Splunk SOAR Integration Guide for Vectra XDR

Version: October 2023

Table of Contents

Introduction

Vectra XDR for Splunk SOAR empowers the SOC to create and manage incidents using Vectra AI's Attack Signal Intelligence for the Respond User Experience (Respond UX).

This integration allows the security operations center to create and manage incidents based on prioritized entities, powered by Vectra AI's Attack Signal Intelligence. Integrating Vectra and Splunk enables security teams to synchronize Vectra XDR Entities with Splunk SOAR events in real time, making it feasible to manage operations from a single place.

Integration value is achieved by injecting Vectra's integrated signal into the security operations center in a structured and highly efficient approach to ultimately transform the analyst experience to enable:

- Rich prioritization
- Incident management
- Detailed investigations
- Enrichment
- Enforcement
- Resolution
- Reporting

Document and Release Information

Vectra XDR for Splunk SOAR v1.0.1 is the initial release that supports Vectra's Respond UX. This is the initial version of this document.

Terminology

There are several terms that can be used interchangeably. The following table provides the Splunk SOAR term as well as other terms that may be used to refer to the same.

Splunk Term	Additional Terms
App	Vectra XDR for Splunk SOAR
Asset	Configuration Profile
Action	Function, Command
Event	Incident, Alert, Container

Architecture

Integrating Vectra with Splunk SOAR utilizes Vectra's REST API (v3.3 or higher) in a PULL model. The Vectra app resides inside the Splunk SOAR platform and uses REST API calls to pull the appropriate data following the operator configured polling interval (figure 1).

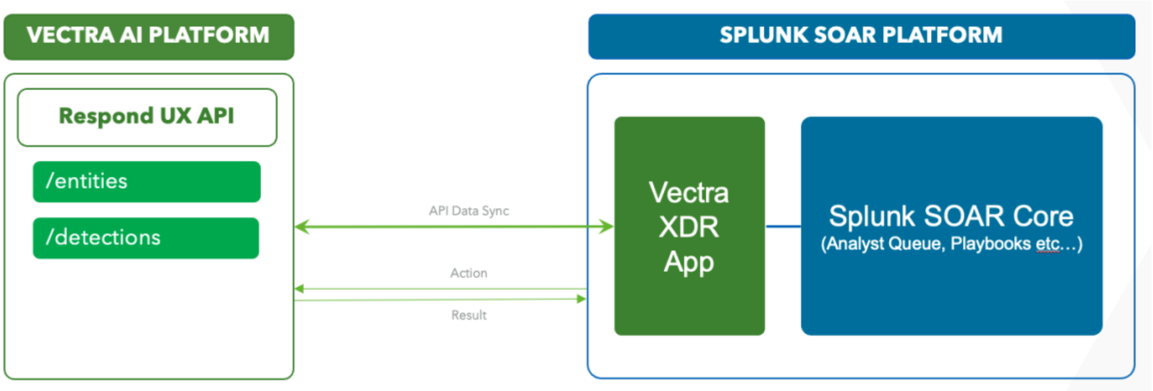


Figure 1 - Simplified Architecture

One or more Vectra Respond UX tenants provide the source of the data. Configuration Profiles (assets) are used to configure the details of how to communicate with a specific tenant (i.e., tenant URL and API credentials), polling schedule, as well as API error handling. The integration is designed to retrieve entity and detection data (along with all associated components such as notes, tags, and assignments) and ingestion filters enable the operator to fine tune the data that is considered for ingestion.

Multiple assets are required when there are multiple Vectra tenants but can also be used when there is a single tenant that requires competing ingestion filters. Figure 2 demonstrates a scenario where the operator wishes to ingest all prioritized entities from Vectra Tenant 1 as well as entities of any priority that has at least one exfiltration detection.

Once data begins ingestion, an app-embedded de-duplication mechanism controls if something is new and unique or if an update is warranted. If a previous event does not exist a new event will be created. If there is an existing event, then the appropriate updates are made to the existing event to ensure no duplicates.

The final components of the app include the supported functions (actions) as well as automations and playbooks. These will be covered later in this document.

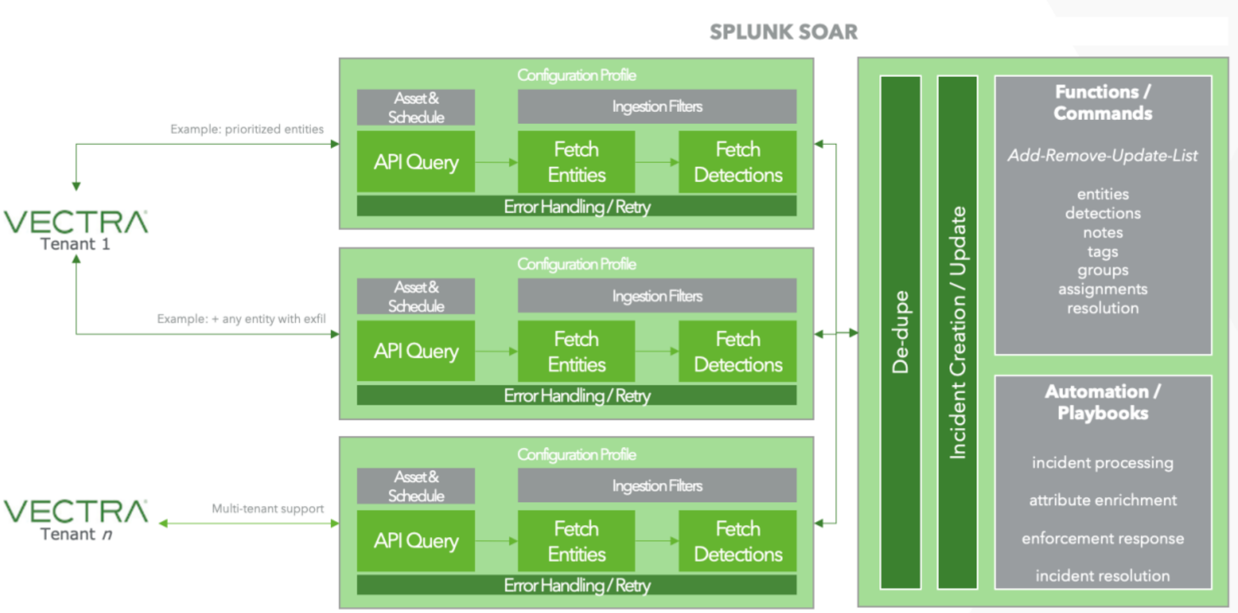


Figure 2 - Block Diagram

The next order of detail includes the operational components. As data makes it through ingestion as per the configuration profile, events are created. Each event will have one or more set of artifacts. There are three types of artifacts which include entity, detection, and assignment artifacts. Each artifact type holds several attributes which present the Vectra data. For example, *entity_name* is an attribute that resides in the entity artifact. The integration includes support for several actions where each action is a command that can be issued to the Vectra AI platform (ex. add tag). Playbooks are used to define automation flows and can consist of multiple commands as well instructions for interfacing with other apps configured in the Splunk SOAR environment.

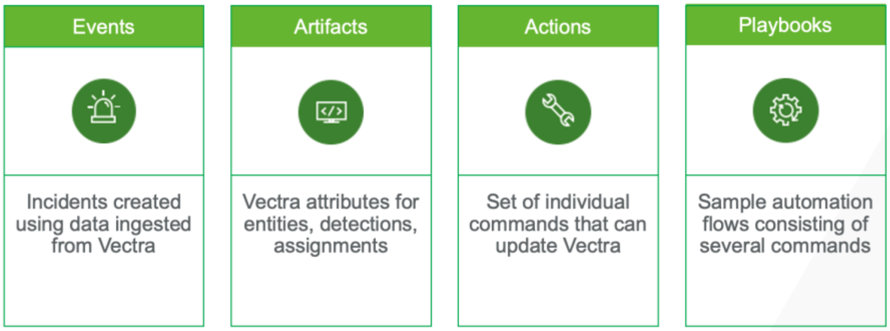

Events
Artifacts
Actions
Playbooks
Incidents created using data ingested from Vectra
Vectra attributes for entities, detections, assignments
Set of individual commands that can update Vectra
Sample automation flows consisting of several commands

Figure 3 - Operational Components

Events (incidents) are a foundational component of the integration as that is the starting point for any investigative or response workflow. Vectra employs Attack Signal Intelligence to conduct ruthless prioritization to ensure operational efficiency. Best practice is to implement a configuration profile that generates incidents based on Vectra prioritized entities. These incidents are funneled into an analyst queue in Splunk SOAR. Incidents are generated at the Vectra entity level only and detections are associated with an entity. An incident includes one or more artifact types which house all the Vectra attributes that make up the entity and its detections. There are several other SOAR components that are part of the incident layout. These components include an Owner (SOAR owner), incident status, as well as several other components that may be blank (ex. files, tags, etc.). It's possible to attach a workbook to an incident, run individual actions or launch playbooks.

