Overview

Cisco ScanCenter allows you to pull information on incidents detected by CTA down to your client for further correlation analysis and archival. The service supports MITRE's Trusted Automated eXchange of Indicator Information (TAXII) standard for integration with your Security Information and Event Management (SIEM) system. The TAXII standard specifies transport mechanisms used to share cyber threat information between systems.

For more information on TAXII, see:
https://taxii.mitre.org/
http://taxiiproject.github.io/

The information in each incident is represented using the Structured Threat Information eXpression (STIX) language format. STIX is a structured language used to describe cyber threat information so it can be shared, stored, and analyzed in a consistent manner. The STIX format allows CTA to represent its breach detection findings in a hierarchical format. The TAXII service uses a subset of the STIX language to describe the incidents CTA has detected. Currently, the supported objects include:

• Campaign—Confirmed threat category, if available
• Incident—Anomalous activity
• TTP—Tactics, Techniques, and Procedures
• Observable—Web requests
• Indicator—Pattern identifying observable conditions

For more information on STIX, see:
https://stix.mitre.org/
Poll Service

The poll service uses standardized TAXII transport mechanisms to send incident information from CTA to clients that support the TAXII standard. To pull incident information, the TAXII client sends a poll request to the TAXII poll service. HTTP basic authentication is used to restrict access for authorized users only. The TAXII poll service then responds by sending incident information from CTA to the TAXII client. HTTPS protocol is used to secure all data transfers.

Your SIEM or other security work-flow system must natively support STIX/TAXII. Configure your third-party TAXII client to periodically poll the TAXII poll service.

- To obtain your account information, log into Cisco ScanCenter and request CTA STIX/TAXII service.
  - Click the Threats tab.
  - Click the global settings icon in the upper-right corner.
  - Click CTA STIX/TAXII API.
  - Click the Add account button.
  - Enter a name to identify your account, and then click the Add account button.

- After the provisioning process is completed, your account information is displayed. Copy this account information to a secure location before closing the window.

  Note
  For security reasons, the secret password is displayed only once. If you lose the secret password, you must revoke the existing secret password and generate a new secret password.

  • Copy your unique attributes into your third-party TAXII client:
    - pollEndpoint or feed service
      URL=https://taxii.cloudsec.sco.cisco.com/skym-taxii-ws/PollService
    - username
    - password
    - collection name or feed name

Note
In August 2018, Cognitive Intelligence (formerly Cognitive Threat Analytics or CTA) started its migration to a new location in Amazon Web Services, which resulted in new IP addresses and an additional URL to access and use the service. To maintain access to the service, it may be necessary to update your outbound firewall rules. After the switchover in November 2018, you will no longer be able to successfully send data to the old data ingest service IP address. Specific details on the required changes and other important information can be found in the Field Notice.
We do not provide technical support for configuring third-party products or SIEM devices. In the event of an issue, consult the vendor-specific support team.

Alternatively, you may download and use an example TAXII client from Cisco. If your SIEM or other security system does not natively support STIX/TAXII, Cisco provides a lightweight Java TAXII Log Adapter that you can deploy to a Linux or Windows VM environment next to your SIEM. Click the link provided to view setup instructions. The adapter uses the TAXII API to perform regular polling of any new intelligence and delivers data in STIX messages. The STIX messages are then transformed by the adapter into other formats accepted by common SIEM systems.

To support the stability, performance, and availability of the poll service:

- Only one poll request from any single TAXII client is allowed within every 10 minutes. Otherwise, a status message indicating this error is returned.
- Each poll request may retrieve incident information spanning up to three days.
- Incident information is stored for retrieval for up to 30 days.

## Poll Request

The following is an example of a poll request from your TAXII client to the TAXII poll service.

