



# Incident Response Platform Integrations

# Splunk Function V1.0.0

Release Date: April 2018

Resilient Functions simplify development of the integrations by sending data from the Resilient platform to a remote program that performs an activity then returns the results to the function. The results can be acted upon by a script and the result of that becomes a decision point in the Resilient workflow.

This guide describes the Function.

Overview

The Splunk function, fn\_splunk\_integration, provides an automated way of managing bi-directional actions between artifact items in Resilient incidents and intelligence items in Splunk threat collections.

The integration with the Resilient platform package provides the followings:

* A search function that perform a query
* An update function to change the status of a ES notable event
* An add function to add a new threat intelligence item to a given Splunk collection
* A delete function to disable a threat intelligence item from a given Splunk collection

Together with the above function, this package includes also the example workflows that demonstrate how to call those functions, rules that start the example workflows, custom fields, and datatables updated by the example workflows.

Setup

The following lists the system requirements:

* Python version 2.7.10 or later, or version 3.6 or later
* Resilient Circuits and Resilient Python libraries version 30.0 or later
* Resilient platform version 30.0 or later
* version 6.6 or later
* ES 4.7.2 or later (only required for the function to update Splunk ES notable event)

Perform the following to install and configure the function:

1. Ensure the environment is up to date:

sudo pip install --upgrade pip

sudo pip install --upgrade setuptools

sudo pip install --upgrade resilient-circuits

1. Install the required software for the function (if not already installed):

sudo pip install fn\_splunk\_integration-<*version*>.tar.gz

1. Add the function to the Resilient platform:

resilient-circuits customize

You will be prompted to import functions, message destinations, workflow, etc.

1. From the account used for Integrations, use one of the following commands to configure the settings. Use –c to create new environments or –u to update existing environments.

resilient-circuits config -c

OR

resilient-circuits config -u

1. Edit the .resilient/app.config file and section [fn\_splunk\_integration]:

url=<*splunk url*>

port=<8089 or the customized port>

username=<*splunk access user*>

splunkpassword=<*splunk access password, key-ring protection recommended*>

verify\_cert=[True|False]

Use False for self-signed certificates.

After completing the configuration steps, enter the resilient-circuits run command. The following is an example of the resulting messages indicating the successful connection to the Resilient platform and the loading of the integration modules.

$ resilient-circuits run

2018-04-07 12:38:04,164 INFO [app] Configuration file: /Users/Integration/.resilient/app.config

2018-04-07 12:38:04,165 INFO [app] Resilient server: <host>

2018-04-07 12:38:04,165 INFO [app] Resilient user: <acct>

2018-04-07 12:38:04,165 INFO [app] Resilient org: <org>

2018-04-07 12:38:04,165 INFO [app] Logging Level: INFO

…

2018-04-07 12:38:05,418 INFO [component\_loader] 'fn\_splunk\_integration.components.splunk\_delete\_threat\_intel\_item.FunctionComponent' loading

2018-04-07 12:38:05,419 INFO [component\_loader] 'fn\_splunk\_integration.components.splunk\_search.FunctionComponent' loading

2018-04-07 12:38:05,420 INFO [component\_loader] 'fn\_splunk\_integration.components.splunk\_update\_notable.FunctionComponent' loading

…

2018-04-07 12:38:05,435 INFO [actions\_component] 'fn\_splunk\_integration.components.splunk\_search.FunctionComponent' function 'splunk\_search' registered to 'splunk\_search'

2018-04-07 12:38:05,436 INFO [actions\_component] 'fn\_splunk\_integration.components.splunk\_update\_notable.FunctionComponent' function 'splunk\_update\_notable' registered to 'splunk\_es\_notable'

2018-04-07 12:38:05,437 INFO [actions\_component] 'fn\_splunk\_integration.components.splunk\_add\_intel\_item.FunctionComponent' function 'splunk\_add\_intel\_item' registered to 'splunk\_es\_rest'

…

2018-04-07 12:38:05,729 INFO [actions\_component] Subscribe to message destination 'splunk\_search'

…

2018-04-07 12:38:05,731 INFO [stomp\_component] Subscribe to message destination actions.<org id>.fn\_splunk\_integration

…

Resilient Platform Configuration

In the Customization Settings section of the Resilient platform, you can verify that the following specific functions, workflows, datatable, and rules are available in the Resilient platform by clicking their respective tabs.

