Cortex XSOAR Content Pack Design Document - Hello World

Introduction

Hello and thank you for your willingness to contribute to the Cortex XSOAR ecosystem. This Design Document is meant to help you define the structure and requirements of your Cortex XSOAR Content Pack and make sure that it is aligned with our best practices.

If you don't know what a Cortex XSOAR Content Pack is or our requirements, please review our Cortex XSOAR Developer Hub at xsoar.pan.dev to get started. Make sure to check out the concepts section.

How to complete this document

Please complete as many sections as possible, since some of them may not be relevant to your use case or pack structure.
This Document is required prior to beginning the build phase of your contribution.

Help & Support

Any question about this document or partner integrations can be answered in:
DFIR Slack community on any of the public channels here
Or by mail to: soar.alliances@paloaltonetworks.com

Resources and Background

To start creating your own contribution it is important to understand some basic concepts and how XSOAR works.
XSOAR Developer Hub can be found here:https://xsoar.pan.dev/docs/welcome
The most important things to understand before starting designing and developing are:

- Getting started guide with all the links to the relevant information such as:
    - Concepts
    - FAQ
    - Product training
    - DFIR Slack channel
    - Recommended tools
    - More relevant information
- **Design tutorial** - here
- Enablement videos of the whole building a pack process - here

General Information

Document Author (Partner)	Hello World
Document Reviewer (Palo Alto Networks)	Edi Katsenelson

Partner Information

Product Information

Each Pack should integrate with one product. Provide details about your product

Product Name	Soar Hello World
Description	The Hello World SOAR API consists of a sample set of API endpoints that allow customers to retrieve and update simulated alerts, run scans against a system and retrieve the results. It also provides reputation commands from IPs and domains.
Supported Version(s)	1.0
Notes	This is a Hello World Mock API designed to learn Cortex XSOAR

Team and Contacts

Name	Email	Title	Role (BD/Tech)
DevRel	your_email@here	PANW Developer Relations	Tech

Content Pack General Details

Content Pack Details

Company Name	**Hello World**
Partner ID	**hello-world-000**
Content Pack Name	**Hello World**
Content Pack Support Email	**support@helloworld.com**
Content Pack Product URL	**www.helloworld.com**

Content Pack Metadata

General metadata information
https://xsoar.pan.dev/docs/packs/packs-format#pack_metadatajson
Information about description, use cases, keywords, tags and categories
https://xsoar.pan.dev/docs/documentation/pack-docs

XSOAR Minimum Version	6.5
Content Pack Version *(major.minor.revision)*	1.0.0
Pack Description	This Content Pack is used to demonstrate the capabilities of a Cortex XSOAR integration and the common design patterns, and is linked to the code. The full integration code can be found here: https://github.com/demisto/content/tree/master/Packs/HelloWord The source code of the Hello World SOAR API, in case you want to deploy your own instance, can be found here:

	https://github.com/fvigo/soarhelloworld
Pack Use Cases	Utilities, Example
Pack Keywords	hello, world, beginners
Pack Tags	helloworld, development
Pack Categories	Utilities

Use Cases

Please provide a general overview of the Use Cases you are going to implement in this Content Pack.

Use Case	Description
Reputation	This Content Pack contains an integration able to retrieve reputation and WHOIS details for IPS and Domains
Alert Management	This Content Pack provides facilities to retrieve and handle alerts of the Hello World type, including search and alert resolution.
Device Scan	This Content Pack enables scanning of Endpoints and retrieval of the scan report via Cortex XSOAR.

Pack Content

XSOAR components - please read the links to get more information about each component.
A content pack must have at least one of the entities

Entity Type	Name(s)
Integrations	HelloWorld
Classifiers and Mappers	HelloWorld, HelloWorld-mapper
Playbooks	Handle Hello World Alert, HelloWorld Scan
Incidents Types	Hello World Alert type (Hello World ID, Hello World Type, Hello World Status fields)
Indicators	N/A
Automations (Scripts)	HelloWorldScript
Widgets & Dashboards	N/A
Other	N/A

If you have Integration as part of your pack, see the next page.

Integrations

Usually each content pack contains one integration, in case your pack includes more than one, you must replicate this section for every integration.
Please read about the fetching incidents/indicator feed options in the link below.

