



# Security Orchestration, Automation and Response Platform

# <Function Name> Function V1.0.0

Release Date: <month> 2019

Resilient functions simplify development of integrations by wrapping each activity into an individual workflow component. These components can be easily installed, then used and combined in Resilient workflows. The Resilient platform sends data to the function component that performs an activity then returns the results to the workflow. The results can be acted upon by scripts, rules, and workflow decision points to dynamically orchestrate the security incident response activities.

About this template (delete when done)

Use this template to create a User Guide for your Resilient function. It is designed to be used in conjunction with the Resilient *Integration Server Guide*. Examples are provided. This guide contains the following sections:

* **Overview**: Describe the purpose of the function package.
* **Installation**: List any special installation steps or requirements above and beyond those requirements listed in the *Integration Server Guide*.
* **Package contents**: List the functions and each component (workflow, rules, and so on).
* **Custom layout**: Only needed if you require the Resilient user to create custom fields or data tables. Provide a recommendation on where to place these components – whether a new or existing incident layout.
* **Function <name> description**: Provide a detailed description of how the function works with workflows, rules, fields, data tables and so on. Provide screenshots if you feel they will be helpful. If you have multiple functions in one package, create a section for each function.

When done and before making a PDF, change the title of your Word file in **File>Info> Advanced Properties**.

If writing a readme for the online community, make it short. Describe the purpose of the function package and point to your guide for details.

If you have a complex function or integration requiring a lot of documentation and you wish to have a Table of Contents, use the [Template for Resilient Integrations User Guide](https://github.com/ibmresilient/resilient-reference/blob/master/developer_guides/Template%20for%20Resilient%20Integrations%20User%20Guide.docx).

Overview

***<Provide a high-level description of the function itself and its remote software or application. The following text is an example.>***

The McAfee ePO function contains the ability to apply a tag to a system managed in ePO from the Resilient platform.

This guide provides a description of the functions and components within the function package, any additional requirements, and a list of settings that need to be added to the Resilient Circuits app.config file.

Installation

You download the function package to a Resilient integration server, and from there you deploy the functions and components to a Resilient platform. These procedures are provided in the [Resilient Integration Server Guide (PDF)](https://github.com/ibmresilient/resilient-reference/blob/master/developer_guides/Integration%20Server%20Guide.pdf).

The functions included this package have the following requirements, which are above and beyond those listed in the *Resilient* *Integration Server Guide*.

* Resilient platform is version ***xx*** or later.
* <*is a new incident tab needed in the Layouts section of the Resilient platform to contain the custom fields and data tables?*>
* <*is there a special port that needs to be opened?*>

After installing the package, Resilient Circuits creates a new section, <*name of section*>, in the app.config file. You need to edit the following settings in that section.

***<Replace the following example with the section created for your function>***

[fn\_microsoft\_security\_graph]

# Graph URL with version number  
microsoft\_graph\_url=https://graph.microsoft.com/v1.0/  
tenant\_id=<Tenant directory id>  
client\_id=<App client id>  
client\_secret=<App client secret>

***<Provide details of each setting as necessary>***

Package contents

The following table lists the functions included in the package, along with associated workflows and rules.

|  |  |  |
| --- | --- | --- |
| **Function** | **Workflow** | **Rule** |
| Function name | Workflow name | Rule name |
| Function name | Workflow name | Rule name |
| Workflow name | Rule name |

***<Provide a list of custom incident fields and data tables.>***

***<NOTE: Describe the function’s input fields in the Function <name> description section.>***

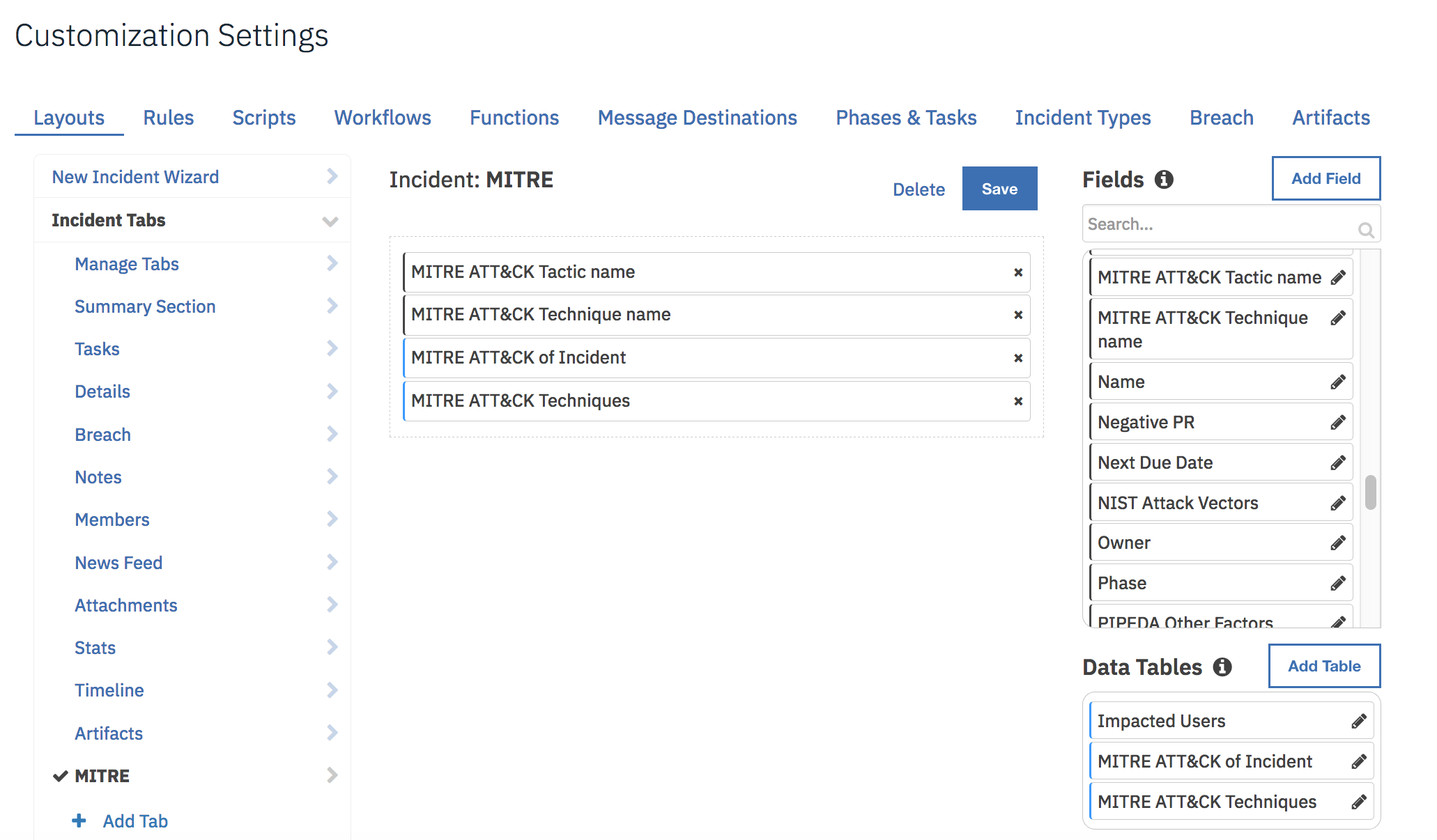
The package also requires that the following custom fields and data tables are created in the Resilient platform:

* Field name
* Field name
* Data table name
* Data table name

Custom layout

***<Use this section to provide guidance on where the user should add any custom fields and data tables. You may wish to recommend a new incident tab, as shown in the following example. Delete this section if not applicable.>***

To use the functions, the Resilient playbook designer needs to create a new Incident tab containing the custom fields and data tables. The examples in this guide assume that the incident tab is named MITRE. For example:



<Function Name> description

***<Provide a detailed description of the function, its inputs, and pre-process and post-process scripts. Here is a simple example.>***

This function adds a new item to an existing QRadar reference set. It uses two input parameters:

* qradar\_reference\_set\_name is the name of an existing reference set in QRadar
* qradar\_reference\_set\_item\_value is the value to be added.

The input is populated by the workflow, “Example of adding an item to QRadar reference set”.

The workflow, “Example of adding an item to QRadar reference set”, sets the function’s input fields:

* “qradar\_reference\_set\_name” is mapped to “Sample Blocked IPs”
* qradar\_reference\_set\_item\_value is mapped to the artifact value, and then runs the function.

The workflow is initiated by the rule, “QRadar Add to Reference Set”.

The example rule, “QRadar Add to Reference Set”, is a menu item rule for an artifact. The user can select this action in the menu to initiate the workflow.

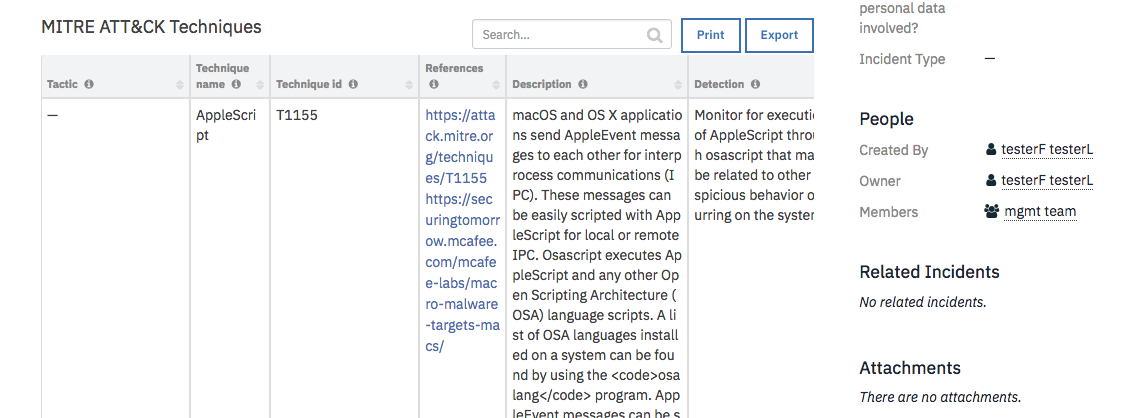
<Function Name> description

***<Provide a detailed description of the function, its inputs, and pre-process and post-process scripts. Here is a more complex example.>***

This function retrieves ATT&CK information on the given MITRE technique from the MITRE STIX TAXII server. To retrieve information on a MITRE technique from within an incident:

1. Open an incident and select the MITRE tab.
2. Edit the custom field, “MITRE ATT&CK Technique name” and enter a technique name, for example, “AppleScript”.
3. Click **Actions->Get MITRE technique information**.

This invokes the “Example of getting MITRE technique information” workflow, which calls the “MITRE Technique Information” function. The workflow uses the results to populate the “MITRE ATT&CK Techniques” data table. For example:



The flag, “mitre\_technique\_mitigation\_only” is only pertinent to the rule, “Create Task for MITRE ATT&CK technique”, which is described later in this guide.