**Method** is POST.

**HTTP Request headers:**

- x-taxii-content-type: urn:taxii.mitre.org:message:xml:1.1
- x-taxii-protocol: urn:taxii.mitre.org:protocol:http:1.1
- x-taxii-services: urn:taxii.mitre.org:services:1.1
- x-taxii-accept: urn:taxii.mitre.org:message:xml:1.1
- content-type: application/xml
- accept: application/xml
- authorization: Basic ...

**Request body:**

```xml
<taxii_11:Poll_Request xmlns:taxii_11="http://taxii.mitre.org/messages/taxii_xml_binding-1.1"
message_id="" collection_name="">
<taxii_11:Exclusive_Begin_Timestamp>2015-01-16T00:00:00+00:00</taxii_11:Exclusive_Begin_Timestamp>
<taxii_11:Inclusive_End_Timestamp>2015-01-17T00:00:00+00:00</taxii_11:Inclusive_End_Timestamp>
<taxii_11:Poll_Parameters allow_asynch="false"/>
<taxii_11:Response_Type>FULL</taxii_11:Response_Type>
</taxii_11:Poll_Parameters>
</taxii_11:Poll_Request>
```

**Supported Request Parameters**

| Poll_Request |
|--------------|-----------------|
| Description  |                 |
### Supported Request Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>message_id</td>
<td>A randomly generated string for each request, according to the TAXII specification. Regenerate a unique string for every request.</td>
</tr>
<tr>
<td>collection_name</td>
<td>Name of collection to extract or pull from the CTA service. This attribute will be provided to you by Cisco after the provisioning process is completed.</td>
</tr>
<tr>
<td>Exclusive_Begin_Timestamp</td>
<td>Adjust this value according to your timeframe.</td>
</tr>
<tr>
<td>Inclusive_End_Timestamp</td>
<td>Adjust this value according to your timeframe.</td>
</tr>
<tr>
<td>Poll_Parameters</td>
<td></td>
</tr>
<tr>
<td>allow_asynch</td>
<td>Always set this attribute to false.</td>
</tr>
</tbody>
</table>

### Note

The maximum supported difference between `Exclusive_Begin_Timestamp` and `Inclusive_End_Timestamp` is three days. In case the difference is more, the returned result is limited to the last three days before `Inclusive_End_Timestamp`.

### Poll Response

The following is an example of a poll response from the TAXII poll service to the TAXII client.

**HTTP Response headers:**

```
x-taxii-content-type: urn:taxii.mitre.org:message:xml:1.1
x-taxii-protocol: urn:taxii.mitre.org:protocol:http:1.1
x-taxii-services: urn:taxii.mitre.org:services:1.1
```