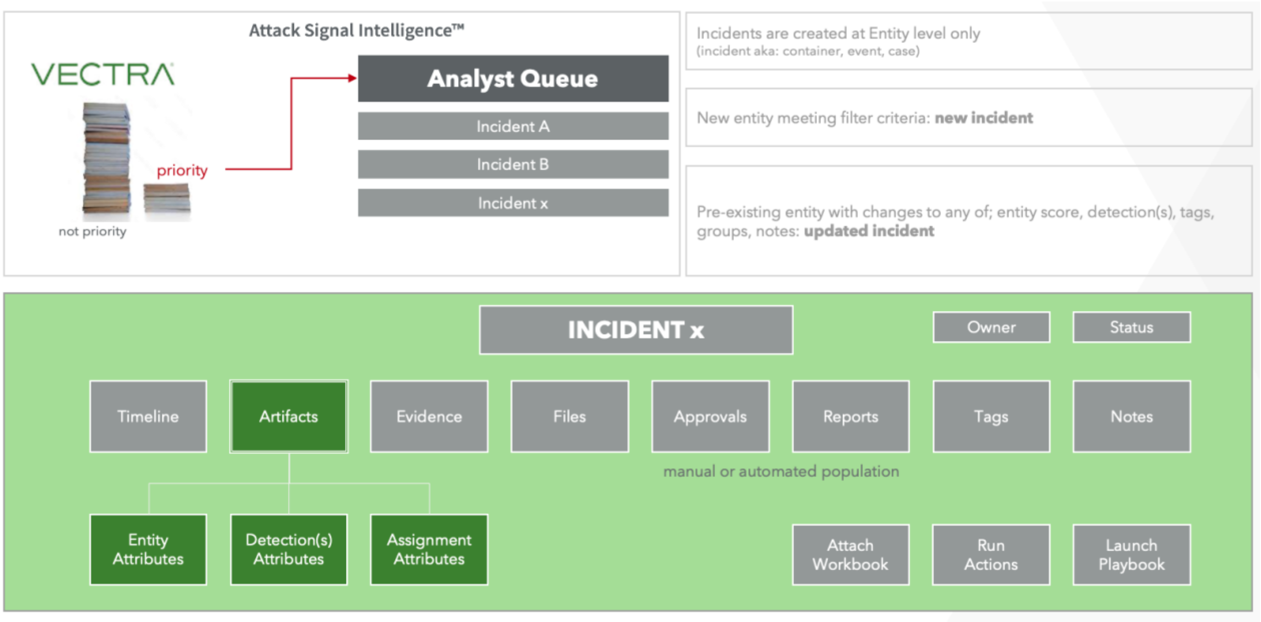

Attack Signal Intelligence™
VECTRA
priority
not priority
Analyst Queue
Incident A
Incident B
Incident x
Incidents are created at Entity level only
(incident aka: container, event, case)
New entity meeting filter criteria: new incident
Pre-existing entity with changes to any of; entity score, detection(s), tags, groups, notes: updated incident
INCIDENT x
Owner
Status
Timeline
Artifacts
Evidence
Files
Approvals
Reports
Tags
Notes
manual or automated population
Entity Attributes
Detection(s) Attributes
Assignment Attributes
Attach Workbook
Run Actions
Launch Playbook

Figure 4 - Incident Structure

Implementation

Vectra Pre-requisites

The minimum requirements on the Vectra side to configure this integration include:

- Vectra Respond UX tenant running API version 3.3 or higher.

  If you aren't sure which analyst experience is being utilized, please refer to this support article: https://support.vectra.ai/s/article/KB-VS-1673

- API client ID and secret configured with the Security Analyst Role.

  To create the API client:

  1. Log in to your Vectra Respond UX, navigate to *Manage* > *API Clients*, and click "Add API Client".
  2. Enter a name for the client you are creating, select the role, optionally enter a description and then click "Generate Credentials".
  3. Copy the client ID and the secret and store in a safe location and then click "Done".

Splunk Pre-requisites

The minimum requirements on the Splunk side to configure this integration include:

- Splunk account that has access to download content from Splunkbase.
- Splunk SOAR software installed (on-prem or cloud) v6.0.2 or higher.
- Splunk SOAR local account with access to install apps.
- Splunk SOAR platform must be able to communicate with the Vectra tenant over port 443.
- Modified permissions for the 'automation' user so that it includes 'delete' permissions for events (this is a deviation from the default permissions).
- Optional – if you don't want to combine the Vectra event data with any other systems then a new label should be created for use in asset configuration. New labels can be created from the Splunk SOAR UI under Administration > Event Settings > Label Settings. PRO TIP: If you intend to conduct some testing prior to production then use a 'throw away' label name and not the name you wish to use in full production. A new label will be required if you wish to 're-ingest' previously ingested data.

Aside from the automation user requiring delete permissions (this prevents duplicates) and best practices, the Vectra integration does not impose any other specific modifications or requirements of the Splunk SOAR platform. Please refer to the Splunk SOAR documentation for recommendations on system requirements and instructions for managing users, permissions, and labels.

Downloading and Installing the App

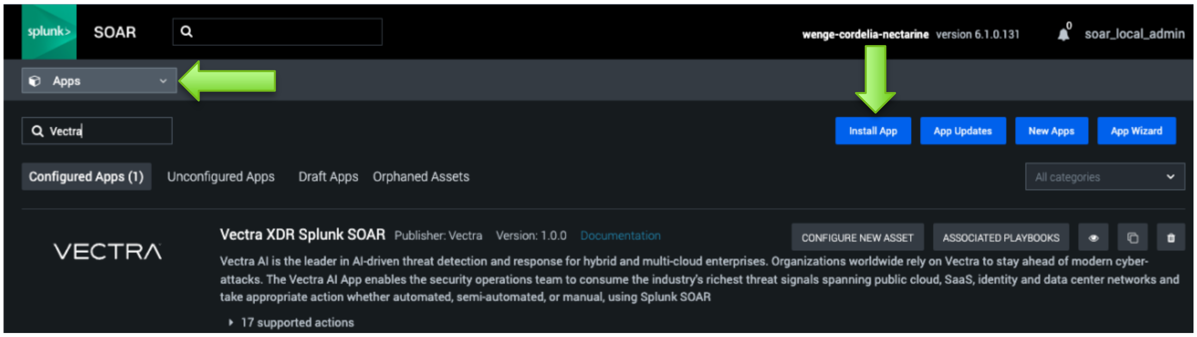
The integration (app) is available for download from Splunkbase at this location:

https://splunkbase.splunk.com/app/7072

Publisher: Vectra
App Version: 1.0.1
Product Name: Vectra XDR for Splunk SOAR
Supported Versions: Vectra Respond UX with Vectra API v3.3+

To install the app (figure 5):

1. Log in to your Splunk SOAR UI, navigate to *Apps*, and click "Install App".
2. Drag the download Vectra app tarball as instructed or browse to select it.
3. Click “Install” and follow the on-screen instructions.



Vectra AI is the leader in AI-driven threat detection and response for hybrid and multi-cloud enterprises. Organizations worldwide rely on Vectra to stay ahead of modern cyber-attacks. The Vectra AI App enables the security operations team to consume the industry's richest threat signals spanning public cloud, SaaS, identity and data center networks and take appropriate action whether automated, semi-automated, or manual, using Splunk SOAR

Figure 5 – Install App

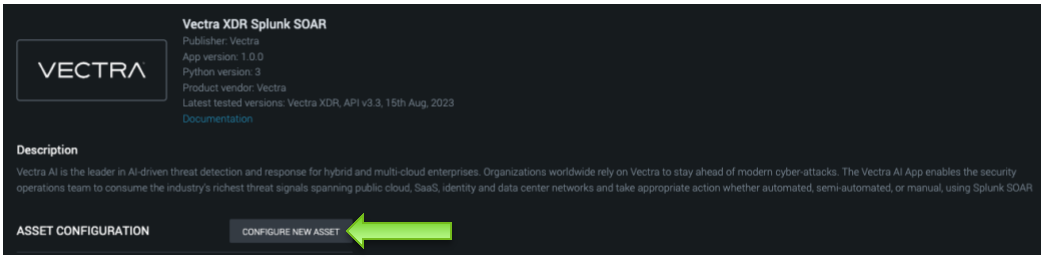
Implementation Checklist

Prior to configuring a new asset, it may be helpful to review the following checklist as there are several parameters that may be required for configuration. Mandatory parameters are prefixed with a *.

