McAfee OpenDXL Functions for IBM Resilient

Table of Contents

Release Notes

v1.2.0

- App Host support

v1.1.0

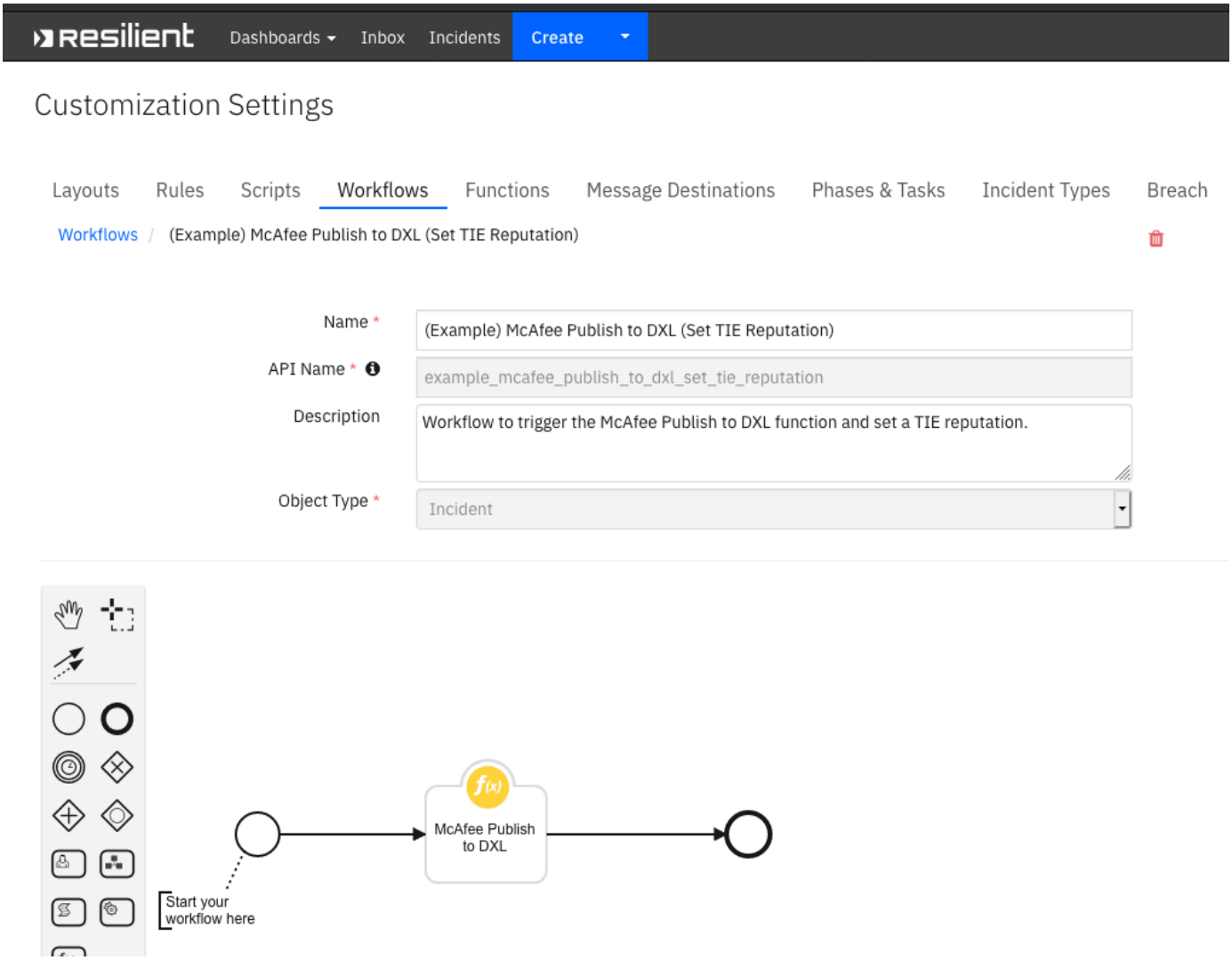
- Added Resilient Subscriber component

v1.0.0

- Initial Release

Overview

Resilient Circuits Components for McAfee publishing to DXL Functions



Resilient Circuits Components for McAfee publishing to DXL Functions

Key Features

- The McAfee Publish to DXL function contains the ability to publish a synchronous or asynchronous message to an event or a service.
- The McAfee DXL Subscriber listens on defined topics and maps the data to the Resilient platform to create incidents and artifacts.

