STIX 2.0 Specification - Pre-Draft

TLO - TTPs - Version 0.1

Document Table of Contents

1. Document Development Status

Object / Concept	Status	MVP	Description
`ttp` (abstract)	Concept	Undecided	TTPs represent the tactics, techniques, and procedures that are used to carry out attacks. TTPs are a type of top-level object: the attack pattern, exploit, infrastructure, malware, persona, tool, and victim targeting top-level objects are all types of TTPs. *As an abstract superclass,* `ttp` *cannot be directly instantiated*.
`attack-pattern`	Concept	Undecided	A pattern of activity (attack pattern) used to carry out attacks.
`exploit`	Concept	Undecided	A set of commands that leverages a vulnerability or configuration setting (exploit target) to cause a system to behave in an unintended way.
`kill-chain`	Concept	Undecided	Typical steps that cyber threat actors use to carry out their attacks.
`kill-chain-phase`	Concept	Undecided	A single phase of a Kill Chain process that cyber threat actors follow to carry out their attacks.
`malicious-infrastructure`	Concept	Undecided	Infrastructure leveraged by cyber threat actors to carry out attacks.
`malicious-tool`	Concept	Undecided	Software designed with a malicious purpose that is used directly by an attacker.

`malware`	Concept	Undecided	Software designed with a malicious purpose that is installed without the user of the system being aware.
`persona`	Concept	Undecided	A description of the assumed representation by a Threat Actor.

2. Top Level Objects

2.1. TTP

Type Name: `ttp`	**Status:** Concept **MVP**: Undecided

In cyber threat intelligence, TTPs represent the tactics, techniques, and procedures that are used to carry out attacks. TTPs are a type of top-level object: the attack pattern, exploit, infrastructure, malware, persona, tool, and victim targeting top-level objects are all types of TTPs.

`ttp` can be considered an “abstract superclass” of the individual TTP types listed below. Common properties that apply to all TTPs are defined on TTP and inherited by the individual TTP types.

As an abstract superclass, `ttp` *cannot be directly instantiated*.

2.1.1. Properties

Inherits From	**Inherited Properties**	
`stix-core`	**`type, id, spec_version, created_time, created_by_ref, object_markings_refs, granular_markings`**	
`descriptive-properties`	**`title, description`**	
Property Name	**Type**	**Description**

impact (optional)	`impact`	The impact on operations if this ttp were to be used.

3. TTP Objects

3.1. Attack Pattern

Type Name: `attack-pattern`	**Status:** Concept **MVP**: Undecided

Attack Pattern describes a general approach for attacking a system or network. In addition to the standard TTP properties, it has a reference to a CAPEC (Common Attack Pattern Enumeration and Classification - http://capec.mitre.org/) ID.

Open Questions:

1. Jason K. had a proposal for all these external IDs to use our standard external IDs structure. That was back when external IDs were actually on the object though, now that you have to use a relationship it would be 2 additional TLOs just to say the CAPEC ID.

3.1.1. Properties

Inherits From	**Inherited Properties**	
`stix-core`	**type, id, spec_version, created_time, created_by_ref, object_markings_refs, granular_markings**	
`descriptive-properties`	**title, description**	
`ttp`	**impact**	
Property Name	**Type**	**Description**
type (required)	`string`	(Overrides `stix-core`) The value of this field **MUST** be `attack-pattern`
capec_id (optional)	`string`	Specifies a reference to an entry in the CAPEC dictionary.

3.1.2. Relationships

These are the default relationships between the Attack Pattern Object and other objects.

Inherited From		Inherited Kinds of Relationships
`stix-core`		`derived-from`, `duplicate-of`, `suggested-update`, `related-to`
Kind of Relationship	**Target**	**Description**
`used-by`	`threat-actor`	Relates the Attack Pattern as one being used by the Threat Actor to perform attacks.
`used-in`	`campaign`	Relates the Attack Pattern to a Campaign to enable tagging it as being an attack pattern performed as part of a Campaign.
`suggested-coa-of`	`course-of-action`	Relates the Attack Pattern to a Course of Action to allow Organizations to protect themselves against the threat described by this Attack Pattern.
`uses-exploit`	`exploit`	Relates the Attack Pattern to an exploit that can exploit the vulnerability/misconfiguration or weakness.
`in-kill-chain-phase`	`kill-chain-phase`	Relates the Attack Pattern to the phase of the Kill Chain it is used within.
`evidenced-by`	`observation`	Relates the Attack Pattern to an Observation that demonstrates the attack pattern described in this Object.