Parameter	Content
* Asset name	Ex. Vectra_*tenantID*
Asset description	Ex. Vectra tenant abc
* Vectra tenant base URL	Ex. https://999999999999.xx1.portal.vectra.ai
* Vectra API client ID	Generated in Vectra – Mange > API Clients
* Vectra API client secret	Generated in Vectra – Mange > API Clients
* Prioritized entity polling	True, false, all (best practice is true)
* Entity types to poll	Host, account, all (best practice is all)
Tag filtering	CSV list – only entities with matching tags will be ingested
Detection category filtering	Single selection (Botnet, C2, Recon, Lateral, Exfil, Info, All)
Detection type filtering	Single item free text – only entities with the detection specified will be ingested
* Low severity level	Vectra entities with urgency score up to this value is mapped to Low
* Medium severity level	Best practice – enter the same value as Vectra priority threshold (-1) This will map Vectra prioritized entities to Splunk high severity level
* Splunk label	To apply Vectra events to
* Polling type	Off (manual), Scheduled, Interval
Polling interval	Specific time of day or interval in minutes
Splunk approvers	Primary and secondary playbook approvers

Initial Configuration of New Asset

An asset configuration is required to communicate with the Vectra platform to pull data. With the Vectra XDR for Splunk SOAR app installed, navigate to the Vectra app from the Apps menu in the Splunk SOAR UI and select “Configure New Asset” (figure 6).



Vectra XDR Splunk SOAR
Publisher: Vectra
App version: 1.0.0
Python version: 3
Product vendor: Vectra
Latest tested versions: Vectra XDR, API v3.3, 15th Aug, 2023
Documentation

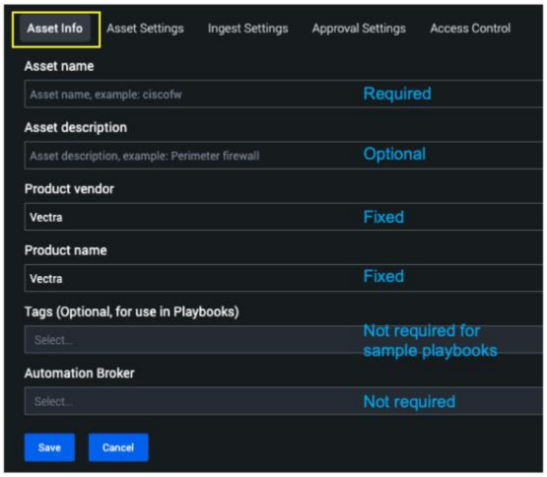
Description

Vectra AI is the leader in AI-driven threat detection and response for hybrid and multi-cloud enterprises. Organizations worldwide rely on Vectra to stay ahead of modern cyber-attacks. The Vectra AI App enables the security operations team to consume the industry's richest threat signals spanning public cloud, SaaS, identity and data center networks and take appropriate action whether automated, semi-automated, or manual, using Splunk SOAR

Figure 6 - Configure New Asset

Note: It’s possible to configure multiple assets for the same or different tenants. Multiple assets are required when you have multiple Vectra tenants or have conflicting filters. Follow these procedures for configuring each asset as needed.

The first tab for the asset configuration is the Asset Info. Fill out the asset info as per the checklist and make sure to hit “Save” before proceeding (figure 7).



Asset name
Asset name, example: ciscofw — Required

Asset description
Asset description, example: Perimeter firewall — Optional

Product vendor
Vectra — Fixed

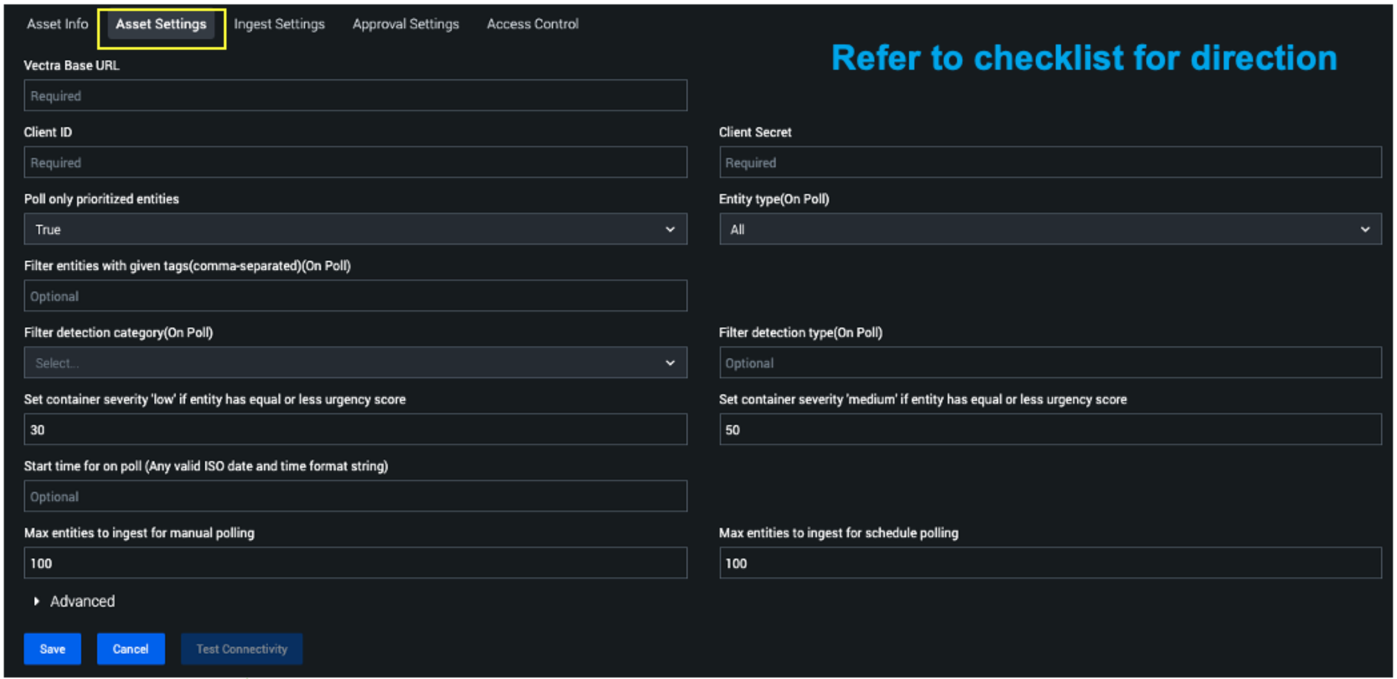
Product name
Vectra — Fixed

Tags (Optional, for use in Playbooks)
Select... — Not required for sample playbooks

Automation Broker
Select... — Not required

Figure 7 - Asset Info

The second tab includes the Asset Settings and these detail the communication with the desired Vectra platform (figure 8). It’s best practice to Test Connectivity after saving to confirm Splunk SOAR can communicate with the Vectra tenant.



Vectra Base URL: Required

Client ID: Required

Client Secret: Required

Poll only prioritized entities: True

Entity type(On Poll): All

Filter entities with given tags(comma-separated)(On Poll): Optional

Filter detection category(On Poll): Select...

Filter detection type(On Poll): Optional

Set container severity 'low' if entity has equal or less urgency score: 30

Set container severity 'medium' if entity has equal or less urgency score: 50

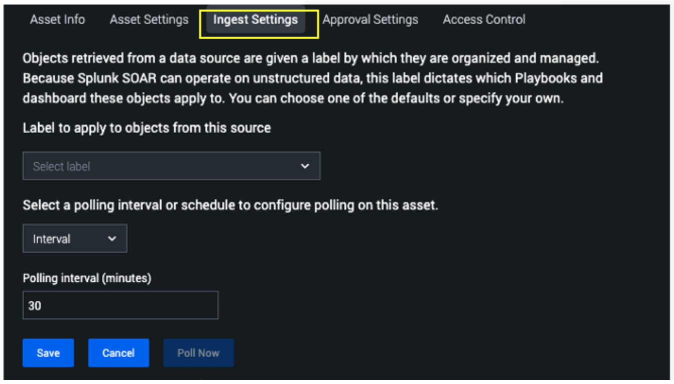
Start time for on poll (Any valid ISO date and time format string): Optional

Max entities to ingest for manual polling: 100

Max entities to ingest for schedule polling: 100

Figure 8 - Asset Settings

The Ingest Settings tab includes the controls for the polling schedule and mapping of Vectra data to Splunk SOAR labels (figure 9). Unless the environment is significantly busy (i.e., several dozen detections per minute), a polling interval of one minute is typical. The Poll Now button can be used to force a poll. When using Poll Now, take note the default setting is to retrieve only one container and 10 artifacts so modify as needed.



Objects retrieved from a data source are given a label by which they are organized and managed. Because Splunk SOAR can operate on unstructured data, this label dictates which Playbooks and dashboard these objects apply to. You can choose one of the defaults or specify your own.

Label to apply to objects from this source: Select label

Select a polling interval or schedule to configure polling on this asset.: Interval

Polling interval (minutes): 30

Figure 9 - Ingest Settings

The last two sections for Approval and Access Control are optional. Please refer to the Splunk SOAR documentation for additional information surrounding these configurations.

https://docs.splunk.com/Documentation/SOARonprem/latest/Admin/AppsAssets

Operational Components

At this stage, the Vectra app is installed, configured and the system should be receiving data. The architecture and implementation is complete so next we will take a deeper look into the operational perspective in more detail.

