

# Alien Vault OTX Search Function for IBM Resilient

---

## Table of Contents

- [About This Package](#)
  - [Prerequisites](#)
  - [Installation](#)
  - [Function Inputs](#)
  - [Function Output](#)
  - [Pre-Process Script](#)
  - [Post-Process Script](#)
  - [Rules](#)
- 

## About This Package:

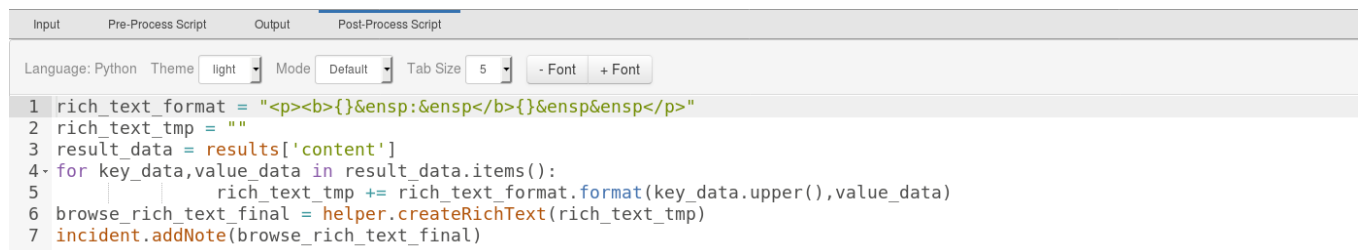
**This package contains a Resilient Function that allows you to search your Alien Vault OTX platform with the given query for Threat Intelligence data about a particular Threat Indicator**

- Threat intelligence indicators that can be searched for are:
  - IP Address
  - Domain
  - Host Name
  - File Hashes
  - URL
  - CVE
- The function makes use of the Alien Vault OTX [api/v1/indicators](#) API call to get information on a given query
- More information on [Alien Vault OTX](#)

Sample Function layout:

## Sample Pre-Process Script

## Sample Post-Process Script



```

1 rich_text_format = "<p><b>{}&ensp;&ensp</b>{}&ensp&ensp</p>"
2 rich_text_tmp = ""
3 result_data = results['content']
4 for key_data,value_data in result_data.items():
5     rich_text_tmp += rich_text_format.format(key_data.upper(),value_data)
6 browse_rich_text_final = helper.createRichText(rich_text_tmp)
7 incident.addNote(browse_rich_text_final)

```

## Prerequisites:

- Resilient Appliance >= v31.0.0
- Integrations Server running resilient\_circuits >= v30.0.0
- Account with [Alien Vault OTX](#)
- A [DirectConnect OTX](#) API Key from Alien Vault

## Installation

This package requires that it is installed on a RHEL or CentOS platform and uses the resilient-circuits framework.

- Install this package using **pip**:
- Download the **.zip** file from our App Exchange and extract it. You will find a file called:  
**fn\_alienvault\_otx-<version>.tar.gz**
- Copy this file to your Integrations Server
- To install the package, run:

```
$ pip install pip install fn_alienvault_otx-<version>.tar.gz
```

- To import the function, example rules and workflows into your Resilient Appliance, run:

```
$ resilient-circuits customize -y -l fn-alienvault-otx
```

- To update your **app.config** file with the required Alien Vault configurations, run:

```
$ resilient-circuits config -u
```

- Then open your **app.config** file and the following configuration data is added:

```

[fn_alienvault_otx]
# OTX API Key to Access the Alien Vault OTX Service
av_api_key=<<DirectConnect OTX API Key>>

#Base URL Path of Alien Vault OTX
av_base_url=https://otx.alienvault.com/api/v1

```

```
# Proxy Server by Default will be None
proxy=None
```

- Run resilient-circuits:

```
$ resilient-circuits run
```

- To uninstall:

```
$ pip uninstall fn-alienvault-otx
```

## Function Inputs

Function Name	Type	Required	Example	Info
<code>alien_search_value</code>	<code>String</code>	Yes	<code>"192.168.0.1"</code>	The search value to send to Alien Vault OTX (may be any String that contains an IP Address, URL, Hash, Threat CVE ID, DNS Name, System Name.)
<code>alien_search_type</code>	<code>String</code>	Yes	<code>IP Address</code>	The search type to send to Alien Vault OTX (may be any String type can be an IP Address, URL, Hash, Threat CVE ID, DNS Name, System Name.)
<code>alien_section</code>	<code>select</code>	Yes	<code>reputation</code>	The section to search for Threat Intelligence Data from Alien Vault, this section may be different for different search type (may be any string general, geo, malware, reputation, url_list, passive_dns, http_scans etc)

