chain_phases (선택사항)	kill-chain-phase 형식의 list	이 멀웨어가 사용될 수 있는 킬 체인 단계의 목록.

Ⅰ-5.2.8 관측 데이터(Observed Data)

관측 데이터는 본 표준의 연계 표준인 STIX 파트 3과 4에 정의된 Cyber Observable Objects 스펙을 사용하여 시스템과 네트워크에 대해 관측한 정보를 전달한다. 예를 들어 관측 데이터는 IP주소, 네트워크 연결, 파일 또는 레지스트리 키의 관측 정보를 포착할 수 있다. 관측 데이터는 인텔리전스에 대한 단정이 아니라 단지 정보일 뿐이다. 즉, 이 파일은 그것이 의미하는 대상에 대한 아무 컨텍스트가 없어도 발견된 것이다.

<표 Ⅰ-5.2-8> 관측 데이터 속성

공통 속성
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings
관측 데이터 고유 속성
first_observed, last_observed, number_observed, objects

속성 이름	형식	설명
type(필수)	string	이 속성의 값은 observed-data이어야 한다.
first_observed(필수)	timestamp	데이터가 관측되는 시간 범위의 시작.
last_observed(필수)	timestamp	데이터가 관측되는 시간 범위의 끝.
number_observed(필수)	integer	objects 속성에 나타난 데이터가 관측된 횟수. 이는 1부터 999,999,999(포함) 사이의 정수이어야 한다. number_observed 속성이 1보다 크면 objects 속성에 포함된 데이터가 여러 번 관측된 것이다. 이러한 경우 객체 작성자는 해당 관측 데이터의 단일 인스턴스에 고유한 사이버 관측 가능 객체(타임스탬프 등)의 속성을 생략할 수 있다.
objects(필수)	observable-objects	관측 대상을 표현하는 사이버 관측 가능 객체의 사전. 이 사전은 적어도 객체 한 개를 포함해야 한다. observable-objects 형식은 STIX™ Version 2.0. Part 3: Cyber Observable Core Concepts에 정의된다.

Ⅰ-5.2.9 리포트(Report)

보고서는 위협 행위자, 멀웨어 또는 컨텍스트와 관련 세부정보를 포함한 공격 기법에 대한 설명과 같이 하나 이상의 주제에 초점을 맞춘 위협 인텔리전스의 모음이다. 관련 위협 인텔리전스들을 포괄적인 사이버 위협 스토리로 게시할 수 있도록 그룹화하기 위해 사용된다. 보고서 SDO는 보고서에 대한 텍스트 설명 및 이름과 함께 SDO와 SRO(보고서에 포함된 CTI 객체)에 대한 참조의 목록을 포함하고 있다.

<표 Ⅰ-5.2-9> 리포트 속성

<table>
<tr><th colspan="3">공통 속성</th></tr>
<tr><td colspan="3">type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings</td></tr>
<tr><th colspan="3">보고서 고유 속성</th></tr>
<tr><td colspan="3">name, description, published, object_refs</td></tr>
<tr><th>속성 이름</th><th>형식</th><th>설명</th></tr>
<tr><td>type(필수)</td><td>string</td><td>이 속성의 값은 report이어야 한다.</td></tr>
<tr><td>labels(필수)</td><td>open-vocab의 list</td><td>이 속성은 이 보고서의 주된 주제를 지정하는 개방형 어휘이다.
이는 개방형 어휘이며 값은 report-label-ov 어휘에서 가져오는 것이 바람직하다.</td></tr>
<tr><td>name(필수)</td><td>string</td><td>보고서를 식별하는 데 사용되는 이름.</td></tr>
<tr><td>description(선택사항)</td><td>string</td><td>보고서에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.</td></tr>
<tr><td>published(필수)</td><td>timestamp</td><td>이 보고서 객체가 이 보고서의 작성자에 의해 공식적으로 게시된 날짜.
게시 날짜(보도 자료, 법적 발표문 등)는 보고서가 작성되거나 내부적으로 공유된 날짜(created 속성의 날짜)와 다를 수 있다.</td></tr>
<tr><td>object_refs(필수)</td><td>identifier 형식의 list</td><td>이 보고서에 의해 참조된 STIX 객체를 지정한다.</td></tr>
</table>

Ⅰ-5.2.10 위협 행위자(Threat Actor)

위협 행위자는 악의적인 의도를 가지고 운영된다고 판단되는 실제 개인, 그룹 또는 조직이다. 위협 행위자는 침입 단체가 아니라 시간이 경과함에 따라 여러 침입 단체, 그룹 또는 조직을 지원하거나 그들과 연합할 수 있다.