Installation

Requirements

- Resilient platform >= `v35.0.0`
- An App Host or an Integration Server:
  - To setup up an App Host see: ibm.biz/res-app-host-setup
  - An Integration Server running `resilient_circuits>=30.0.0` (if using an Integration Server)
    - To set up an Integration Server see: ibm.biz/res-int-server-guide
    - If using an API key account, minimum required permissions are:

Name	Permissions
Org Data	Read
Incident	Create, Read All
Function	Read

- Proxy supported: No (OpenDXL does not support proxy in on-prem installations)

Install

- To install or uninstall an App using the App Host see ibm.biz/res-install-app
- To install or uninstall an Integration using the Integration Server see the ibm.biz/res-install-int

---

App Configuration

The following table describes the settings you need to configure in the app.config file. If using App Host, see the Resilient System Administrator Guide. If using the integration server, see the Integration Server Guide.

Config	Required	Example	Description
dxlclient_config	Yes	`/home/integration/.resilient/fn_mcafee_opendxl/dxlclient.config`	*Path to the dxlclient.config file*
topic_listener_on	Yes	`False`	*Boolean to turn ON/OFF Listener*
custom_template_dir	No	``	*Path to custom jinja template*

In addition to updating the app.config file and before running the McAfee OpenDXL functions, the dxlclient.config, certificates and key files must be created using a provisioning command. More information on the dxlclient.config file and provisioning the system can be found here:

https://opendxl.github.io/opendxl-client-python/pydoc/provisioningoverview.html https://opendxl.github.io/opendxl-client-python/pydoc/basicliprovisioning.html#basicliprovisioning

Here is an example of the dxlclient CLI provisioning command:

```
python -m dxlclient -vv provisionconfig /home/integration/.resilient/fn_mcafee_opendxl X.X.X.X client1 -u admin -p password
```

In this example, `X.X.X.X` is the IP address of the McAfee ePO server or OpenDXL Broker.

The generated files are created in the /home/integration/.resilient/fn_mcafee_tie directory.

If installing the app on an integration server, set the dxlclient_config app.config parameter to the location of the created dxlclient.config file.

```
[fn_mcafee_opendxl]
 dxlclient_config=/home/integration/.resilient/fn_mcafee_opendxl/dxlclient.config
```

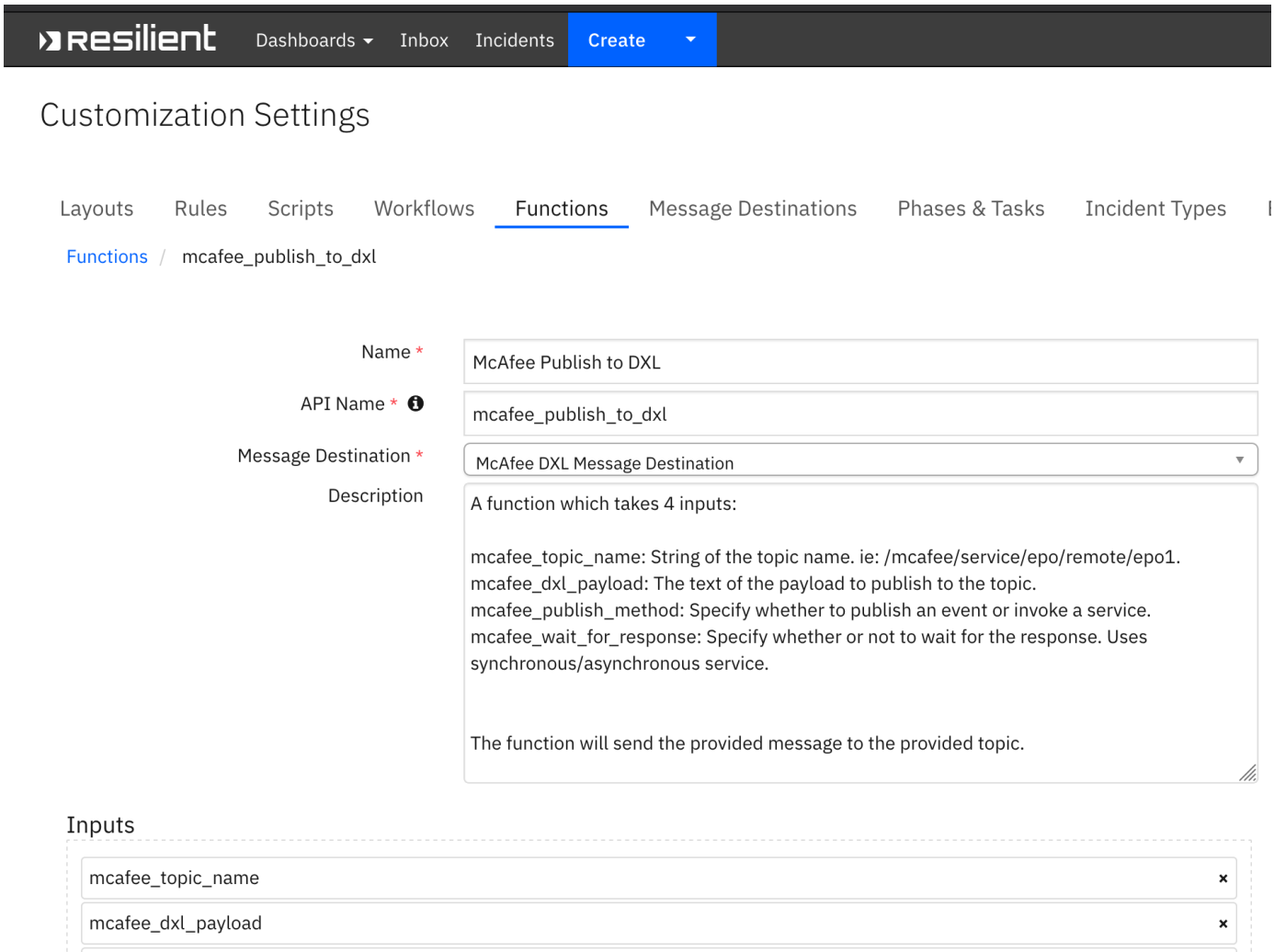
If installing the app directly to the Resilient platform (App Host environment), you need to use the New File button to create each file that was created by the provisioning command. As you create each file, copy the contents of the file into your new file. Make sure to enter `/etc/rescircuits/fn_mcafee_opendxl` as the File Path. See the Resilient Platform System Administrator Guide for the detailed procedure.