**Response body:**

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<t:Poll_Response xmlns:t="http://taxii.mitre.org/messages/taxii_xml_binding-1.1"
    xmlns:c="http://cybox.mitre.org/cybox-2"
    xmlns:cc="http://cybox.mitre.org/common-2"
    xmlns:co="http://cybox.mitre.org/objects#CustomObject-1"
    xmlns:coa="http://stix.mitre.org/CourseOfAction-1"
    xmlns:sc="http://stix.mitre.org/common-1"
    xmlns:ind="http://stix.mitre.org/Indicator-2"
    xmlns:ttp="http://stix.mitre.org/TTP-1"
    xmlns:inc="http://stix.mitre.org/Incident-1"
    xmlns:s="http://stix.mitre.org/stix-1"
    collection_name=" " more="true"
    result_id="" result_part_number="1"
    in_response_to="generatedMessageID" message_id="responseMessageID">
    <t:Exclusive_Begin_Timestamp>2015-01-17T15:11:00.648Z</t:Exclusive_Begin_Timestamp>
    <t:Inclusive_End_Timestamp>2015-01-20T15:11:00.649Z</t:Inclusive_End_Timestamp>
    <t:Content_Block>
        <t:Content_Binding binding_id="STIX_XML_1.1"/>
        <t:Content>
```
Incident: malware using automatically generated domain (DGA)
Victim: JohnDoe

Related Indicator:
- timestamp: 1421623882432
  - xElapsedTime: 1810
  - scHttpStatus: 0
  - csContentBytes: 622
  - scContentBytes: 907
  - csUrl: 
  - sIP: 195.22.26.231
  - cIP: 33.196.39.11
  - cUsername: JohnDoe
  - sReputation: -580
  - sCategory: unclassified

- timestamp: 1421623896635
  - xElapsedTime: 1942
  - scHttpStatus: 0
  - csContentBytes: 361
  - scContentBytes: 582
  - csUrl: 
  - sIP: 195.22.26.231
  - cIP: 33.196.39.11
  - cUsername: JohnDoe
<cc:Property name="sReputation">-580</cc:Property>
<cc:Property name="sCategory">unclassified</cc:Property>
</c:Custom_Properties>
</c:Properties>
</c:Object>
</c:Observable>
</c:Observable_Composition>
</ind:Observable>
</ind:Indicated_TTP>
</sc:TTP xsi:type="ttp:TTPType">
<ttp:Title>communication to automatically generated domain (DGA)</ttp:Title>
</sc:TTP>
</inc:Indicated_TTP>
</inc:Related_Indicator>
</inc:Related_Indicators>
<inc:Discovery_Method>Log Review</inc:Discovery_Method>
<inc:COA_Requested>
<inc:Course_Of_Action xsi:type="coa:CourseOfActionType">
<coa:Stage>Remedy</coa:Stage>
<coa:Type>Eradication</coa:Type>
<coa:Parameter_Observables cybox_major_version="2" cybox_minor_version="1">
<c:Observable_Package_Source>
<cc:Time>
<cc:Produced_Time>2016-08-15T17:02:02.616Z</cc:Produced_Time>
</cc:Time>
</c:Observable_Package_Source>
</c:Observable>
<c:Observable>
<c:Object>
<c:Properties xsi:type="user:UserAccountObjectType">
<u:Username>JohnDoe</u:Username>
</c:Properties>
</c:Object>
</c:Observable>
</c:Observable>
</coa:Parameter_Observables>
</inc:Course_Of_Action>
</inc:COA_Requested>
<inc:Confidence>
<sc:Value>Low</sc:Value>
</inc:Confidence>
<inc:Information_Source>
<sc:Tools>
<cc:Tool idref="cta:tool-cta"/>
</sc:Tools>
</inc:Information_Source>
</s:Incident>
</s:Incidents>
</s:STIX_Package>
</t:Content>
</t:Content_Block>
</t:Poll_Response>
In **Poll_Response**, if there are no more threat items, the two attributes of *more* and *result_id* are not present. When *more=true* is present, you can request the next pages of the response by using a **Poll_Fulfillment**.

<table>
<thead>
<tr>
<th>Supported Response Objects</th>
<th>Description of Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poll_Response</td>
<td></td>
</tr>
<tr>
<td>collection_name</td>
<td>Name of collection to extract or pull from the CTA service. This attribute will be provided to you by Cisco after the provisioning process is completed.</td>
</tr>
<tr>
<td>result_id</td>
<td>Copy this value to the poll fulfillment request.</td>
</tr>
<tr>
<td>Exclusive_Begin_Timestamp</td>
<td>Exclusive beginning of the time range covered by this poll response. Absence of this field indicates that the poll response covers the earliest time for this TAXII data feed.</td>