## Function Output

- To see the output of each of the API calls for this Function, we recommend running `resilient-circuits` in `DEBUG` mode.
- To do this run:

```
$ resilient-circuits run --loglevel=DEBUG
```

## Sample Output Displayed on Incident Notes Section

**Notes**

Nitin Kandhare added a note to the Incident 03/05/2019 15:46

**INDICATOR** : rghost.net

**BASE\_INDICATOR** : {u'indicator': u'rghost.net', u'access\_reason': u'', u'access\_type': u'public', u'description': u'', u'id': 1581738780, u'title': u'', u'type': u'domain', u'content': u''}

**WHOIS** : http://whois.domaintools.com/rghost.net

**ALEXA** : http://www.alexa.com/siteinfo/rghost.net

**TYPE** : domain

**PULSE\_INFO** : {u'references': [], u'count': 0, u'pulses': []}

**SECTIONS** : [u'general', u'geo', u'url\_list', u'passive\_dns', u'malware', u'whois', u'http\_scans']

## Pre-Process Script

This example sets the **alienvault\_search\_value**, **alienvault\_search\_type**, **alienvault\_section** inputs to the value and type of the Artifacts and sections the user took action on

```
# The search value to send to Alien Vault OTX (may be any String that
contains an IP Address, URL, Hash,Threat CVE ID,DNS Name,System Name.)
inputs.alienvault_search_value = artifact.value
#The search type to send to Alien Vault OTX (may be any String type can be
an IP Address, URL, Hash,Threat CVE ID,DNS Name,System Name.)
inputs.alienvault_search_type = artifact.type
#The section to search for Threat Intelligence Data from Alien Vault, this
section may be different for different search type(may be any string
general, geo, malware,reputation, url_list, passive_dns, http_scans etc)
inputs.alienvault_section = rule.properties.alienvault_search_section_ip
```

## Post-Process Script

```
result_data = results['content']
for key_data,value_data in result_data.items():
    rich_text_tmp +=
    rich_text_format.format(key_data.upper(),value_data)browse_rich_text_final
= helper.createRichText(rich_text_tmp)
incident.addNote(browse_rich_text_final)
```

## Rules

Rule Name	Object Type	Workflow Triggered	Activity Fields
Example: Alien Vault - CVE Lookup	Artifact	Example: Alien Vault OTX CVESearch	Alien Vault Search Section CVE values : general

Rule Name	Object Type	Workflow Triggered	Activity Fields
Example: Alien Vault - DNS Name Lookup	Artifact	Example: Alien Vault OTX DNS Name	Alien Vault Search Section DNS Name values : general,geo,malware,url_list,passive_dns,whois,http_scans
Example: Alien Vault - File Hash Lookup	Artifact	Example: Alien Vault OTX Hash	Alien Vault Search Section Hash values : general,analysis
Example: Alien Vault - Host Name Lookup	Artifact	Example: Alien Vault OTX Host Name	Alien Vault Search Section Host Name values : general,geo,malware,url_list,passive_dns,http_scans
Example: Alien Vault - IP Address Lookup	Artifact	Example: Alien Vault OTX IP Address	Alien Vault Search Section IP Address values : general,reputation,geo,malware,url_list,passive_dns,http_scans
Example: Alien Vault - URL Lookup	Artifact	Example: Alien Vault OTX URL	Alien Vault Search Section URL values : general,url_list

## Using the Alien Vault OTX Function

- The Alien Vault Function can be called on artifact like IP Address, DNS Name, System Name, URL, URL Referer, Hashes, Threat CVE ID.
- After invoking a Rule on the Artifact, we need to choose the **Section** based on the Artifact
- For more info on what **section** refers to, please see: <https://otx.alienvault.com/api>