Events

Also referred to as containers, incidents, and alerts, an event is the starting point for incident management, workflow, and automation. Vectra events are accessible via the Sources menu in the Splunk UI by selecting a filter or the label that was provided during the app configuration (figure 10).

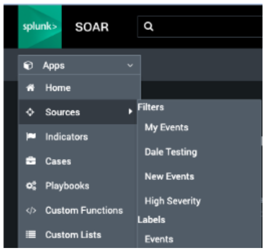
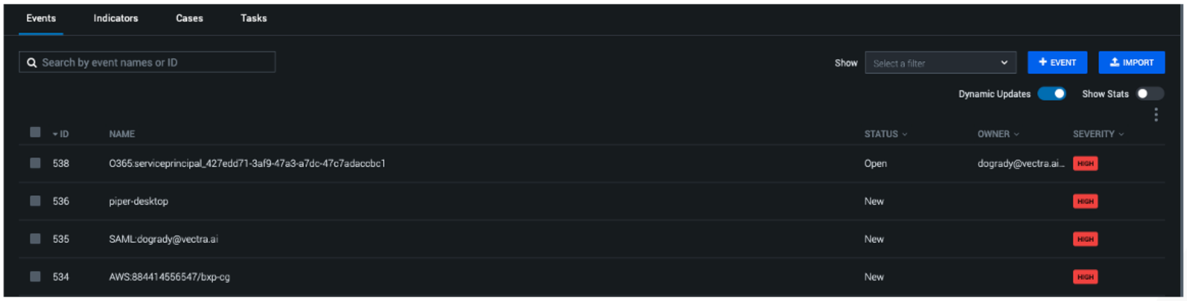


Figure 10 - Event Sources

Events (figure 11) can be promoted to cases in Splunk, or they can be managed individually as events. Since Vectra is already attributing several pieces of evidence (i.e., detections) to a single prioritized entity, the operator may find it unnecessary to promote to cases but that is operator preference.

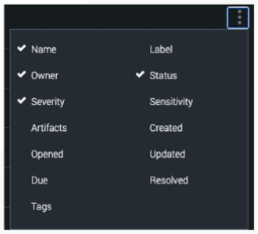
The ingestion mechanism includes de-duplication logic natively to ensure that if a container doesn't already exist for a new event that one is created. Conversely, if a container already exists then the event is updated rather than creating a duplicate. Security analysts should strive to resolve events as they appear as they are already prioritized via Vectra Attack Signal Intelligence.



ID	NAME	STATUS	OWNER	SEVERITY
538	O365:serviceprincipal_427edd71-3af9-47a3-a7dc-47c7adaccbc1	Open	dogrady@vectra.ai...	HIGH
536	piper-desktop	New		HIGH
535	SAML:dogrady@vectra.ai	New		HIGH
534	AWS:884414556547/bxp-cg	New		HIGH

Figure 11 – Events

Within the event the operator can define which columns will appear in the layout as per personal preference (figure 12)

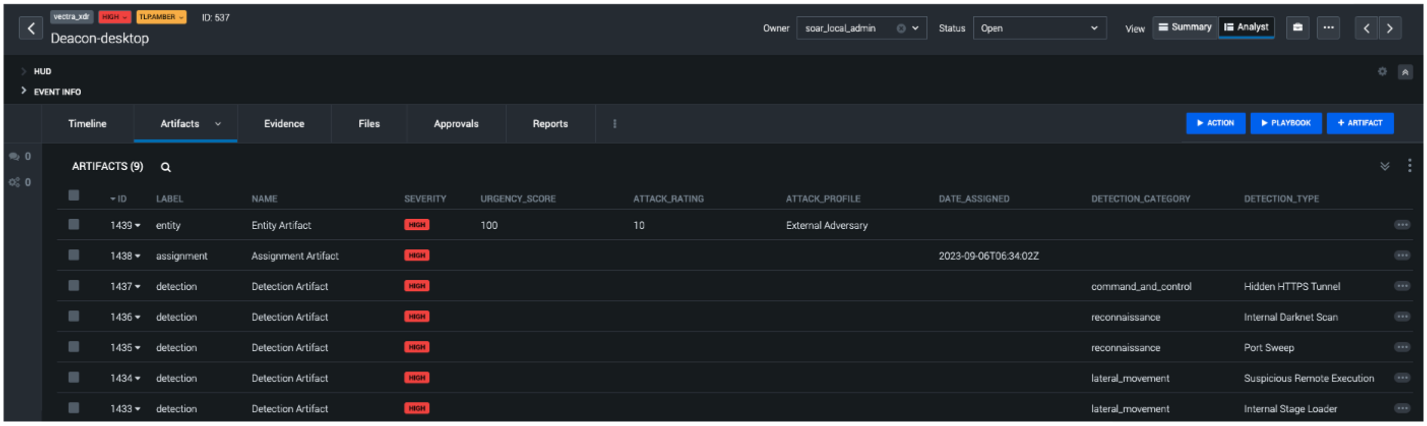


✓ Name
✓ Owner
✓ Severity
Artifacts
Opened
Due
Tags
Label
✓ Status
Sensitivity
Created
Updated
Resolved

Figure 12 - Events Column Layout

Artifacts

Each container (event) contains one or more artifacts which are used to hold the Vectra attributes. Artifacts exist for entity, detection, and assignment. There should only be a single entity artifact while there may be several detection artifacts or even assignment artifacts in the event there are re-assignments. The available artifacts are viewed by selecting the “Artifacts” tab from within an event (figure 13). If the artifact tab is not available, change your view in the top right from summary to analyst.



ID: 537

Deacon-desktop

Owner: soar_local_admin

Status: Open

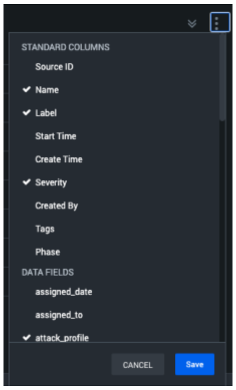
View: Analyst

ARTIFACTS (9)

ID	LABEL	NAME	SEVERITY	URGENCY_SCORE	ATTACK_RATING	ATTACK_PROFILE	DATE_ASSIGNED	DETECTION_CATEGORY	DETECTION_TYPE
1439	entity	Entity Artifact	HIGH	100	10	External Adversary			
1438	assignment	Assignment Artifact	HIGH				2023-09-06T06:34:02Z		
1437	detection	Detection Artifact	HIGH					command_and_control	Hidden HTTPS Tunnel
1436	detection	Detection Artifact	HIGH					reconnaissance	Internal Darknet Scan
1435	detection	Detection Artifact	HIGH					reconnaissance	Port Sweep
1434	detection	Detection Artifact	HIGH					lateral_movement	Suspicious Remote Execution
1433	detection	Detection Artifact	HIGH					lateral_movement	Internal Stage Loader

Figure 13 – Artifacts

The operator can choose to display certain artifact attributes in the layout by selecting the three dots and then checking the columns to include/exclude (figure 14). The order that the columns are ‘selected’ determine their placement left-to-right in the layout.



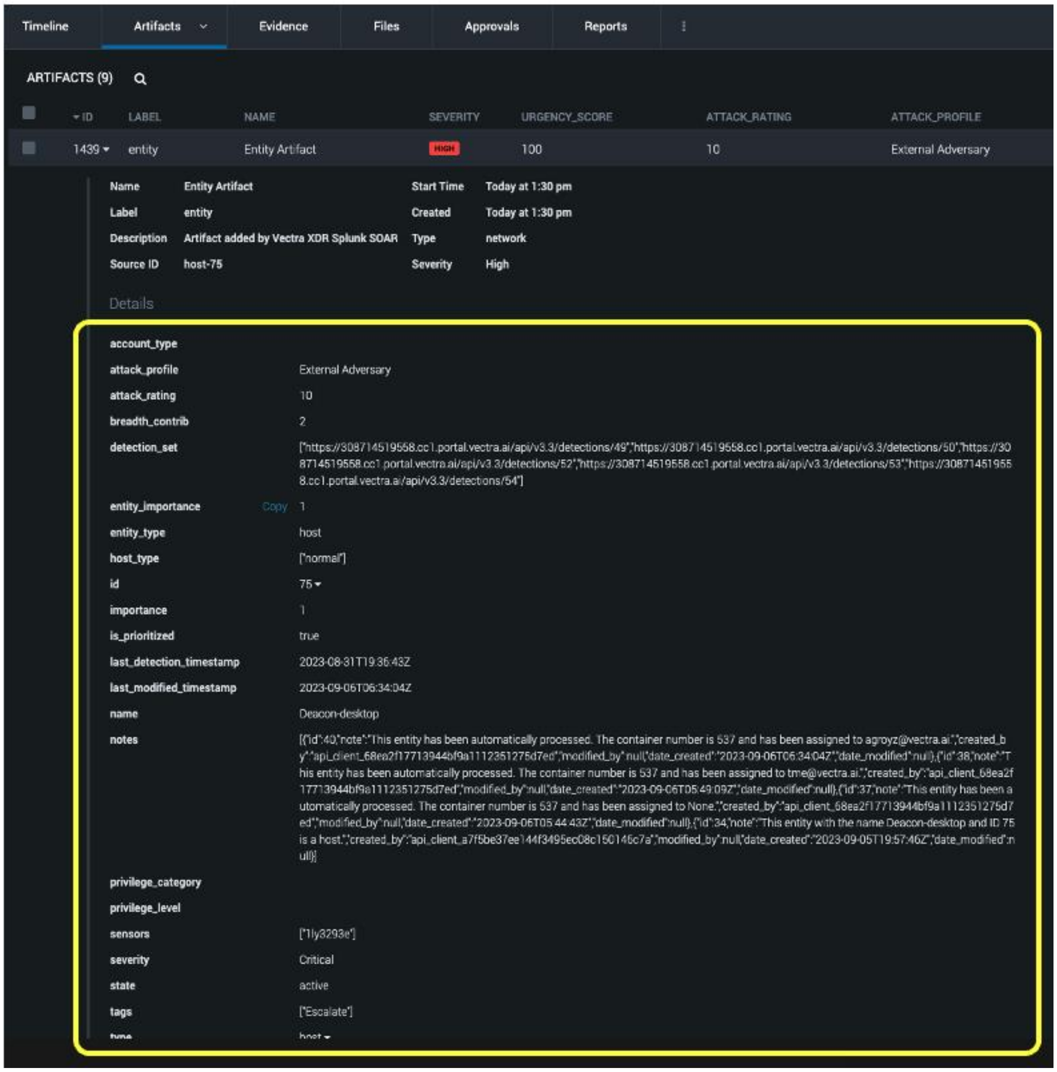
STANDARD COLUMNS
- Source ID
- ✓ Name
- ✓ Label
- Start Time
- Create Time
- ✓ Severity
- Created By
- Tags
- Phase