Integration Name (PascalCase)	HelloWorld
Integration Description	Interacts with the SOAR HelloWorld API to retrieve incidents, reputation and perform scans.
Minimum XSOAR version	6.5
Fetches incidents? *(Yes/No)*	Yes
(only if fetches incidents) **What Incident types does it support?**	Hello World Incident Type
(only if fetches incidents) ***What filters does it support? (i.e. severity, priority, etc.)***	The available filters for this integration are: min_severity: minimum severity of an alert. Options are “Low”, “Medium”, “High”, “Critical”. The integration will fetch all the alerts of the severity level specified here and all the ones above (i.e. if set to High, it will fetch High and Critical ones). alert_status: the status of the alert to fetch. Status can be either “ACTIVE” or “CLOSED”. By default we will filter only on ACTIVE alerts. alert_type: the type of HelloWorld Alert to retrieve. There are several types of alerts to be supported. The API supports only one at a time, unless the information is not specified: in that case all alerts are retrieved. There is no predefined list of alert types, as they keep changing over time.
Indicator Feed? *(Yes/No)*	No
(only if indicator feed) **What Indicator types it supports?**	N/A
Supports mirroring? *(Yes/No)*	No
(only if implements mirroring)	N/A

What mirroring capabilities does it support (select all that apply)? ● **Inbound** ● **Outbound** ● **Mapping**	

Integration Parameters

Please fill in the parameters that your integration supports.

- ***Connectivity Parameters**: include all necessary parameters to allow an integration to connect to your Product API, such as **URL**, **API Key**, etc: these parameters depend on your product/API (i.e. some products use username/password instead of API keys). Note that **proxy** and **insecure** are required and should not be removed.*

- ***Fetch Parameters**: if your integration supports fetching incidents, you should provide all the parameters needed to successfully work, such as maximum incidents to fetch per time, first fetch and all the filters. Additional common filters are **severity** and **type**, but they vary depending on your Product and its API. Note that **isFetch** and **incidentType** are required for integrations that fetch incidents and should not be removed.*

You can also check out the Hello World example as a reference.

*Supported parameter types are: **Short Text, Long Text, Boolean, Encrypted, Single Select, Multi Select.***

Connectivity Parameters

Parameter name	Type	Required *(Yes/No)*	Default Value	Description
proxy	Boolean	Yes	False	Whether to use XSOAR's system proxy settings to connect to the API.
insecure	Boolean	Yes	False	Whether to allow connections without verifying SSL certificates validity.
url	Short Text	Yes	https://soar.monstersofhack.com	The FQDN/IP the integration should connect to.
apiKey	Encrypted	Yes		The API Key required to authenticate to the service.

Fetch Incidents Parameters

Parameter name	Type	Required *(Yes/No)*	Default Value	Description
isFetch	Boolean	Yes	No	Enable fetch incidents
incidentType	Single-Select	No	None	Incident type to map if no classifier is provided.
max_fetch	Short Text	Yes	20	Maximum number of incidents to fetch every time.

firstFetch	Short Text	No	2 weeks	Date or relative timestamp to start fetching incidents from.
severity	Multi-Select	No	High,Critical	Severity levels of the alerts to fetch from the third party API (that will generate incidents in Cortex XSOAR).
alertType	Multi-Select	No	All	Type(s) of alerts to fetch from the third party API.
threshold_ip	Short Text	No	65	Score threshold for IP reputation command.
threshold_domain	Short Text	No	65	Score threshold for domain reputation command.

Integration Commands

Use this section to provide information about the commands that are supported by your integration: the name, a brief description of what they do, the list of their arguments and whether they are reputation commands (that are commands that return a reputation about a specific indicator type, such as !ip, !domain, !url, !cve), use the following naming convention for commands: PRODUCTNAME-OBJECTNAME-ACTION

Command Name	**Description**	**Arguments** *(comma separated)*	**Reputation command?** *(Yes/No)*	**Output Path**
helloworld-say-hello	Hello command - prints hello to anyone.	name	No	
helloworld-search-alerts	Search HelloWorld Alerts.	severity, status, alert_type, max_results, start_time	No	HelloWorld.Alert
helloworld-g	Retrieve alert	alert_id	No	HelloWorld.Al

et-alert	extra data by ID.			ert
helloworld-update-alert-status	Update the status for an alert.	alert_id, status	No	HelloWorld.Alert
helloworld-start-scan	Start scan on an asset.	hostname	No	HelloWorld.Scan
helloworld-scan-status	Retrieve scan status for one or more scan IDs.	scan_id	No	HelloWorld.Scan
helloworld-scan-results	Retrieve scan status in Context or as a File (default) for a Scan.	scan_id, format	No	HelloWorld.Scan
ip	Return IP information and reputation	ip, threshold	Yes	HelloWorld.IP
domain	Return domain information and reputation	domain, threshold	Yes	HelloWorld.Domain