Function - McAfee Publish to DXL

A function which takes 4 inputs:

mcafee_topic_name: String of the topic name. ie: /mcafee/service/epo/remote/epo1. mcafee_dxl_payload: The text of the payload to publish to the topic. mcafee_publish_method: Specify whether to publish an event or invoke a service. mcafee_wait_for_response: Specify whether or not to wait for the response. Uses synchronous/asynchronous service.

The function will send the provided message to the provided topic.



▶ Inputs:

Name	Type	Required	Example	Tooltip
mcafee_dxl_payload	text	Yes	–	The text of the payload to publish to the topic
mcafee_publish_method	select	Yes	–	Specify whether to publish an event or invoke a service
mcafee_topic_name	text	Yes	–	String of the topic name. ie: /mcafee/service/epo/remote/epo1
mcafee_wait_for_response	select	No	–	Specify whether or not to wait for the response. Uses synchronous/asynchronous service

▶ Outputs:

```
results = {
    {'version': '1.0',
     'success': True,
     'reason': None,
     'content': {'mcafee_topic_name': '/mcafee/service/epo/remote/epo1',
                 'mcafee_dxl_payload': '{"command": "system.applyTag",
                                         "output": "json",
                                         "params": {"names": "10.0.2.15", "tagName": "Shut
Down"}}',
                 'mcafee_publish_method': 'Service',
                 'mcafee_wait_for_response': 'Yes',
                 'response': {'_version': '2',
                              '_message_id': '{eb976a7f-2051-43f7-bd13-0205630385a7}',
```

```
                                '_source_client_id': '',
                                '_source_broker_id': '',
                                '_destination_topic': '',
                                '_payload': '',
                                '_broker_ids': [],
                                '_client_ids': [],
                                '_other_fields': {},
                                '_source_tenant_guid': '',
                                '_destination_tenant_guids': [],
                                '_request': None,
                                '_request_message_id': None,
                                '_service_id': ''}},
        'raw': '{"mcafee_topic_name": "/mcafee/service/epo/remote/epo1", "mcafee_dxl_payload": "
{\\"command\\": \\"system.applyTag\\", \\"output\\": \\"json\\", \\"params\\":       {\\"names\\":
\\"10.0.2.15\\", \\"tagName\\": \\"Shut Down\\"}}", "mcafee_publish_method": "Service",
"mcafee_wait_for_response": "Yes", "response": {"_version": "2", "_message_id": "{eb976a7f-2051-
43f7-bd13-0205630385a7}", "_source_client_id": "", "_source_broker_id": "", "_destination_topic":
"", "_payload": "", "_broker_ids": [], "_client_ids": [], "_other_fields": {},
"_source_tenant_guid": "", "_destination_tenant_guids": [], "_request": null,
"_request_message_id": null, "_service_id": ""}}',

        'inputs': {'mcafee_publish_method': {'id': 305, 'name': 'Service'},
                   'mcafee_topic_name': '/mcafee/service/epo/remote/epo1',
                   'mcafee_dxl_payload': '{"command": "system.applyTag", "output": "json",
"params": {"names": "10.0.2.15", "tagName": "Shut Down"}}',
                   'mcafee_wait_for_response': {'id': 302, 'name': 'Yes'}},

        'metrics': {'version': '1.0',
                    'package': 'fn-mcafee-opendxl',