위협 행위자는 자신의 리소스 및 침입 단체의 리소스를 활용하여 공격을 감행하고 대상에 대해 캠페인을 실행한다. 위협 행위자는 자신의 동기, 능력, 목적, 정교화 수준, 과거 활동, 접근할 수 있는 리소스 및 조직 내에서의 역할에 따라 특성화될 수 있다.

<표 Ⅰ-5.2-10> 위협 행위자 속성

공통 속성		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
위협 행위자 고유 속성		
name, description, aliases, roles, goals, sophistication, resource_level, primary_motivation, secondary_motivations, personal_motivations		
속성 이름	형식	설명
type(필수)	string	이 속성의 값은 threat-actor이어야 한다.
labels(필수)	open-vocab의 list	이 속성은 위협 행위자의 형식을 지정한다. 이는 개방형 어휘이며 값은 threat-actor-label-ov 어휘에서 가져오는 것이 바람직하다.
name(필수)	string	이 위협 행위자 또는 위협 행위자 그룹을 식별하는 데 사용되는 이름.
description(선택사항)	string	위협 행위자에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.
aliases(선택사항)	string 형식의 list	이 위협 행위자가 사용한다고 여겨지는 다른 이름의 목록.
roles(선택사항)	open-vocab의 list	위협 행위자가 하는 역할의 목록. 이는 개방형 어휘이며 값은 threat-actor-label-ov 어휘에서 가져오는 것이 바람직하다.
goals(선택사항)	string 형식의 list	이 위협 행위자의 고급 목적, 즉 수행하려고 시도하는 목적.
sophistication(선택사항)	open-vocab	위협 행위자가 공격을 수행하기 위해 가지고 있어야 하는 기량, 구체적 지식, 특수 훈련 또는 전문지식. 이는 개방형 어휘이며 값은 threat-actor-sophistication-ov 어휘에서 가져오는 것이 바람직하다.
resource_level(선택사항)	open-vocab	이 위협 행위자가 일반적으로 일하고, 그에 따라 공격에 사용하기 위해 이 위협 행위자에 사용할 수 있는 리소스를 결정하는 조직 수준을 정의한다. 이 특성은 sophistication 속성에 연결된다. 특정 리소스 레벨은 위협 행위자가 적어도 특정 정교화 수준에 접근할 수 있다는 것을 암시한다. 이는 개방형 어휘이며 값은 attack-resource-level-ov 어휘에서 가져오는 것이 바람직하다.
primary_motivation(선택사항)	open-vocab	이 위협 행위자의 이면에 존재하는 주된 이유, 동기 또는 목적. 동기는 위협 행위자가 목표를 달성하려고 하는 이유(달성하고자 하는 목표의 내용)이다. 이는 개방형 어휘이며 값은 attack-motivation-ov 어휘에서 가져오는 것이

		바람직하다.
secondary_motivations(선택사항)	open-vocab의 list	이 위협 행위자의 이면에 존재하는 2차적 이유, 동기 또는 목적. 이는 개방형 어휘이며 값은 attack-motivation-ov 어휘에서 가져오는 것이 바람직하다.
personal_motivations(선택사항)	open-vocab의 list	조직의 목적과 상관없이 위협 행위자의 개인적 이유, 동기 또는 목적. 이는 개방형 어휘이며 값은 attack-motivation-ov 어휘에서 가져오는 것이 바람직하다.

Ⅰ-5.2.11 도구(Tool)

도구는 위협 행위자가 공격을 수행하기 위해 사용할 수 있는 합법적인 소프트웨어이다. 위협 행위자가 그러한 도구를 사용하는 방법과 시기를 아는 것은 캠페인이 실행되는 방법을 이해하기 위해 중요할 수 있다. 멀웨어와 달리 이러한 도구 또는 소프트웨어 패키지는 흔히 시스템에서 발견되며 고급 사용자, 시스템 관리자, 네트워크 관리자 또는 심지어 일반 사용자에 대한 합법적인 목적을 가지고 있다. 원격 액세스 도구(예: RDP) 및 네트워크 스캔 도구(예: Nmap)는 위협 행위자가 공격 중에 사용할 수 있는 도구의 예이다. 도구 SDO는 이러한 소프트웨어 도구의 속성을 특성화하며 위협 행위자가 공격 중에 이들을 사용하는 방법에 관하여 단정하는 근거로 사용될 수 있다.