3.2. Exploit

Type Name: `exploit`	**Status:** Concept **MVP**: Undecided

Exploit describes a cyber threat exploit: a set of commands that leverages a vulnerability or misconfiguration (exploit target) in order to cause unintended behavior in the targeted software.

3.2.1. Properties

Inherits From	Inherited Properties	
`stix-core`	**`type`, `id`, `spec_version`, `created_time`, `created_by_ref`, `object_markings_refs`, `granular_markings`**	
`descriptive-properties`	**`title`, `description`**	
`ttp`	`impact`	
Property Name	**Type**	**Description**
`type` (required)	`string`	(Overrides `stix-core`) The value of this field **MUST** be `exploit`

3.2.2. Relationships

These are the default relationships between the Exploit Object and other objects.

Inherited From		Inherited Kinds of Relationships
`stix-core`		`derived-from`, `duplicate-of`, `suggested-update`, `related-to`
Kind of Relationship	**Target**	**Description**
`used-by`	`threat-actor`	Relates the Exploit as one being used by the Threat Actor to perform attacks.
`used-in`	`campaign`	Relates the Exploit to a Campaign to enable tagging it as being an Exploit used as part of a Campaign.
`suggested-coa-of`	`course-of-action`	Relates the Exploit to a Course of Action to allow Organizations to protect themselves against the threat described by this Exploit.

`uses-exploit`	`attack-pattern`	Relates the Exploit to an Attack Pattern that describes what it does.
`in-kill-chain-phase`	`kill-chain-phase`	Relates the Exploit to the phase of the Kill Chain it is used within.
`evidenced-by`	`observation`	Relates the Exploit to an Observation that demonstrates the exploit described in this Object.
`exploits`	`configuration`, `vulnerability`, `weakness`	Relates the Exploit to the vulnerability, misconfiguration, or weakness that it leverages to conduct the attack.

3.3. Kill Chain

Type Name: `kill-chain`	**Status:** Concept **MVP**: Undecided

Kill chain describes the typical steps that attackers use to carry out their objectives. One popular example of a kill chain is the Lockheed Martin kill chain. Each phase of the Kill Chain is described in a separate object called a Kill Chain Phase. The distinction was made between Kill Chains and Kill Chain Phases to enable exploits to refer directly to individual phases of a kill chain.

3.3.1. Properties

Inherits From	Inherited Properties	
`stix-core`	**`type`, `id`, `spec_version`, `created_time`, `created_by_ref`, `object_markings_refs`, `granular_markings`**	
`descriptive-properties`	**`title`, `description`**	
`ttp`	**`impact`**	
Property Name	**Type**	**Description**

type (required)	`string`	(Overrides `stix-core`) The value of this field **MUST** be `kill-chain`

3.3.2. Relationships

These are the default relationships between the Kill Chain Object and other objects.

Inherited From		Inherited Kinds of Relationships
`stix-core`		`derived-from`, `duplicate-of`, `suggested-update`, `related-to`
Kind of Relationship	**Target**	**Description**
`has-kill-chain-phase`	`kill-chain-phase`	Relates the Kill Chain to the Kill Chain Phase.
`evidenced-by`	`observation`	Relates the Kill Chain to an Observation providing evidence that backs up the assertions provided in this Object.

3.4. Kill Chain Phase

Type Name: `kill-chain-phase`	**Status:** Concept **MVP**: Undecided

Kill chain phase describes an individual phase of a Kill Chain that attackers use to carry out their objectives. One popular example of a kill chain is the Lockheed Martin kill chain. This Kill Chain Phase describes one phase of a Kill Chain. The distinction was made between Kill Chains and Kill Chain Phases to enable exploits to refer directly to individual phases of a kill chain.