DATA FIELDS
- assigned_date
- assigned_to
- ✓ attack_profile

Figure 14 - Artifact Attributes Columns

Attributes

The Vectra data from entities, detections, and assignments are stored in artifact attributes (figure 15). Each artifact type and specific artifact (ex. a specific detection) will have different attributes, but all data retrieved from the API endpoints are stored in the artifact attributes. As evidence in figure 15, there is a wealth of detail (attributes) included including, in this example, attributes such as notes, tags, attack_profile etc. These attributes are available for use in playbooks.



ARTIFACTS (9)

	ID	LABEL	NAME	SEVERITY	URGENCY_SCORE	ATTACK_RATING	ATTACK_PROFILE
□	1439	entity	Entity Artifact	HIGH	100	10	External Adversary

Name	Entity Artifact	Start Time	Today at 1:30 pm
Label	entity	Created	Today at 1:30 pm
Description	Artifact added by Vectra XDR Splunk SOAR	Type	network
Source ID	host-75	Severity	High

Details

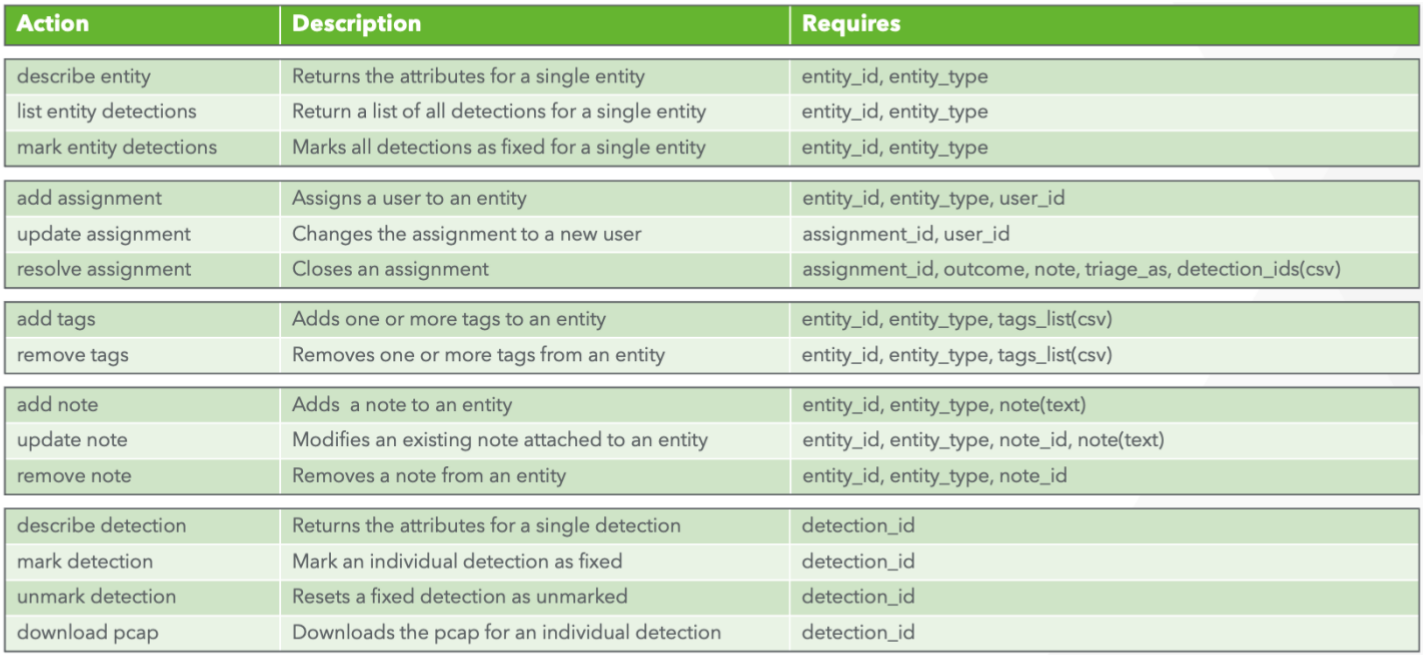
account_type	
attack_profile	External Adversary
attack_rating	10
breadth_contrib	2
detection_set	["https://308714519558.cc1.portal.vectra.ai/api/v3.3/detections/49","https://308714519558.cc1.portal.vectra.ai/api/v3.3/detections/50","https://308714519558.cc1.portal.vectra.ai/api/v3.3/detections/52","https://308714519558.cc1.portal.vectra.ai/api/v3.3/detections/53","https://308714519558.cc1.portal.vectra.ai/api/v3.3/detections/54"]
entity_importance	1
entity_type	host
host_type	["normal"]
id	75
importance	1
is_prioritized	true
last_detection_timestamp	2023-08-31T19:36:43Z
last_modified_timestamp	2023-09-06T06:34:04Z
name	Deacon-desktop
notes	[{"id":40,"note":"This entity has been automatically processed. The container number is 537 and has been assigned to agroyz@vectra.ai.","created_by":"api_client_68ea2f17713944bf9a1112351275d7ed","modified_by":null,"date_created":"2023-09-06T06:34:04Z","date_modified":null},{"id":38,"note":"This entity has been automatically processed. The container number is 537 and has been assigned to tme@vectra.ai.","created_by":"api_client_68ea2f17713944bf9a1112351275d7ed","modified_by":null,"date_created":"2023-09-06T05:49:09Z","date_modified":null},{"id":37,"note":"This entity has been automatically processed. The container number is 537 and has been assigned to None.","created_by":"api_client_68ea2f17713944bf9a1112351275d7ed","modified_by":null,"date_created":"2023-09-06T05:44:43Z","date_modified":null},{"id":34,"note":"This entity with the name Deacon-desktop and ID 75 is a host.","created_by":"api_client_a7f5be37ee144f3495ec08c150146c7a","modified_by":null,"date_created":"2023-09-05T19:57:46Z","date_modified":null}]
privilege_category	
privilege_level	
sensors	["1ly3293e"]
severity	Critical
state	active
tags	["Escalate"]
type	host

Figure 15 - Attributes

Actions

Actions are commands that can be run against the Vectra platform. The following table (Table 1) outlines all the supported commands, a description of what the command does, as well as what mandatory parameters are required for running the action.