<표 Ⅰ-5.2-11> 도구 속성

공통 속성		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
도구 고유 속성		
name, description, kill_chain_phases, tool_version		
속성 이름	형식	설명
type(필수)	string	이 속성의 값은 tool이어야 한다.
labels(필수)	open-vocab의 list	설명하는 도구의 종류. 이는 개방형 어휘이며 값은 tool-label-ov 어휘에서 가져오는 것이 바람직하다.
name(필수)	string	도구를 식별하는 데 사용되는 이름.
description (선택사항)	string	도구에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.
kill_chain_phases (선택사항)	kill-chain-phase 형식의 list	이 도구가 사용될 수 있는 킬 체인 단계의 목록.
tool_version (선택사항)	string	도구와 연결된 버전 식별자.

Ⅰ-5.2.12 취약점(Vulnerability)

취약점은 "해커가 시스템 또는 네트워크에 대한 액세스 권한을 얻기 위해 직접 사용할 수 있는 소프트웨어의 실수"이다. 예를 들어 멀웨어의 한 부분이 CVE-2015-12345를 악용한다면 멀웨어 객체가 CVE-2015-12345를 참조하는 취약점 객체에 연결될 수 있다.

취약점 SDO는 주로 취약점에 대한 외부 정의에 연결하거나 아직 외부 정의를 가지고 있지 않은 제로데이 취약점을 설명하기 위해 사용된다. 일반적으로 다른 SDO는 특정 취약점이 대상이 되어 악의적인 사이버 활동의 일부로 악용되는 경우 취약점 객체와의 관계로 연결된다. 그런 점에서 취약점 객체를 자산 관리 및 규정 준수 프로세스에 대한 연결고리로 사용할 수 있다.

<표 I-5.2-12> 취약점 속성

공통 속성		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
취약점 고유 속성		
name, description		
속성 이름	형식	설명
type(필수)	string	이 속성의 값은 vulnerability이어야 한다.
external_references (선택사항)	external-reference 형식의 list	STIX가 아닌 정보를 참조하는 외부 참조의 목록. 이 속성은 CVE ID 같은 하나 이상의 취약점 ID를 제공하는 데 사용할 수 있다.
name(필수)	string	취약점을 식별하는 데 사용되는 이름.
description(선택사항)	string	취약점에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.

I -5.3 STIX 관계 객체

STIX 관계 객체(SRO)는 CTI를 설명하는 데 사용되는 관계의 형식을 표현한다. 일반 관계 SRO는 다양한 관계 형식을 설명하는데 사용되는 반면, 특정 발견 SRO는 발견 관계를 표현하기 위한 추가 속성을 포함하고 있다.

본 표준에 정의한 각 SRO에 속성 정보, 관계 정보 및 예제가 제공된다. 속성 정보에는 공통 속성과 함께 각 SRO 특유의 속성도 포함된다. SRO는 다른 SRO의 소스 또는 대상일 수 없기 때문에 관계 정보는 포함되지만 포함된 관계(예: created_by_ref)를 설명하기 위해서만 포함된다.

I -5.3.1 관계(Relationship)

관계 객체는 두 SDO가 서로 관련된 방법을 설명하기 위해 이들을 서로 연결하는데 사용된다. SDO가 그래프의 "노드" 또는 "정점"이라고 생각하면 관계 객체(SRO)는 "에지"에 해당한다.

STIX는 SDO들을 연결하기 위해 많은 관계 형식을 정의한다. 일관성을 확보하기 위해 표준에 정의된 관계 형식을 사용하는 것이 바람직하다. 관계의 한 예는 인디케이터가 캠페인을 나타내는(indicates) 것이다. 관계 형식은 인디케이터 SDO 정의의 관계 섹션에 열거된다. 또한 STIX를 사용하여 어떤 SDO에서 본 표준에 정의되지 않은 SDO로 향하는 관계를 정의할 수 있다. 이러한 관계는 related-to 관계 형식을 사용할 수도 있고 사용자 지정 관계 형식을 사용할 수도 있다.