                    'package_version': '1.2.0',
                    'host': 'MacBook-Pro.local',
                    'execution_time_ms': 2534,
                    'timestamp': '2020-10-20 17:34:14'},
        'mcafee_topic_name': '/mcafee/service/epo/remote/epo1',
        'mcafee_dxl_payload': '{"command": "system.applyTag", "output": "json", "params":
{"names": "10.0.2.15", "tagName": "Shut Down"}}',
        'mcafee_publish_method': 'Service',
        'mcafee_wait_for_response': 'Yes'}
}
```

▶ Example Pre-Process Script:

```
# Replaces trust level string with acceptable value to publish to topic

inputs.mcafee_dxl_payload = inputs.mcafee_dxl_payload.replace("\"Known Malicious\"", "1")

inputs.mcafee_dxl_payload = inputs.mcafee_dxl_payload.replace("\"Most Likely Malicious\"", "15")

inputs.mcafee_dxl_payload = inputs.mcafee_dxl_payload.replace("\"Might Be Malicious\"", "30")
```

▶ Example Post-Process Script:

```
"""
Response returned provides the input values in the following format
{
  "mcafee_topic_name": "<topic_name>",
  "mcafee_dxl_payload": "<payload>",
  "mcafee_publish_method": "<method>",
  "mcafee_wait_for_response": "<wait for response>"
"""

trust_level = ""

content = results.get("content")
```

```
if content.get("mcafee_dxl_payload").find("30") > 0:
  trust_level = "Might Be Malicious"

elif content.get("mcafee_dxl_payload").find("15") > 0:
  trust_level = "Most Likely Malicious"

elif content.get("mcafee_dxl_payload").find("1") > 0:
  trust_level = "Known Malicious"



text = """The following was published to DXL:<br>
<b>Payload:</b> {}<br>
<b>Topic:</b> {}<br>
<b>Method:</b> {}<br>

Setting Trust Level to {}
""".format(content.get("mcafee_dxl_payload"), content.get("mcafee_topic_name"),
content.get("mcafee_publish_method"), trust_level)

noteText = helper.createRichText(text)
incident.addNote(noteText)
```

Rules

Rule Name	Object	Workflow Triggered
(Example) McAfee Publish to DXL (Set TIE Reputation Known Malicious)	incident	`example_mcafee_publish_to_dxl_set_tie_reputation`
(Example) McAfee Publish to DXL (Tag System Shut Down)	incident	`example_mcafee_publish_to_dxl_tag_system`

DXL Subscriber

The DXL subscriber is designed using Resilient Circuits but does not rely on the functions capabilities. The subscriber connects to the Data Exchange Layer and listens on the topic specified topic(s). When a message is sent to the topic, the integration uses a mapping template to map the data into a Resilient incident DTO and create incidents and artifacts within the Resilient platform.

To use the DXL Subscriber, set the topic_listener_on configuration parameter to True.

When you run Resilient Circuits, the subscriber listens on the default topic, /mcafee/event/epo/threat/response, and uses the default provided jinja template to map incident and artifact data into the Resilient Platform.

Troubleshooting & Support

If using the app with an App Host, see the Resilient System Administrator Guide and the App Host Deployment Guide for troubleshooting procedures. You can find these guides on the IBM Knowledge Center, where you can select which version of the Resilient platform you are using.

If using the app with an integration server, see the Integration Server Guide

For Support

This is an IBM Supported app. Please search https://ibm.com/mysupport for assistance.