Table 1 - Actions List

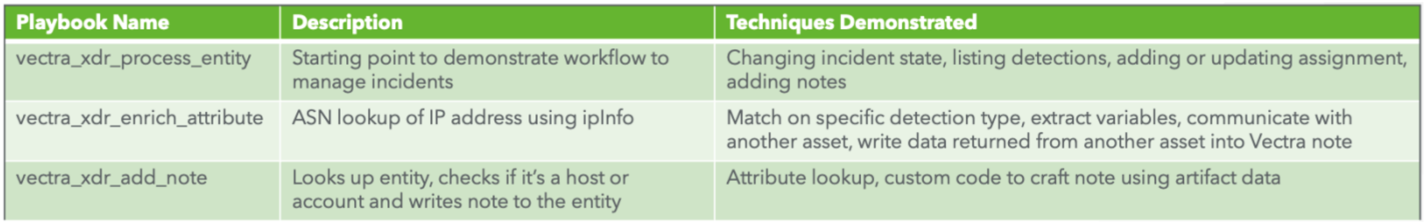


Action	Description	Requires
describe entity	Returns the attributes for a single entity	entity_id, entity_type
list entity detections	Return a list of all detections for a single entity	entity_id, entity_type
mark entity detections	Marks all detections as fixed for a single entity	entity_id, entity_type
add assignment	Assigns a user to an entity	entity_id, entity_type, user_id
update assignment	Changes the assignment to a new user	assignment_id, user_id
resolve assignment	Closes an assignment	assignment_id, outcome, note, triage_as, detection_ids(csv)
add tags	Adds one or more tags to an entity	entity_id, entity_type, tags_list(csv)
remove tags	Removes one or more tags from an entity	entity_id, entity_type, tags_list(csv)
add note	Adds a note to an entity	entity_id, entity_type, note(text)
update note	Modifies an existing note attached to an entity	entity_id, entity_type, note_id, note(text)
remove note	Removes a note from an entity	entity_id, entity_type, note_id
describe detection	Returns the attributes for a single detection	detection_id
mark detection	Mark an individual detection as fixed	detection_id
unmark detection	Resets a fixed detection as unmarked	detection_id
download pcap	Downloads the pcap for an individual detection	detection_id

Playbooks

Playbooks are used to define automation flows and can consist of multiple commands as well instructions for interfacing with other apps configured in the Splunk SOAR environment. Some sample playbooks are available for the Vectra integration and the intended purpose is to provide a few workflows that demonstrate key techniques. The operator can take specific techniques from the sample playbooks to create their own automations based on their individual use cases. The sample playbooks, their descriptions, and the techniques demonstrated are outlined in Table 2. A deeper dive into playbooks is covered later in this document.

Table 2 - Sample Playbooks



Playbook Name	Description	Techniques Demonstrated
vectra_xdr_process_entity	Starting point to demonstrate workflow to manage incidents	Changing incident state, listing detections, adding or updating assignment, adding notes
vectra_xdr_enrich_attribute	ASN lookup of IP address using ipInfo	Match on specific detection type, extract variables, communicate with another asset, write data returned from another asset into Vectra note
vectra_xdr_add_note	Looks up entity, checks if it's a host or account and writes note to the entity	Attribute lookup, custom code to craft note using artifact data

Operations

Incident Creation Philosophy

Best practice: Generate incidents on an entity-by-entity basis versus detection-by-detection.

Why: A key pillar of Vectra AI's value proposition to organizations is SOC efficiency. Vectra accomplishes this by attributing behavioral detections to entities (currently, hosts & accounts), by leveraging AI to compute an urgency score that considers multiple factors such as detections, velocity of progression, significance of the entity itself, and finally bringing all detection and non-detection context together in one prioritized place. For this reason, we promote the generation of incidents based on entities in external tools to mirror the Vectra AI Platform value proposition which results in decreased ticket volume, alert fatigue, and false positives. The following tables show a real-world difference between a detection-centric (Table 3) approach (which competitors employ) to an entity-centric (Table 4) approach with Vectra. The result is an average reduction of 80% in ticket load, all while being laser-focused on what is most urgent.

Table 3 - Detection Centric

Month	# of Detections	Avg # of Tickets / Day
June 2023	418	14
July 2023	472	15
Aug 2023	762	25

Table 4 - Entity Centric

Month	# of Entities	Avg # of Tickets / Day	% change of tickets created
June 2023	107	4	-74%
July 2023	107	3	-77%
Aug 2023	88	3	-88%

With Vectra's entity-centric prioritization, the detections are still available and relevant but instead of managing each detection as an isolated incident, they are managed holistically at the entity.

Orientation

The event layout contains a lot of information. The following diagram (Figure 16) is included to highlight a few areas that are of particular interest for Vectra events and does not include details on every item. For additional details around the event layout, please refer to your Splunk SOAR documentation.

Starting with the list of events it's advised to sort on the status column (sort from new to resolved) so that it's very clear which events have active investigations, and which need to be dispatched. If the best practice of ingesting prioritized entities is employed, then only high severity (Splunk) events will generate incidents as these are already mapped to Vectra prioritized entities.

Once an event is selected, it will present the event layout and from there the operator can specify the desired view (summary vs analyst), review artifacts and artifact details or call actions or playbooks from within the event. An owner should be assigned to the event and the status should be updated accordingly.

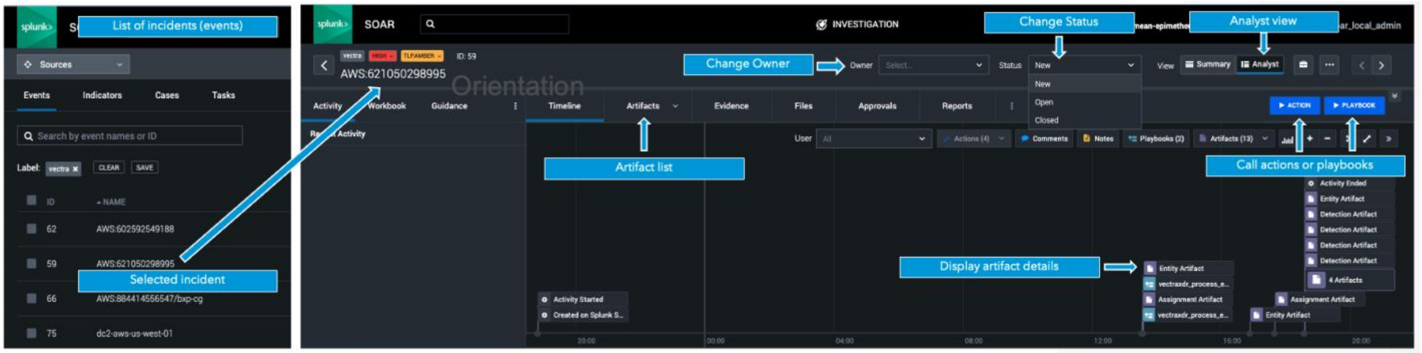


Figure 16 - Event Orientation

Workflow

The following diagram outlines a starter workflow that can be employed. The top line refers to the three Splunk SOAR status labels (new, open, closed) that are tied to an event. The second line provides more color on the stage of the incident life cycle.

The status of an un-opened event is set to new by default and this equates to the pending stage since no analyst is working the incident yet. Once the analyst (or incident commander) starts working the event they should manually change the status to “Open”, specify an “Owner” and assign the entity in Vectra by running the “vectra_xdr_process_entity” playbook to complete the assignment stage.

The event is in the open state and should progress through the investigation and/or remediation stages which are outside the scope of this document. The intent is to adjudicate the event, and this may involve manual or automated processes.

Once the event has been adjudicated it can move to the resolution stage. In the resolution stage the operator should run the resolve assignment action to close the incident on Vectra. Finally, the operator should set the Splunk SOAR event status to “Closed”.