</tr>
<tr>
<td>Inclusive_End_Timestamp</td>
<td>Inclusive end of the time range covered by this poll response.</td>
</tr>
<tr>
<td>Content_Block</td>
<td>Returned content.</td>
</tr>
<tr>
<td>Content_Binding</td>
<td></td>
</tr>
<tr>
<td>Content</td>
<td></td>
</tr>
<tr>
<td>STIX_Package</td>
<td>Information about the STIX language.</td>
</tr>
<tr>
<td>STIX_Header</td>
<td>Information about this package of STIX content.</td>
</tr>
<tr>
<td>Incidents</td>
<td>One or more incidents.</td>
</tr>
<tr>
<td>Incident</td>
<td>Information about a single incident.</td>
</tr>
<tr>
<td>Title</td>
<td>Title describing this incident.</td>
</tr>
<tr>
<td>Victim</td>
<td>Information about the victim of this incident.</td>
</tr>
<tr>
<td>Related_Indicators</td>
<td>Identifies indicators related to this incident.</td>
</tr>
<tr>
<td>Related_Indicator</td>
<td>Identifies a single indicator related to this incident.</td>
</tr>
<tr>
<td>Indicator</td>
<td>Indicator made up of a pattern that identifies certain observable conditions as well as contextual information about the pattern's meaning, how and when it should be acted upon, etc.</td>
</tr>
<tr>
<td>Observable</td>
<td>Relevant observable for this indicator.</td>
</tr>
<tr>
<td>Observable_Composition</td>
<td>Enables specifying higher-order composite observables by composing logical combinations of other observables.</td>
</tr>
</tbody>
</table>
### Supported Response Objects

<table>
<thead>
<tr>
<th>Objects</th>
<th>Description of Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Observable</td>
<td>Represents a single observable.</td>
</tr>
<tr>
<td>Object</td>
<td>Identifying characteristics of a specific object (e.g. file, registry key, process)</td>
</tr>
<tr>
<td>Properties</td>
<td>Properties that were enumerated as a result of the action on the object.</td>
</tr>
<tr>
<td>Custom_Properties</td>
<td>Enables specifying a set of custom object properties that may not be defined</td>
</tr>
<tr>
<td></td>
<td>in existing Properties schemas.</td>
</tr>
<tr>
<td>Property</td>
<td>A single property that was enumerated as a result of the action on the object.</td>
</tr>
<tr>
<td>Indicated_TTP</td>
<td>Specifies the relevant Tactics, Techniques, and Procedures (TTP) indicated by this</td>
</tr>
<tr>
<td></td>
<td>indicator.</td>
</tr>
<tr>
<td>Discovery_Method</td>
<td>Information about the method and/or tool used to discover the code.</td>
</tr>
<tr>
<td>COA_Requested</td>
<td>Recommended course of actions for this incident.</td>
</tr>
<tr>
<td>Confidence</td>
<td>Information about the level of confidence held in the characterization of this incident.</td>
</tr>
<tr>
<td>Information_Source</td>
<td>Information about the source of this incident.</td>
</tr>
<tr>
<td>Tools</td>
<td></td>
</tr>
<tr>
<td>Tool</td>
<td>Which tool, CTA or AMP, detected this incident.</td>
</tr>
</tbody>
</table>

---

In case of an error, an error message is returned. For example:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<t:Status_Message
    xmlns:t="http://taxii.mitre.org/messages/taxii_xml_binding-1"
    xmlns:c="http://cybox.mitre.org/cybox-2"
    xmlns:cc="http://cybox.mitre.org/common-2"
    xmlns:co="http://cybox.mitre.org/objects#CustomObject-1"
    xmlns:sc="http://stix.mitre.org/common-1"
    xmlns:ind="http://stix.mitre.org/Indicator-2"
    xmlns:ttp="http://stix.mitre.org/TTP-1"
    xmlns:inc="http://stix.mitre.org/Incident-1"
    xmlns:s="http://stix.mitre.org/stix-1"
    status_type="FAILURE" in_response_to="23537"
    message_id="16ed0b75-2af6-4537-b71c-da00e0a0c419">
    <t:Message>An error occurred during request processing.</t:Message>
</t:Status_Message>
```