Here are the details about how each function is used in the example workflow and rule.

Splunk Search

This function performs query to fetch data from server.

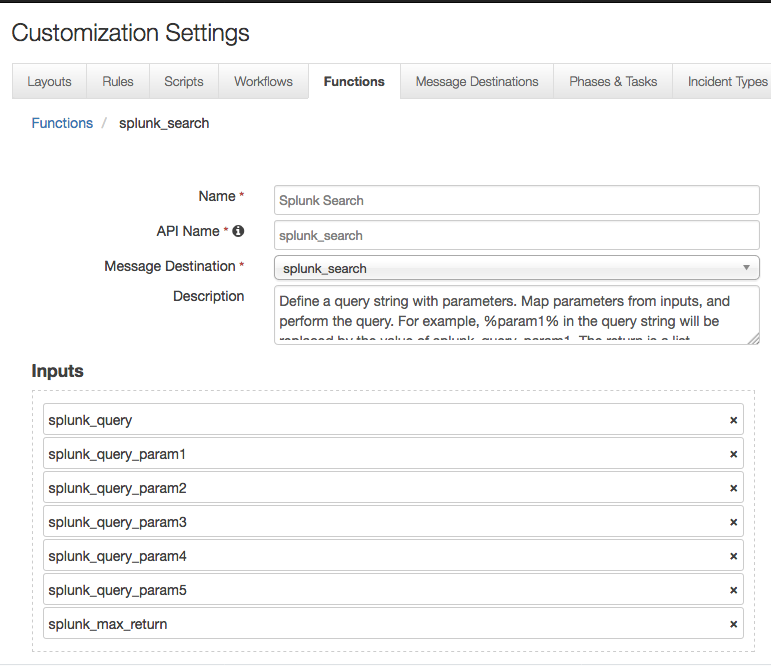


Figure : Splunk Search

As shown above, this function takes the following parameters:

* splunk\_query is the query to perform. It contains demo template queries the you can select from within the calling workflow. The demo queries contain parameters which will be replaced by the splunk\_query\_param[n] below. For example, one demo query is: SELECT %param1% FROM events WHERE username=%param2% LAST %param3% MINUTES. Then users can set values for splunk\_query\_param1, splunk\_query\_param2, and splunk\_query\_param3 in the workflow.
* splunk\_query\_param[n]: parameters used in the query
* splunk\_max\_return: specify how many events to return from .

The example workflow (object type = Artifact) that calls this function is “Example of searching ES ip\_intel”. It is an artifact workflow. The Input tab of this function is shown in Figure 2. It shows that the “%param1% in the query is mapped to “ip\_intel”, etc.