<표 I-5.3-1> 관계 속성

공통 속성		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
관계 고유 속성		
relationship_type, description, source_ref, target_ref		
속성 이름	형식	설명
type(필수)	string	이 속성의 값은 relationship이어야 한다.
relationship_type(필수)	string	관계 형식을 식별하는 데 사용되는 이름. 이 값은 소스와 대상 SDO에 대한 관계에 열거된 정확한 값인 것이 바람직하지만 임의의 문자열일 수 있다. 이 속성의 값은 ASCII이어야 하며 문자 a~z(소문자 ASCII), 0~9 및 대시(-)로 제한된다.
description(선택사항)	string	관계에 관한 더 자세한 사항과 컨텍스트를 제공하는 설명이며 목적과 주요 특성을 포함할 수 있다.
source_ref(필수)	identifier	소스(출발) 객체의 id. 값은 SDO에 대한 ID 참조이어야 한다(즉, SDO, 번들 또는 표시 정의를 가리킬 수 없음).
target_ref(필수)	identifier	대상(목적지) 객체의 id. 값은 SDO에 대한 ID 참조이어야 한다(즉, SDO, 번들 또는 표시 정의를 가리킬 수 없음).

<표 I-5.3-2> 사양 정의 관계 요약

소스	형식	대상	소스	형식	대상
attack-pattern	targets	vulnerability	intrusion-set	attributed-to	threat-actor
attack-pattern	targets	identity	intrusion-set	targets	identity
attack-pattern	uses	malware	intrusion-set	targets	vulnerability
attack-pattern	uses	tool	intrusion-set	uses	attack-pattern
campaign	attributed-to	intrusion-set	intrusion-set	uses	malware
campaign	attributed-to	threat-actor	intrusion-set	uses	tool
campaign	targets	identity	malware	targets	identity
campaign	targets	vulnerability	malware	targets	vulnerability
campaign	uses	attack-pattern	malware	uses	tool
campaign	uses	malware	malware	variant-of	malware
campaign	uses	tool	threat-actor	attributed-to	identity
course-of-action	mitigates	attack-pattern	threat-actor	impersonates	identity
course-of-action	mitigates	malware	threat-actor	targets	identity
course-of-action	mitigates	tool	threat-actor	targets	vulnerability
course-of-action	mitigates	vulnerability	threat-actor	uses	attack-pattern
indicator	indicates	attack-pattern	threat-actor	uses	malware
indicator	indicates	campaign	threat-actor	uses	tool
indicator	indicates	intrusion-set	tool	targets	identity
indicator	indicates	malware	tool	targets	vulnerability
indicator	indicates	threat-actor			
indicator	indicates	tool			

Ⅰ-5.3.2 발견(Sighting)

발견은 대상으로 삼는 사람과 대상 및 공격이 수행된 방법을 추적하고 공격 행위의 추세를 추적하기 위해 사용된다.

발견 관계 객체는 특수한 형식의 SRO이며, 일반 관계 객체에 존재하지 않는 추가 속성을 포함하고 있는 관계이다. 이러한 추가 속성은 발견 관계 고유의 데이터(예: 무언가가 발견된 횟수를 나타내는 count)를 표현하기 위해 포함되지만, 다른 목적을 위해서라면 발견은 "~의 발견"이라는 이름을 가진 관계라고 생각할 수 있다. 발견은 무엇이 발견되었는지에 대한 관계가 있어야만 의미를 갖는다.