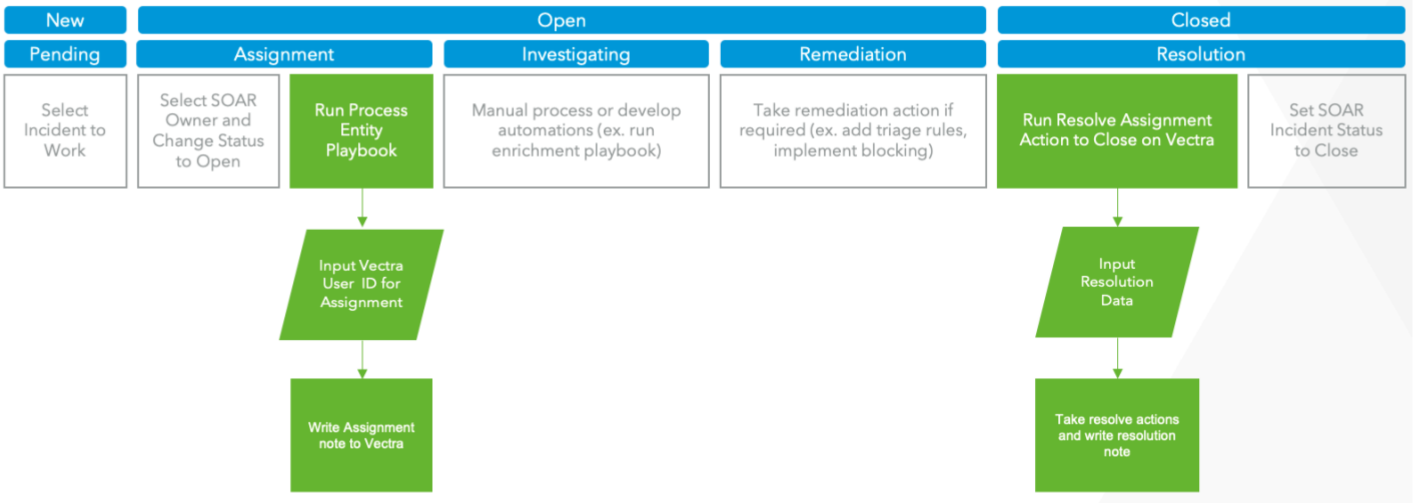
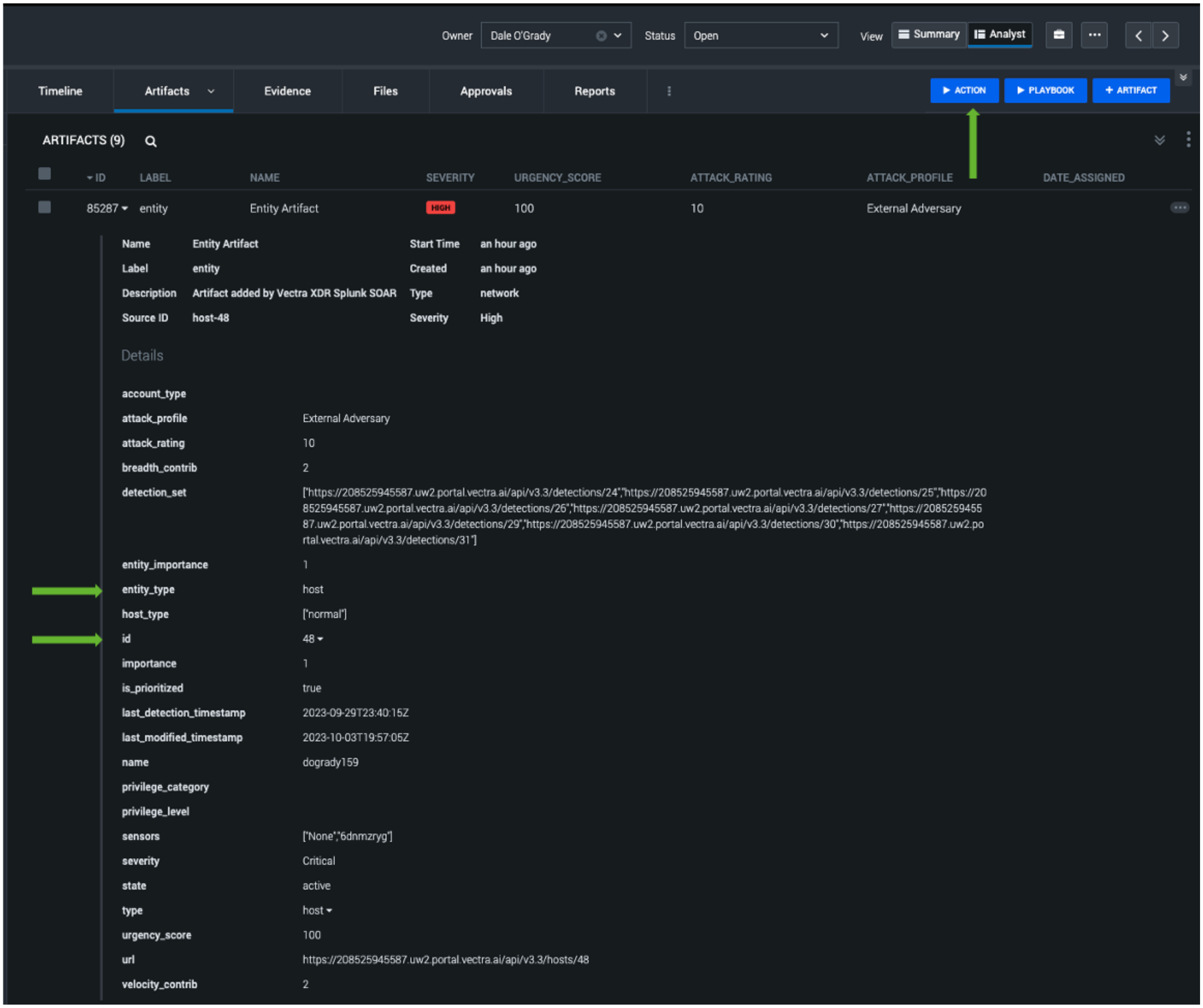

New
Open
Closed
Pending
Assignment
Investigating
Remediation
Resolution
Select Incident to Work
Select SOAR Owner and Change Status to Open
Run Process Entity Playbook
Manual process or develop automations (ex. run enrichment playbook)
Take remediation action if required (ex. add triage rules, implement blocking)
Run Resolve Assignment Action to Close on Vectra
Set SOAR Incident Status to Close
Input Vectra User ID for Assignment
Input Resolution Data
Write Assignment note to Vectra
Take resolve actions and write resolution note

Figure 17 - Recommended Workflow

Running Actions - General

It’s possible to run actions from within the event container. Since every action has varying mandatory inputs, it’s advised to review the corresponding artifact first and have the required input on hand. For the most part, the action names include the artifact type they apply to. For instance, actions such as ‘describe entity’ or ‘list entity detections’, apply to entity artifacts, whereas ‘mark detection’ or ‘describe detection’, applies to detection artifacts. Table 1 identifies the artifact type by the content in the requires column. In the following example (Figure 18), I wish to list the detections for an entity, and I know from Table 1 that this requires entity_id and entity_type so I have the entity artifact open and have found the necessary data so next I can select “Action”.



ID	LABEL	NAME	SEVERITY	URGENCY_SCORE	ATTACK_RATING	ATTACK_PROFILE	DATE_ASSIGNED
85287	entity	Entity Artifact	HIGH	100	10	External Adversary	

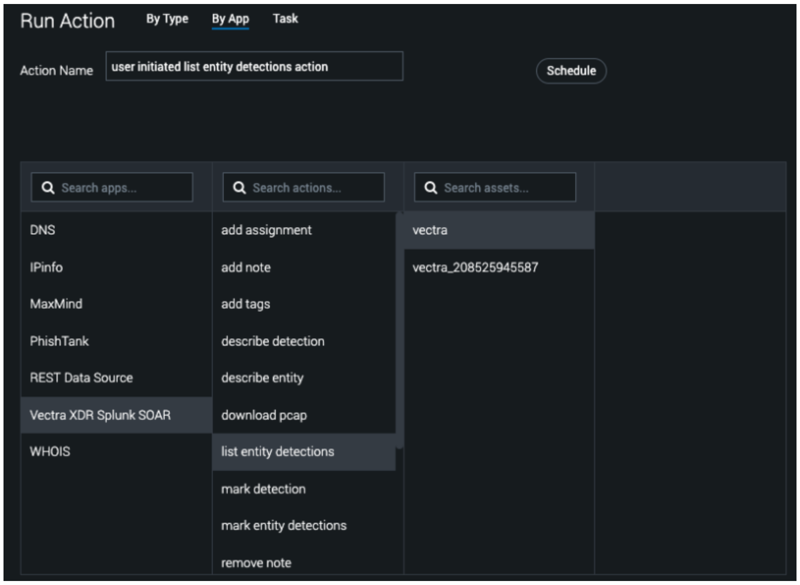
Name	Entity Artifact	Start Time	an hour ago
Label	entity	Created	an hour ago
Description	Artifact added by Vectra XDR Splunk SOAR	Type	network
Source ID	host-48	Severity	High

Details

account_type	
attack_profile	External Adversary
attack_rating	10
breadth_contrib	2
detection_set	["https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/24","https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/25","https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/26","https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/27","https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/29","https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/30","https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/31"]
entity_importance	1
entity_type	host
host_type	["normal"]
id	48
importance	1
is_prioritized	true
last_detection_timestamp	2023-09-29T23:40:15Z
last_modified_timestamp	2023-10-03T19:57:05Z
name	dogrady159
privilege_category	
privilege_level	
sensors	["None","6dnmzryg"]
severity	Critical
state	active
type	host
urgency_score	100
url	https://208525945587.uw2.portal.vectra.ai/api/v3.3/hosts/48
velocity_contrib	2

Figure 18 - Action Requirements

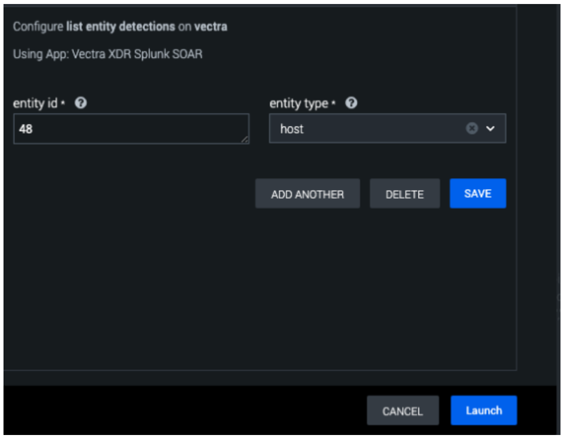
Select “Action” brings up a “Run Action” display where various options exist to find the appropriate action to run. In this scenario (Figure 19), I selected “By App”, then selected the Vectra XDR Splunk SOAR app to restrict the actions to only those supported by this app. An asset is selected if there are multiple assets and then the input parameters that appear next will vary by the action selected.