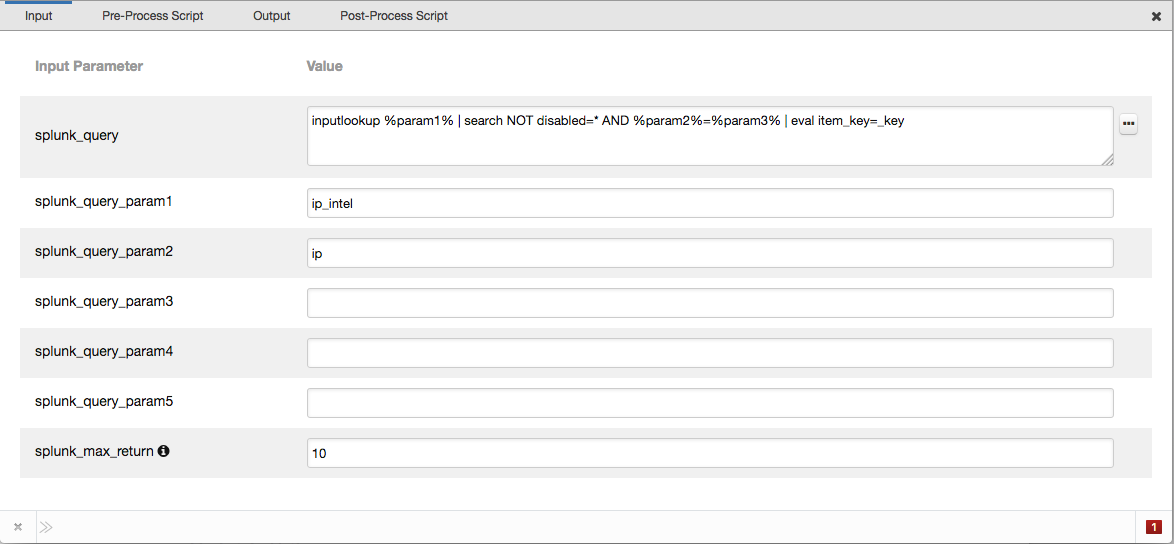


Figure : Example of searching Splunk ES ip\_intel

In the Pre-Process Script, the “%param3% is set to the value of the artifact associated with this workflow as shown in Figure 3.

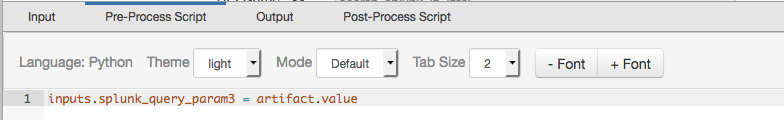


Figure : Pre-Process Script

A Menu Item rule called “Search ES ip\_intel” is also included. This rule calls the workflow above.

/Users/yongjian.feng@ibm.com/Desktop/Screen Shot 2018-04-12 at 1.26.33 PM.png

Figure : Rule

With these components in place, if an IP Address artifact is now added, users can then click the Actions button, and the above rule will appear as shown in Figure 5. By clicking the menu item, this search will be activated. The search result from will be used to update the custom datatable called “splunk\_ip\_intel” shown in Figure 5. This definition of this datatable is also included in the package.