<표 Ⅰ-5.3-3> 발견 속성

<table>
<tr><th colspan="3">공통 속성</th></tr>
<tr><td colspan="3">type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings</td></tr>
<tr><th colspan="3">발견 고유 속성</th></tr>
<tr><td colspan="3">first_seen, last_seen, count, sighting_of_ref, observed_data_refs, where_sighted_refs, summary</td></tr>
<tr><th>속성 이름</th><th>형식</th><th>설명</th></tr>
<tr><td>type(필수)</td><td>string</td><td>이 속성의 값은 sighting이어야 한다.</td></tr>
<tr><td>first_seen(선택사항)</td><td>timestamp</td><td>sighting_of_ref 속성에 의해 참조한 SDO가 발견된 시간 범위의 시작.</td></tr>
<tr><td>last_seen(선택사항)</td><td>timestamp</td><td>sighting_of_ref 속성에 의해 참조한 SDO가 발견된 시간 범위의 끝.</td></tr>
<tr><td>count(선택사항)</td><td>integer</td><td>이는 0부터 999,999,999(포함) 사이의 정수이어야 하며 sighting_of_ref 속성에 의해 참조한 SDO가 발견된 횟수를 나타낸다.</td></tr>
<tr><td>sighting_of_ref(필수)</td><td>identifier</td><td>발견된 SDO(예: 인디케이터 또는 멀웨어)에 대한 ID 참조.
이 속성은 SDO 또는 사용자 지정 객체만 참조해야 한다.</td></tr>
<tr><td>observed_data_refs(선택사항)</td><td>identifier 형식의 list</td><td>이 발견에 대한 원시 사이버 데이터를 포함하고 있는 관측 데이터 객체에 대한 ID 참조 목록. 이 속성은 관측 데이터 SDO만 참조해야 한다.</td></tr>
<tr><td>where_sighted_refs(선택사항)</td><td>identifier 형식의 list</td><td>발견을 확인한 엔터티의 ID(피해자) 객체에 대한 ID 참조의 목록.
이 속성은 ID SDO만 참조해야 한다.</td></tr>
<tr><td>summary(선택사항)</td><td>boolean</td><td>summary 속성은 발견을 요약 데이터로 간주해야 하는지 여부를 나타낸다. 요약 데이터는 이전 발견 보고서의 집계이며 기본 소스 데이터로 간주하지 않아야 한다. 기본값은 false이다.</td></tr>
</table>

발견 관계는 발견의 다음 세 가지 측면을 연결시킨다.

- 발견된 것, 즉 인디케이터, 멀웨어, 캠페인 또는 다른 SDO(sighting_of_ref) 등
- 발견한 사람 및/또는 발견된 위치, ID(where_sighted_refs)에 의해 표현됨, 그리고
- 시스템 및 네트워크에서 실제로 발견된 것, 관측 데이터(observed_data_refs)로 표현됨

발견은 무엇이 발견되었는지는 포함되는 것이 필수이다. 발견한 사람, 발견된 장소 및 실제로 발견된 것은 선택 사항이다. 많은 경우 가치를 제공하기 위해서 그 정도의 세부적인 수준을 제공할 필요는 없다.
발견은 특정 SDO가 "발견되었을" 때마다 사용된다. 경우에 따라 객체 작성자는 발견에 관한 매우 일부의 정보를 전달하고자 할 수 있지만(세부 정보가 민감할 겨우) 멀웨어 인스턴스 또는 위협 행위자를 발견하는 것은 중요할 수 있고, 이에 대한 세부정보를 제공하면 유용할 수 있다. 예를 들어 인디케이터 내의 IP주소 1000개 중 발견된 수를 정확히 말하는 것은 해당 IP 중 여전히 악의적인 것을 추적할 때 유용하다.

Ⅰ-5.4 적합성

Ⅰ-5.4.1 객체 생산자(Object Producers)

Ⅰ-5.2(STIX 도메인 객체) 또는 Ⅰ-5.3(STIX 관계 객체)의 객체를 만드는 "STIX 2.0 생산자"는 해당 객체의 "생산자"이다. 객체 생산자는 해당 객체의 모든 표준 요구사항을 준수해야 한다.
예를 들어 인디케이터를 생산할 수 있는 "STIX 2.0 생산자"는 "인디케이터 생산자"이다. 이 생산자는 섹션 2.5, 인디케이터에 나오는 모든 표준 요구사항을 준수해야 한다.

Ⅰ-5.4.2 객체 소비자(Object Consumers)

Ⅰ-5.2(STIX 도메인 객체) 또는 Ⅰ-5.3(STIX 관계 객체)의 객체를 수신하는 "STIX 2.0 소비자"는 해당 객체의 "소비자"이다. 객체 소비자는 해당 객체의 모든 표준 요구사항을 준수해야 한다.
예를 들어 캠페인을 수신할 수 있는 "STIX 2.0 소비자"는 "캠페인 소비자"이다. 이 소비자는 Ⅰ-5.2.2, 캠페인에 나오는 모든 표준 요구사항을 준수해야 한다.

부 록 Ⅰ-6

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

표준의 이력

판수	채택일	표준번호	내용	담당 위원회
제1판	2018.XX.XX	제정 TTAE.OT-xx.xxxx	-	사이버보안 프로젝트 그룹 (PG503), 정보보호 기술위원회(TC5)