3.4.1. Properties

Inherits From	Inherited Properties

`stix-core`	**`type, id, spec_version, created_time, created_by_ref, object_markings_refs, granular_markings`**	
`descriptive-properties`	**`title, description`**	
`ttp`	**`impact`**	
Property Name	**Type**	**Description**
`type` (required)	`string`	(Overrides `stix-core`) The value of this field **MUST** be `kill-chain-phase`

3.4.2. Relationships

These are the default relationships between the Kill Chain Phase Object and other objects.

Inherited From		**Inherited Kinds of Relationships**
`stix-core`		`derived-from`, `duplicate-of`, `suggested-update`, `related-to`
Kind of Relationship	**Target**	**Description**
`indicates`	`threat-actor`	Relates the Kill Chain Phase to a Threat Actor who operates within that phase of the Kill Chain.
`used-by-exploit`	`exploit`	Relates the Kill Chain Phase to an exploit that uses it.
`in-kill-chain`	`kill-chain`	Relates the Kill Chain Phase to the phase of the Kill Chain it belongs to. Requires an extension with the name `ordinality` to indicate the ordinality of the phase within the Kill Chain.
`evidenced-by`	`observation`	Relates the Kill Chain Phase to an Observation providing evidence that backs up the assertions provided in this Object.

3.5. Malicious Infrastructure

Type Name: `malicious-infrastructure`	**Status:** Concept **MVP**: Undecided

Infrastructure describes infrastructure leveraged by cyber threat actors, such as command and control servers, domain registration, botnets, or hosting/delivery.

3.5.1. Properties

Inherits From	**Inherited Properties**	
`stix-core`	**`type`, `id`, `spec_version`, `created_time`, `created_by_ref`, `object_markings_refs`, `granular_markings`**	
`descriptive-properties`	**`title`, `description`**	
`ttp`	**`impact`**	
Property Name	**Type**	**Description**
`type` (required)	`string`	(Overrides `stix-core`) The value of this field **MUST** be `malicious-infrastructure`
`kinds_of_malicious_infrastructures` (required)	`array` of type `malicious-infrastructure-kind-cv`	The kind(s) of infrastructure being described.
`kinds_of_malicious_infrastructures_ext` (optional)	`array` of type `vocab-ext`	Specifies alternate values for the **`kinds_of_malicious_infrastructures`** property.

3.5.2. Relationships

These are the default relationships between the Malicious Infrastructure Object and other objects.

Inherited From		Inherited Kinds of Relationships
`stix-core`		`derived-from`, `duplicate-of`, `suggested-update`, `related-to`
Kind of Relationship	**Target**	**Description**
`used-by`	`threat-actor`	Relates the Malicious Infrastructure as one being used by the Threat Actor to perform attacks.
`used-in`	`campaign`	Relates the Malicious Infrastructure to a Campaign to enable tagging it as being used as part of a Campaign.
`uses-exploit`	`exploit`	Relates the Malicious Infrastructure to an Exploit to enable Organizations to quickly find similar Infrastructure that delivers the same exploits.
`uses-asset`	`asset`	Relates the Malicious Infrastructure to an Asset that is part of this malicious infrastructure.
`uses-exploit`	`attack-pattern`	Relates the Malicious Infrastructure to an Attack Pattern that describes what it does.
`in-kill-chain-phase`	`kill-chain-phase`	Relates the Malicious Infrastructure to the phase of the Kill Chain it is used within.
`evidenced-by`	`observation`	Relates the Malicious Infrastructure to an Observation that involves the Malicious Infrastructure described in this Object.

3.6. Malicious Tool

Type Name: `malicious-tool`	**Status:** Concept **MVP**: Undecided

Tool describes a piece of software designed with a malicious purpose that is used directly by an attacker.

3.6.1. Properties

Inherits From	Inherited Properties	
`stix-core`	**`type`, `id`, `spec_version`, `created_time`, `created_by_ref`, `object_markings_refs`, `granular_markings`**	
`descriptive-properties`	**`title`, `description`**	
`ttp`	**`impact`**	
Property Name	**Type**	**Description**
`type` (required)	`string`	(Overrides `stix-core`) The value of this field **MUST** be `malicious-tool`
`kinds_of_malicious_tools` (required)	`array` of type `malicious-tool-kind-cv`	The kind(s) of tool(s) being described.
`kinds_of_malicious_tools_ext` (optional)	`array` of type `vocab-ext`	Specifies alternate values for the **`kinds_of_malicious_tools`** property.
`compensation_model` (optional)	`string`	The type of compensation model used by this tool.
`tool_information` (required)	tool	The description of the tool itself

3.6.2. Relationships

Status: stub

3.7. Malware

Type Name: `malware`	**Status:** Concept **MVP**: Undecided

Malware describes software designed specifically with a malicious purpose by a malicious threat actor often installed without the user of the system being aware. This DOES NOT include software created for legitimate purposes which are then abused by threat actors for their own purposes.