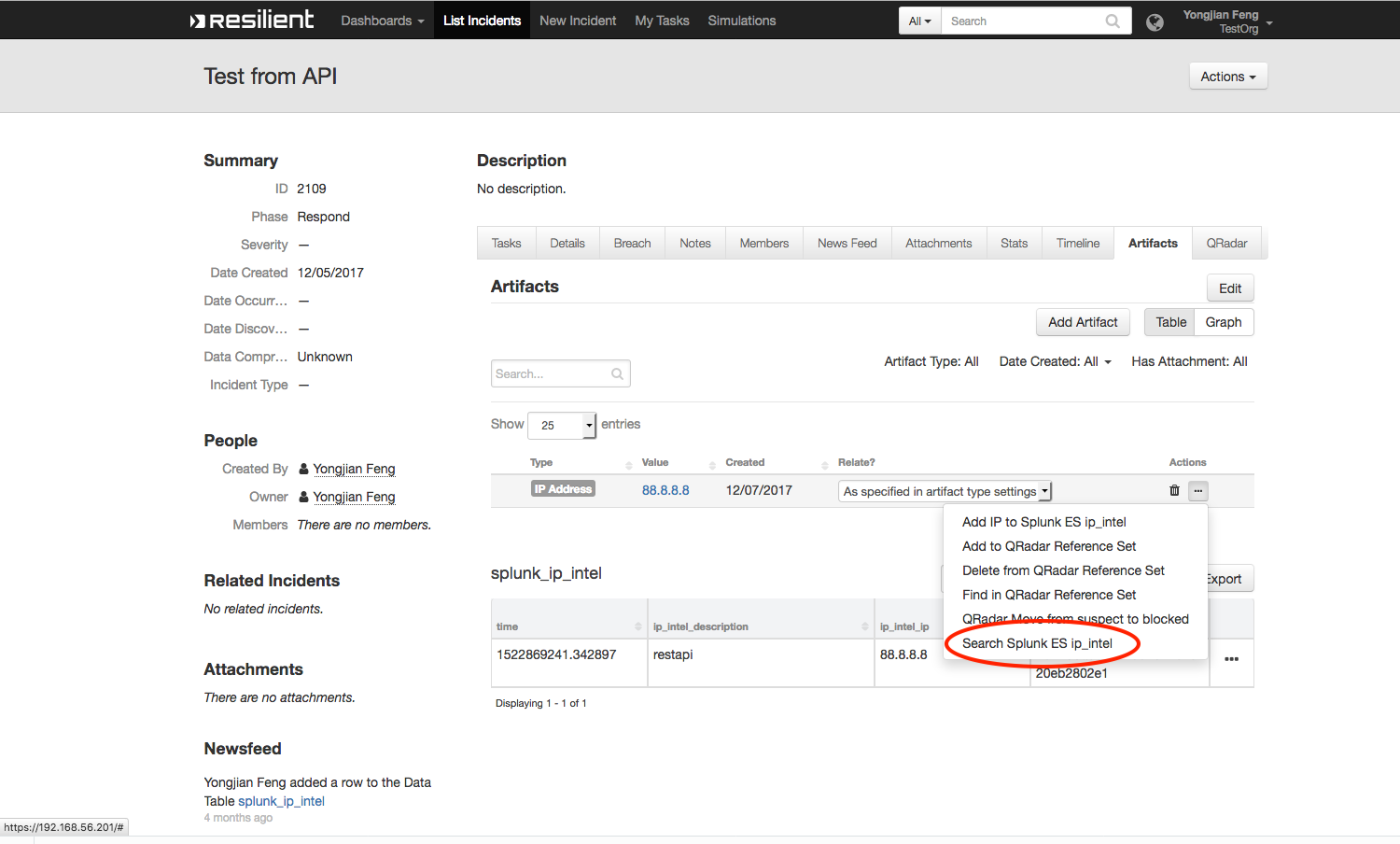


Figure : Menu item

Add Intelligence Item

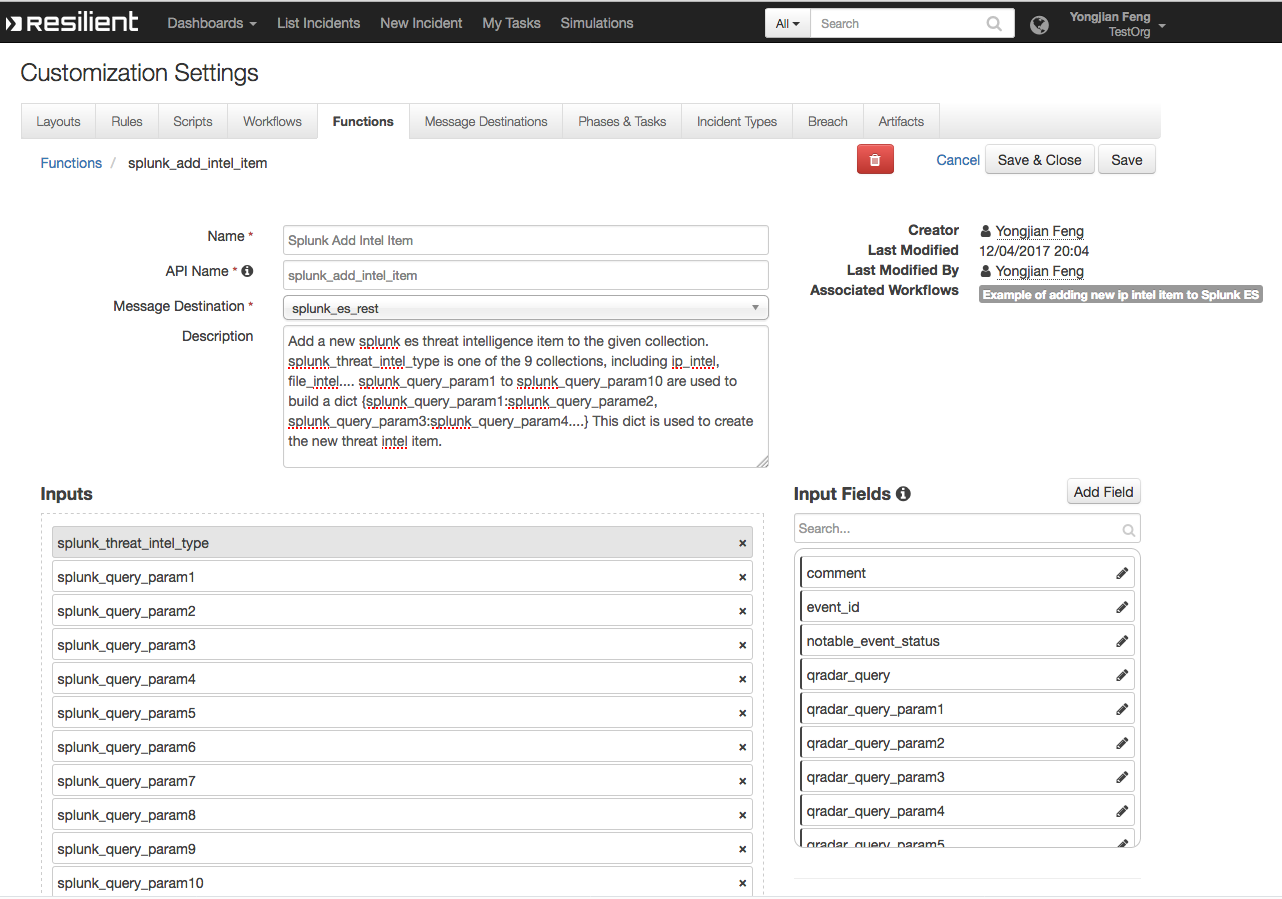
This function adds a new threat intelligence item to a given collection. 