### TAXII status_type

<table>
<thead>
<tr>
<th>status_type</th>
<th>Description of Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>User is not authenticated, HTTP response status code of 404</td>
<td></td>
</tr>
<tr>
<td>User is not authorized, HTTP response status code of 401</td>
<td></td>
</tr>
<tr>
<td>TAXII status_type</td>
<td>Description of Error</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>BAD_MESSAGE</td>
<td>Invalid request message, refer to Message parameter</td>
</tr>
<tr>
<td>FAILURE</td>
<td>Unspecified error, refer to Message parameter</td>
</tr>
</tbody>
</table>

### Poll Fulfillment

The following is an example of a poll fulfillment request from your TAXII client to the TAXII poll service. Method is POST.

HTTP Request headers:

```plaintext
x-taxii-content-type: urn:taxii.mitre.org:message:xml:1.1
x-taxii-protocol: urn:taxii.mitre.org:protocol:http:1.1
x-taxii-services: urn:taxii.mitre.org:services:1.1
x-taxii-accept: urn:taxii.mitre.org:message:xml:1.1
content-type: application/xml
accept: application/xml
authorization: Basic ...
```

Request body:

```xml
<taxii_11:Poll_Request
xmlns:taxii_11="http://taxii.mitre.org/messages/taxii_xml_binding-1.1"
message_id=" " collection_name=" "
result_id=" " result_part_number="2" />
```

```xml
<taxii_11:Exclusive_Begin_Timestamp>2015-01-16T00:00:00+00:00</taxii_11:Exclusive_Begin_Timestamp>
```

```xml
<taxii_11:Inclusive_End_Timestamp>2015-01-17T00:00:00+00:00</taxii_11:Inclusive_End_Timestamp>
```

```xml
<taxii_11:Poll_Parameters allow_asynch="false"/>
```

```xml
<taxii_11:Response_Type>FULL</taxii_11:Response_Type>
</taxii_11:Poll_Parameters>
</taxii_11:Poll_Request>
```

### Supported Request Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poll_Request</td>
<td></td>
</tr>
<tr>
<td>message_id</td>
<td>A randomly generated string for each request, according to the TAXII specification. Regenerate a unique string for every request.</td>
</tr>
<tr>
<td>collection_name</td>
<td>Name of collection to extract or pull from the CTA service. This attribute will be provided to you by Cisco after the provisioning process is completed.</td>
</tr>
<tr>
<td>result_id</td>
<td>Paste this value from the poll response.</td>
</tr>
<tr>
<td>result_part_number</td>
<td>Increment this value by 1 from the value in the poll response.</td>
</tr>
<tr>
<td>Exclusive_Begin_Timestamp</td>
<td>Adjust this value according to your timeframe.</td>
</tr>
</tbody>
</table>
### Common Queries

This section describes some common queries used in the Cisco STIX/TAXII API to help prioritize findings for further investigation. The syntax used in the example queries is based on SPLUNK integration and is symbolic. The particular fields and values may differ depending on your local integration, but the meaning of the queries is broadly applicable across SIEM systems and integrations.

If you are collecting other data in SPLUNK, prepend your query with host, index, or source name to search through only CTA data.

#### Users Affected by Confirmed Threats

This query returns all users with confirmed threats and may be reported to your Incident Response Team for desktop remediation. If these incidents are also high risk, consider reimaging the affected device. This query generates a table with usernames and campaign names by which they are affected. Search for nonempty campaign name and then deduplicate username+campaign pairs:

```plaintext
campaign!="" | table cUsername campaign | dedup cUsername campaign | sort + cUsername
```

Alternatively, with multi-value field for campaign name:

```plaintext
campaign!="" | transaction cUsername | table cUsername campaign | sort + cUsername
```

#### Users Affected by Confirmed Threats Within a Timeframe

This query also includes first-seen and last-seen columns. Search for nonempty campaign, aggregate by username+campaign pair, and compute min and max of the web-flow time stamp. Results are in epoch-milliseconds and can be converted to calendar time, if necessary.

```plaintext
campaign!="" | stats min(timestamp) max(timestamp) by cUsername campaign
```

Alternatively, include the epoch conversion using the strftime function. This example divides the timestamp by 1000 to remove milliseconds:

```plaintext
campaign!="" | stats min(timestamp) as oldest max(timestamp) as newest by cUsername campaign
```