Every command that needs to save results to context, should have outputs - read more here.
A very easy way to create outputs is using **demisto-sdk** command with the **generate-outputs** flag, read about it here

Classifiers and Mappers

*If your Content Pack includes integration(s), and they provide a **fetch incidents** functionality, it is recommended to provide at least one Incident Type in your Pack, and either a Classifier or a Mapper to help customers work with the incidents that your integration fetches. If your integration(s) provide the **fetch indicators** functionality (aka it is a feed integration), it is also recommended to provide Classifiers and Mappers for the Indicator Types you support (either out of the box ones, or custom). More details here.*

Classifier(s) names	HelloWorld
(only if the integration fetches incidents) **Supported Incident Types**	Hello World Alert
(only if the integration feeds indicators) **Supported Indicator types**	
Mapper(s) names	HelloWorld-mapper

Incidents Fields

If you are including custom Incident Fields and Layouts in your Content Pack, please list all the elements and dependencies here. More details here, here and here.
***It is recommended to use built-in XSOAR incident fields**. Only if needed you should consider adding new fields.*

*The following table should list all the custom Incident Fields and where they are used. Supported Incident Field types are: **Attachments, Boolean, Date Picker, Grid (table), HTML, Long Text, Markdown, Multi-select, Number, Role, Short Text, Single-select, Tags, Timer/SLA, URL, User.***

Incident Field Name	Incident Field Type	Used In *(Incident Types)*	Used In *(Layout, if custom)*
Hello World ID	Short Text	Hello World Alert	Hello World Layout
Hello World Status	Short Text	Hello World Alert	Hello World Layout
Hello World Type	Short Text	Hello World Alert	Hello World Layout

Indicators Fields

If you are including custom Indicator Types, Fields and Layouts in your Content Pack, please list all the elements and dependencies here.

The following table lists all the custom Indicator Fields and where they are used. Supported Indicator Field types are: ***Boolean, Date Picker, Grid (table), Long Text, Markdown, Multi-select, Number, Role, Short Text, Single-select, Tags, URL, User.***

Indicator Field Name	**Indicator Field Type**	**Used In** *(Indicator Types)*	**Used In** *(Layout, if custom)*

Playbooks

Fill in the following table the required information for each playbook that is part of the content pack (excluding the test playbook).
Playbook overview, task types, inputs and outputs

Playbook Name	Handle Hello World Alert
Description *(What it does)*	This playbook is used to retrieve and manage Alerts from Hello World. Once a Hello World incident is created, the playbook starts by retrieving additional details for the alert using the “helloworld-get-alert” command. Additional tasks can be optionally performed, as well as automatic enrichment of IP information is also used. Once the alert has been managed by the SOC, it is marked as closed in the Hello World system.
Triggers *(What triggers the incident - Incident Types, Feed, Sub-playbook)*	Incident: Hello World Alert
Steps (*The main steps and decisions that are part of the playbook)*	1. Retrieve extra alert information
Inputs *(comma separated list of the Inputs that this Playbook supports.)*	N/A
Outputs (output the playbook returns)	N/A
Dependencies *(optional)*	N/A

Playbook Name	Hello World Scan
Description *(What it does)*	This playbook is used to run a scan against a host and retrieve the report as a file or human readable context. Since the scan takes time to run, a polling mechanism is implemented.

Triggers *(What triggers the incident - Incident Types, Feed, Sub-playbook)*	Sub-Playbook
Steps (*The main steps and decisions that are part of the playbook)*	2. Start the scan on the hostname 3. Poll for the results of the scan using GenericPolling 4. Once the scan is completed, retrieve the results in JSON format
Inputs *(comma separated list of the Inputs that this Playbook supports.)*	hostname
Outputs (output the playbook returns)	Scan results using HelloWorld.Scan
Dependencies *(optional)*	

Scripts (Automations)

If your Content Pack includes Scripts (aka Automations), fill in the table below with general details of the scripts and what capabilities they provide. Fill in the Tags cells only if your scripts serve specific purposes that require them (i.e. Transformers, Dynamic Layout Sections, etc).

Script Name	**Description**	**Arguments** *(comma separated)*	**Tags** *(comma separated)*
HelloWorldScript	Hello command - prints hello to anyone.	name	

Widgets and Dashboards

Brief description here

Widget Name	Widget Type	Widget Description	Used In (Dashboard(s))

Dashboard Name	Description