Figure : Splunk Add Intelligence Item

Here, splunk\_threat\_intel\_type is the name of the Splunk threat intelligence collection, and splunk\_query\_param1 to splunk\_query\_param10 are inputs used to create a python dictionary that will add a new threat intelligence item to a given collection.

In the example workflow for artifact, we set splunk\_threat\_intel\_type to be ip\_intel, and slunk\_query\_param1 to be “ip” for the Input. In the Pre-Process Script, we set splunk\_query\_param2 to be the value of the associated artifact. This will create a python dictionary: {“ip”: “the\_associated\_artifact\_value”}, and a new item will be added to the ip\_intel collection.

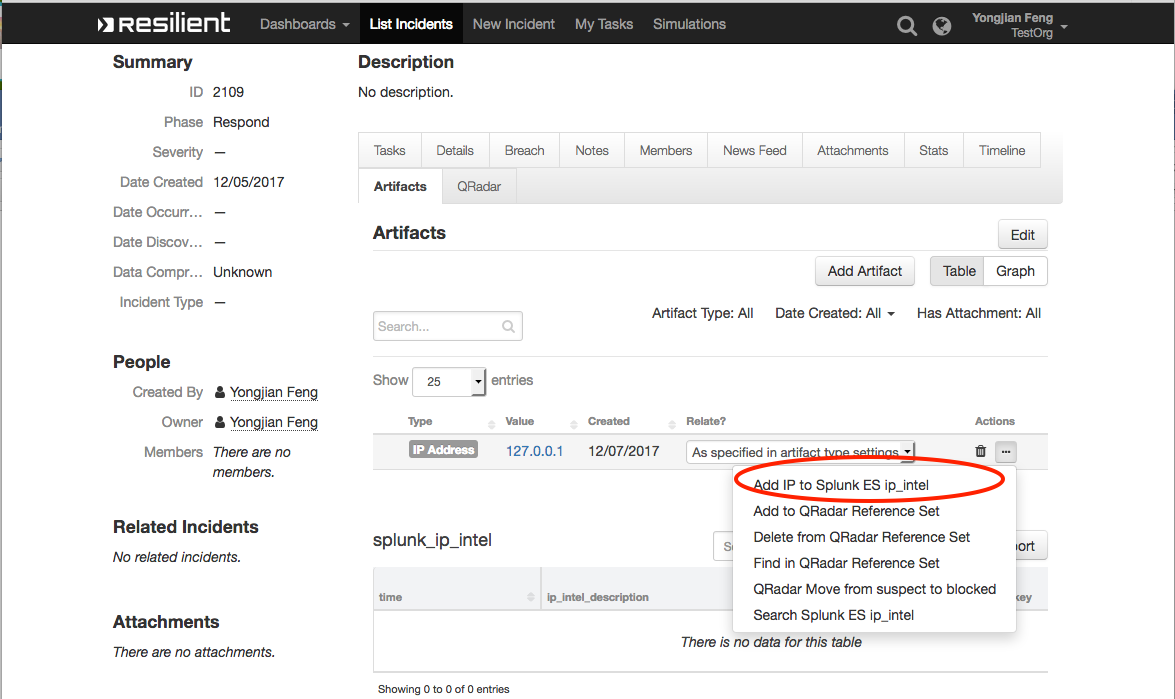
An example rule “Add IP to ES ip\_intel” calls this example workflow. As a result, a user can click on this menu item to add an IP Address artifact to the ip\_intel collection of ES.