3.7.1. Properties

Inherits From	Inherited Properties	
`stix-core`	**`type`, `id`, `spec_version`, `created_time`, `created_by_ref`, `object_markings_refs`, `granular_markings`**	
`descriptive-properties`	**`title`, `description`**	
`ttp`	**`impact`**	
Property Name	**Type**	**Description**
`kinds_of_malware` (required)	`array` of type `malware-kind-cv`	The kind(s) of malware being described.
`kinds_of_malware_ext` (optional)	`array` of type `vocab-ext`	Specifies alternate values for the **`kinds_of_malware`** property.
`maec` (optional)	*MAEC*	A description of the malware leveraging MAEC.

3.7.2. Relationships

These are the default relationships between a Malware Object and other objects.

Inherited From		Inherited Kinds of Relationships
`stix-core`		`derived-from`, `duplicate-of`, `suggested-update`, `related-to`
Kind of Relationship	**Target**	**Description**
`used-by`	`threat-actor`	Relates the Malware as one being used by the Threat Actor to perform attacks.

`created-by`	`threat-actor`	Relates the Malware to a Threat Actor used to create the malware.
`used-in`	`campaign`	Relates the Malware to a Campaign to enable tagging it as being used as part of a Campaign.
`uses-exploit`	`exploit`	Relates the Malware to an Exploit to enable Organizations to quickly find similar Malware that uses the same exploits.
`uses-asset`	`asset`	Relates the Malware to an Asset that is delivered by or communicates with this Asset.
`uses-malicious-infrastructure`	`malicious-infrastructure`	Relates the Malware to an Asset that is delivered by or communicates with this malicious infrastructure.
`uses-attack-pattern`	`attack-pattern`	Relates the Malware to an Attack Pattern that describes what it does.
`in-kill-chain-phase`	`kill-chain-phase`	Relates the Malware to the phase of the Kill Chain it is used within.
`evidenced-by`	`observation`	Relates the Malware to an Observation that demonstrates activity associated with the malware described in this Object.

3.8. Persona

Type Name: `persona`	**Status:** Concept **MVP**: Undecided

<enter description>

Open Questions:

1. Is this a separate TLO, or is it a relationship (uses-persona) from a threat actor to an identity?
2. Regardless of how persona is defined, it needs to be a many-to-many relationship between threat actors and personas.

3.8.1. Properties

Inherits From	Inherited Properties	
`stix-core`	**`type, id, spec_version, created_time, created_by_ref, object_markings_refs, granular_markings`**	
`descriptive-properties`	**`title, description`**	
`ttp`	**`impact`**	
Property Name	**Type**	**Description**
`type` (required)	`string`	(Overrides `stix-core`) The value of this field **MUST** be `persona`

3.8.2. Relationships

Status: stub

3.9. Victim Targeting

Type Name: `victim-targeting`	**Status:** Concept **MVP**: Undecided

Victim targeting describes the types of victims targeted by a particular threat. This includes identity-based targeting (by sector, company, country, etc.), system-based targeting (web systems, etc), or information type-based targeting (credentials, PII, trade secrets, etc.).

Open Questions:

- How much of this should be relationships to assets, identities, etc? Can you represent combinations of these (HR information in energy sector) via relationships?

3.9.1. Properties

Inherits From	Inherited Properties

stix-core	**type, id, spec_version, created_time, created_by_ref, object_markings_refs, granular_markings**	
descriptive-properties	**title, description**	
ttp	impact	
Property Name	**Type**	**Description**
type (required)	string	The value of this field MUST be victim-targeting
targets (required)	array of type identity-target, system-target, information-target	The list of targets.

3.9.2. Relationships

Status: stub