---

**Supported Request Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inclusive_End_Timestamp</td>
<td>Adjust this value according to your timeframe.</td>
</tr>
<tr>
<td>Poll_Parameters</td>
<td>Always set this attribute to false.</td>
</tr>
<tr>
<td>allow_asynch</td>
<td></td>
</tr>
</tbody>
</table>

---

**Note**

The maximum supported difference between **Exclusive_Begin_Timestamp** and **Inclusive_End_Timestamp** is three days. In case the difference is more, the returned result is limited to the last three days before **Inclusive_End_Timestamp**.
Users Affected by High Risk and High Confidence Incidents

This query generates a priority list table of high risk and high confidence users regardless of whether they have a confirmed campaign. Search for high risk, high confidence, and deduplicate usernames. Since all these incidents are both high risk and high confidence, consider reimaging the affected device.

```
confidence="High" risk="High" | dedup cUsername | table cUsername campaign
```

Users Affected by Campaign

This query generates a chart of the number of infected users over time and broken down by campaign. Search for nonempty campaign, bin by a time span of one day, and compute a distinct count of usernames within that bin.

```
campaign!="" | timechart dc(cUsername) span=1d by campaign
```

**Note**

In SPLUNK, the time chart shortcut can be used.

Command and Control Servers

This query generates a list of all detected command-and-control (C&C) servers in the Confirmed category. Search for nonempty campaign, while showing server IP address and campaign, and then deduplicate server IP addresses. The result lists C&C IP destination addresses being used by the infected devices to maintain C&C communication. For each C&C IP address, you also see which Threat campaign it is involved with. Can be used to query other systems for more intelligence, provide indicators of compromise (IOCs), and identify malicious processes and applications on the infected endpoint.

```
campaign!="" | table sip campaign | dedup sip
```

CTA Integration with Cisco ISE

Cisco Identity Services Engine (ISE) is a security policy management platform that provides secure access to network resources. Cisco ISE functions as a policy decision point and enables enterprises to ensure compliance, enhance infrastructure security, and streamline service operations. Cisco ISE allows enterprises to gather real-time contextual information from networks, users, and devices. You can then use that information to make proactive governance decisions by tying identity to various elements in the network.

CTA integrates with Cisco ISE to deliver a network-level quarantine, which features the ability to cut an infected device from the network so that no sensitive data can be exfiltrated further. The integration between CTA and Cisco ISE uses STIX/TAXII. For critical-level risk findings in which the system is able to attribute the infection to an individual user, Cisco ISE receives a Requested Course of Action that suggests a Threat Centric Network Access Control (TC-NAC) Quarantine, which is part of the Cisco Rapid Threat Containment framework. Depending on the risk associated with an infection, the Requested Course of Action could be Monitoring, Eradication, Internal Blocking, or a combination. Internal Blocking is the course of action intended to be used in the blocking policies in TC-NAC. For more information, see [Cisco Rapid Threat Containment](#).
You can develop your own solution by using Cisco ISE and the data feed provided by the CTA STIX/TAXII service. The data feed includes information on identifying the infected device and the action to be performed. You can define quarantine policies in Cisco ISE based on the recommendations in the CTA STIX/TAXII feed. For information on how to configure the CTA adapter in Cisco ISE, see the Cisco ISE Administrator Guide, Release 2.2.

Note
CTA works with user identities listed in the web proxy logs as client IP's or user names. Specifically, in the case of an IP addresses, the IP address that is available through the proxy logs may be an IP address that collides with another IP address (for another device) on the internal corporate network. For example, roaming users connected via AnyConnect with a split-tunnel directly to the Internet may acquire a local IP address they have at home (for instance, a 10.0.0.x address), which may collide with an IP address in an overlapping private range used in the internal corporate network. When you define the Rapid Threat Containment policies, consider your logical network architecture to avoid quarantine actions being applied to mismatched devices.