Figure : Rule and Menu Item

Splunk Delete Intelligence Item

This function is used to disable a threat intelligence item from a given collection. A workflow “Example of deleting a Splunk ES ip\_intel item” calls this function, and is activated by a rule called “Delete IP from Splunk ES ip\_intel”.

The rule is a menu item to a row in the included datatable. As shown in Figure 8, a row contains the item\_key corresponding to this intelligence item. This menu item calls the function to delete the item associated with that item\_key.

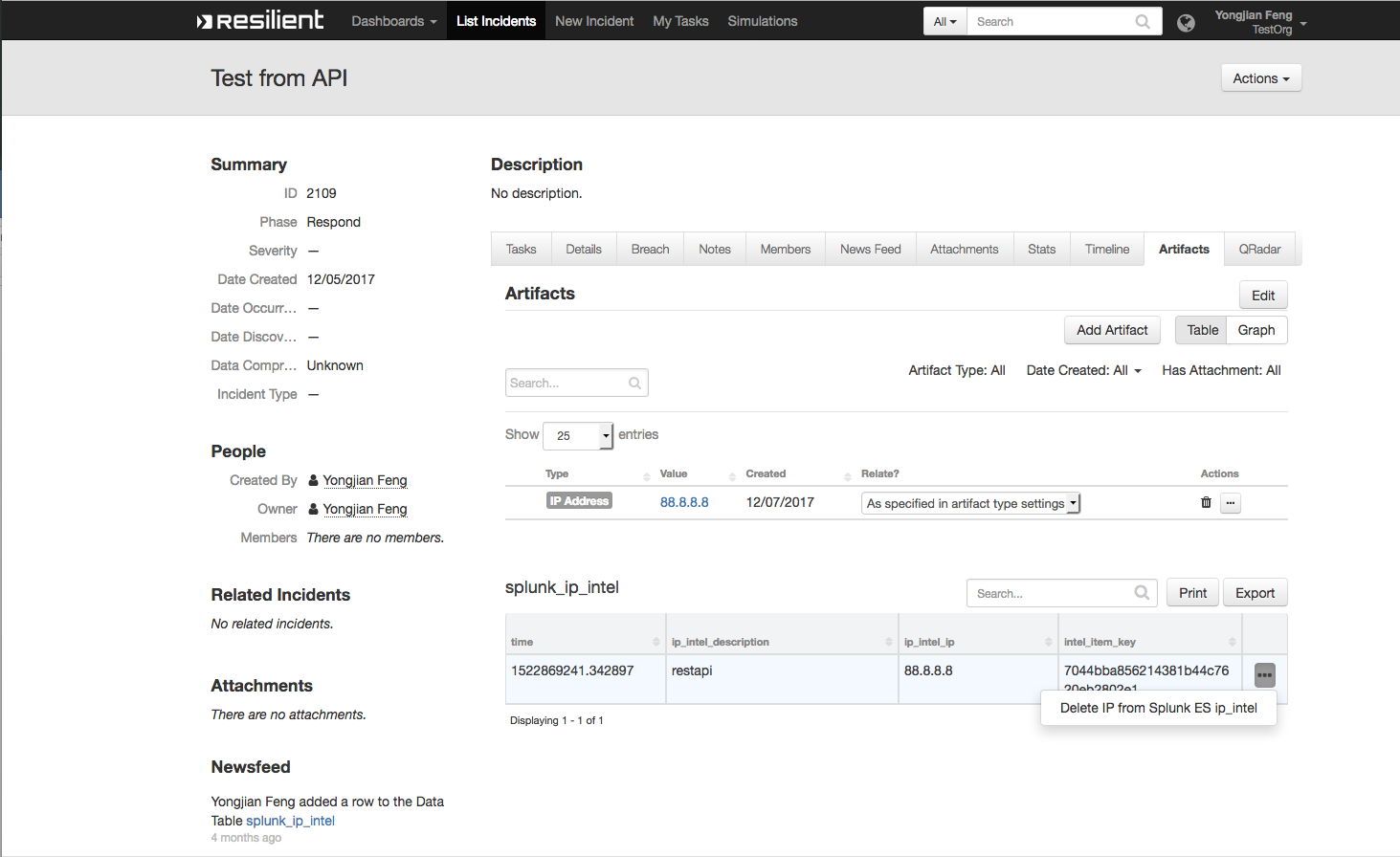


Figure 8: Datatable row with data including intel\_item\_key

Update ES Notable Event

This function updates the status and comment of a given notable event, using the event\_id stored in an incident. This function can be used together with the “Resilient Integration for Splunk and Splunk ES” addon.

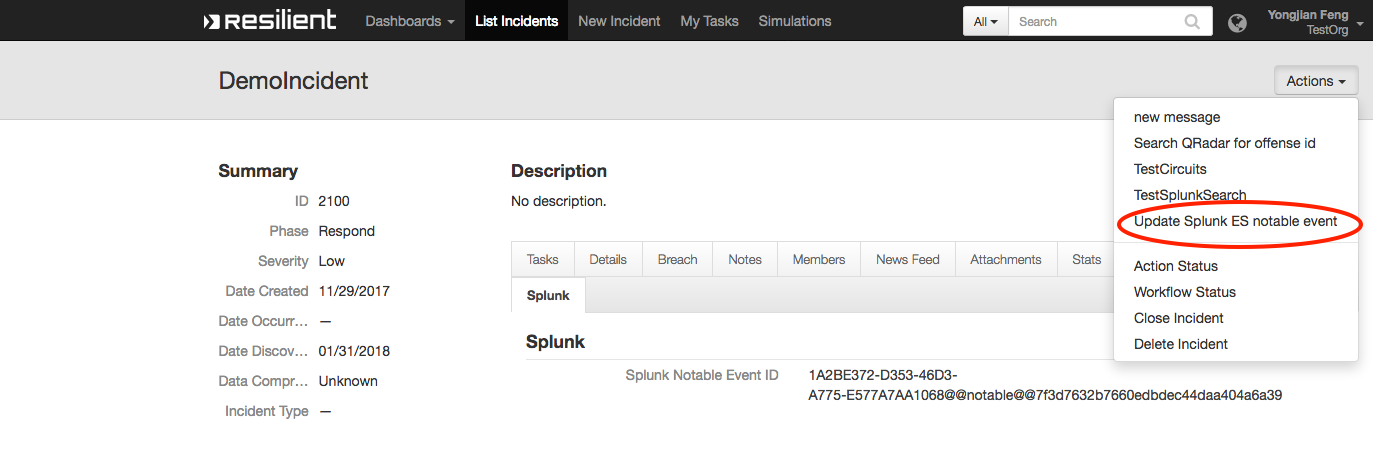
For an incident escalated from the “Resilient Integration for Splunk and Splunk ES” addon, it contains a custom property called splunk\_notable\_event\_id. In the workflow, the status of the incident is mapped to the status of notable event. Also, a comment is given in the Input tab. As a result, this menu item will update the notable event identified by this event id accordingly. 

Figure : Update Splunk ES Notable Event

Troubleshooting

There are several ways to verify the successful operation of a function.

* Resilient Action Status

When viewing an incident, use the Actions menu to view Action Status. By default, pending and errors are displayed. Modify the filter for actions to also show Completed actions. Clicking on an action displays additional information on the progress made or what error occurred.

* Resilient Workflow Status

Check workflow status for any error.

* Resilient Scripting Log

A separate log file is available to review scripting errors. This is useful when issues occur in the pre-processing or post-processing scripts. The default location for this log file is:

/var/log/resilient-scripting/resilient-scripting.log

* Resilient Logs

By default, Resilient logs are retained at /usr/share/co3/logs. The client.log may contain additional information regarding the execution of functions.

* Resilient-Circuits

The log is controlled in the .resilient/app.config file under the section [resilient] and the property logdir. The default file name is app.log. Each function creates progress information. Failures show up as errors and may contain Python trace statements.

Support

For additional support, contact [support@resilientsystems.com](mailto:support@resilientsystems.com).

Including relevant information from the log files will help us resolve your issue.