Run Action

Action Name: user initiated list entity detections action

Figure 19 - Run Action

Since the I select the action ‘list entity detections’, I’m required to enter the required input data (Figure 20) which I retrieved from the artifact attributes. Selecting “Launch” will run the action against the asset and input data provided.


Configure **list entity detections** on **vectra**

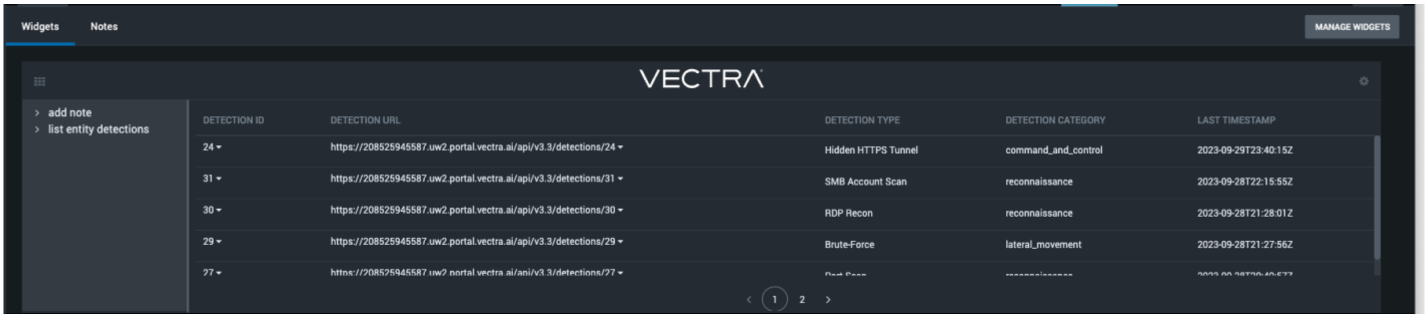
Using App: Vectra XDR Splunk SOAR

entity id •: 48

entity type •: host

Figure 20 - Action Input

The output of the action is displayed in the ‘Widgets’ section of the event (Figure 21).



DETECTION ID	DETECTION URL	DETECTION TYPE	DETECTION CATEGORY	LAST TIMESTAMP
24	https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/24	Hidden HTTPS Tunnel	command_and_control	2023-09-29T23:40:15Z
31	https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/31	SMB Account Scan	reconnaissance	2023-09-28T22:15:55Z
30	https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/30	RDP Recon	reconnaissance	2023-09-28T21:28:01Z
29	https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/29	Brute-Force	lateral_movement	2023-09-28T21:27:56Z
27	https://208525945587.uw2.portal.vectra.ai/api/v3.3/detections/27	Port Scan	reconnaissance	[illegible]

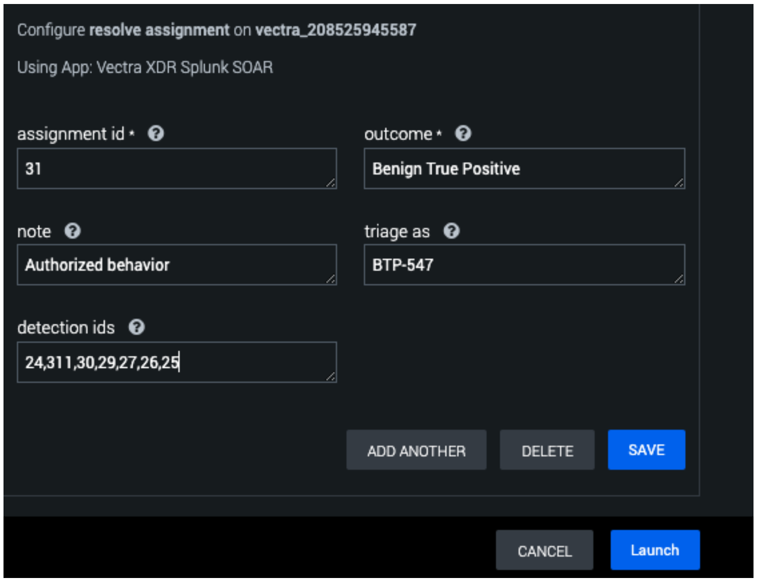
Figure 21 - Action Output

Running Actions – Resolve Assignment

Most actions are self-explanatory and are relatively finite so it's not necessary to cover each one in detail. The action that requires the most explanation is the 'Resolve Assignment' action. This action covers several components and is very powerful, but it does require some prep work. When run, this action will take an entity that has been assigned and will mark it resolved including adding a resolution outcome, adding a note for the Vectra operational metrics report, optionally triage the detections associated with the entity and place the desired label on the triaged detections. The following table (Table 5) highlights the input that is required to run the action (Figure 22) and where to find the relevant data.

Table 5 - Resolve Assignment Parameters

Parameter	Format	Source
Assignment ID	Single integer	Assignment artifact at label assigned_to id
Outcome	Free Form String	Benign True Positive, Malicious True Positive, False Positive (exact match)
Note	Free Form String	Free form – this note only appears in operational metrics report
Triage As	Free Form String	Short label that shows up beside the triaged detection in Vectra
Detection ID's	Multiple integers csv	Run describe entity detections action and obtain from widget data



Configure **resolve assignment** on **vectra_208525945587**

Using App: Vectra XDR Splunk SOAR

assignment id *: 31

outcome *: Benign True Positive

note: Authorized behavior

triage as: BTP-547

detection ids: 24,311,30,29,27,26,25

Figure 22 - Resolve Assignment Action

Working with Playbooks

Playbooks will evolve over time and new content will be created to support specific use cases. They playbooks created or curated by Vectra aren't necessarily intended to function out of the box. In most cases some form of customization is required to operate inside the destination environment. The playbooks provided offer the starting point and structure to ease any customization burden.

Playbooks

Vectra provided playbooks and corresponding documentation are available for download from the following GitHub repository:

https://github.com/vectranetworks/splunk_soar_vectra_xdr

This repository includes documentation pertaining to using playbooks with the Vectra platform as well as several playbooks. Please refer to the repository for additional details on the topic of playbooks.

Known Limitations

Source IP

The source IP for the host entity is not present in the initial release but that is only a temporary limitation. Shortly after the initial release, the IP attribute will be added to the return API call and once the data is being returned, there is nothing required on the Splunk side. The inclusion of the source IP attribute is expected for inclusion before the end of 2023.

Detection Notes and Tags

It is not possible to add notes or tags directly to detections with this integration. This is by design so that the entire entity is investigated. Where appropriate, notes and tags should be added to the entity level and/or inside Splunk SOAR event management. All detection artifacts are attributed to the entity and are accessible via the entity itself.

Assignment on ID

When assignment an event to a user in Vectra an ID (integer) is required rather than a name. The API call requires an integer for assignment so this is by design, but it may be confusing to complete assignments. It's recommended to use an API query tool such as Postman to retrieve the list of users along with their IDs. As the user ID doesn't change once created the recommended approach is to add the list of users along with their ID to a playbook prompt to make it easier for the operator to complete the assignment.

The following support article includes additional details on using the Vectra SaaS Postman Collection: https://support.vectra.ai/s/article/KB-VS-1572

Troubleshooting

No events

If no events are displaying after configuration this could be the result of several things.

An incorrect key could have been entered during configuration. Use the 'test connectivity' button under asset settings to validate.

The filters that have been configured restricts the data that is initially received and if the filters are too restrictive, it's possible there is no matching data. Modify the "Asset Settings" filters to poll all entities for prioritized and remove any other filters to isolate the issue.

Review the polling settings under "Ingest Settings" to ensure that a polling interval or schedule has been set correctly. Alternatively, use the "Poll Now" button to initiate the poll and make sure to set a reasonable number for the maximum number of containers to receive (the default is only one container).

Ingested events are placed into a source that matches the label that was configured for the asset. Make sure to review the correct source or else you won't see the events.

Too many duplicates

Duplicate artifacts will be created when automation doesn't have correct permissions. This will be readily apparent as you may see several dozen artifacts for entity and detections under the same event. From within the "Administration" menu, select *User Management* > *Roles & Permissions* and then the "Automation" role. Under "Basic Permissions" modify "Events" to include "delete". Alternatively, adding the administrator role to automation will provide the level of access albeit more permissions than are necessary.

Playbook not running properly

When troubleshooting playbooks use "Playbook Debugger" to isolate testing to specific artifacts. It may be helpful to review each code block in the "Python Playbook Editor" to review any 'custom code' blocks for issues. The most common issue in the custom code blocks is when the code is referring to an incorrect asset. There may be a code block that includes assets=["vectra"] and if your asset isn't named "vectra" in this example, that will cause issues.

Worldwide Support Contact Information

- Support portal: https://support.vectra.ai/ (preferred contact method)
- Email: support@vectra.ai
- Additional information: https://www.vectra.ai/support