



# Incident Response Platform Integrations

# McAfee ESM Functions and Case Polling Integration V1.0.0

Release Date: October 2018

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.

This guide describes the McAfee ESM Functions and the ESM Case Polling Integration.

Overview

The McAfee ESM functions contain the ability to call multiple API endpoints within ESM while the Case Polling Integration allows for creation of new Incidents in the Resilient platform.

This document describes the McAfee ESM functions, its customization options, and how to configure them in custom workflows.

Installation

Before installing, verify that your environment meets the following prerequisites:

* Resilient platform is version 30 or later.
* You have a Resilient account to use for the integrations. This can be any account that has the permission to view and modify administrator and customization settings, and read and update incidents. You need to know the account username and password.
* You have access to the command line of the Resilient appliance, which hosts the Resilient platform; or to a separate integration server where you will deploy and run the functions code. If using a separate integration server, you must install Python version 2.7.10 or later, or version 3.6 or later, and “pip”. (The Resilient appliance is preconfigured with a suitable version of Python).

Install the Python components

The functions package contains Python components that will be called by the Resilient platform to execute the functions during your workflows. These components run in the ‘resilient-circuits’ integration framework.

The package also includes Resilient customizations that will be imported into the platform later.

Ensure that the environment is up to date,

sudo pip install --upgrade pip

sudo pip install --upgrade setuptools

sudo pip install --upgrade resilient-circuits

To install the package,

sudo pip install --upgrade fn\_mcafee\_epo-*<*1.0.0*>*.tar.gz

Configure the Python components

The ‘resilient-circuits’ components run as an unprivileged user, typically named `integration`. If you do not already have an `integration` user configured on your appliance, create it now.

Perform the following to configure and run the integration:

1. Using ‘sudo’, become the integration user.

sudo su - integration

1. Create or update the resilient-circuits configuration file.

resilient-circuits config -c

or

resilient-circuits config -u

1. Edit the resilient-circuits configuration file.
   1. In the [resilient] section, ensure that you provide all the information needed to connect to the Resilient platform.
   2. In the [fn\_mcafee\_epo] section, edit the settings as follows:

ePO\_url=https://<your\_epo\_server>:<port>

epo\_username=<your\_epo\_username>

epo\_password=<your\_epo\_password>

epo\_trust\_cert=[true|false]

Use false for self-signed SSL certificates.

Deploy customizations to the Resilient platform

The package contains the function definition that you can use in workflows, and an example workflow and rule that show how to use the function.

Install these customizations to the Resilient platform with the following command:

resilient-circuits customize

Answer the prompts to deploy the function, message destination, workflow and rule. The following data will be imported.

Function inputs: mcafee\_esm\_alarm\_triggered\_end\_time, mcafee\_esm\_alarm\_triggered\_start\_time, mcafee\_esm\_alarm\_triggered\_time\_range, mcafee\_esm\_case\_id, mcafee\_esm\_edit\_case\_json, mcafee\_esm\_qry\_config, mcafee\_esm\_qry\_event\_type, mcafee\_event\_id\_list

Message Destination: McAfee ESM Message Destination

Functions: mcafee\_esm\_edit\_case, mcafee\_esm\_get\_case\_detail, mcafee\_esm\_get\_case\_events\_detail, mcafee\_esm\_get\_list\_of\_cases, mcafee\_esm\_get\_triggered\_alarms, mcafee\_esm\_query

Workflows: McAfee ESM Close Case, Mcafee ESM Get Case Details, McAfee ESM Get Case Events Detail, McAfee ESM Get Case List, McAfee ESM Get Triggered Alarms, McAfee ESM Query

Rules: Close McAfee ESM Case, McAfee ESM Get Case Details, McAfee ESM Get Case Events Detail, McAfee ESM Get Case List, McAfee ESM Get Triggered Alarms, Run McAfee ESM Query

Run the integration framework

Run the integration manually with the following command:

resilient-circuits run

The resilient-circuits command starts, loads its components, and continues to run until interrupted. If it stops immediately with an error message, check your configuration values and retry. Below shows a successful connection to the Resilient platform and loading of components

2018-04-10 12:05:09,686 INFO [app] Resilient server: 9.108.163.130

2018-04-10 12:05:09,687 INFO [app] Resilient org: TestOrg

2018-04-10 12:05:09,687 INFO [app] Logging Level: INFO

2018-04-10 12:05:09,688 WARNING [co3] Unverified HTTPS requests (cafile=false).

2018-04-10 12:05:10,142 INFO [app] Components auto-load directory: (none)

2018-04-10 12:05:10,306 INFO [component\_loader] Loading 7 components

2018-04-10 12:05:10,307 INFO [component\_loader] 'fn\_mcafee\_epo.components.mcafee\_esm\_get\_case\_detail.FunctionComponent' loading

2018-04-10 12:05:10,307 INFO [component\_loader] 'fn\_mcafee\_epo.components.mcafee\_esm\_get\_list\_of\_cases.FunctionComponent' loading

2018-04-10 12:05:10,307 INFO [component\_loader] 'fn\_mcafee\_epo.components.mcafee\_esm\_get\_case\_events\_detail.FunctionComponent' loading

2018-04-10 12:05:10,307 INFO [component\_loader] 'fn\_mcafee\_epo.components.mcafee\_esm\_edit\_case.FunctionComponent' loading

2018-04-10 12:05:10,307 INFO [component\_loader] 'fn\_mcafee\_epo.components.mcafee\_esm\_get\_get\_triggered\_alarms.FunctionComponent' loading

2018-04-10 12:05:10,307 INFO [component\_loader] 'fn\_mcafee\_epo.components.mcafee\_esm\_query.FunctionComponent' loading

2018-04-10 12:05:10,307 INFO [component\_loader] 'fn\_mcafee\_epo.components.mcafee\_esm\_case\_polling.ESM\_CasePolling loading

2018-04-10 12:05:10,308 WARNING [actions\_component] Unverified STOMP TLS certificate (cafile=false)

2018-04-10 12:05:10,309 INFO [stomp\_component] Connect to 9.108.163.130:65001

2018-04-10 12:05:10,310 INFO [actions\_component] 'fn\_mcafee\_esm.components.mcafee\_esm\_get\_case\_detail.FunctionComponent' function 'mcafee\_esm\_get\_case\_detail' registered to 'mcafee\_esm\_message\_destination'

2018-04-10 12:05:10,310 INFO [actions\_component] 'fn\_mcafee\_esm.components.mcafee\_esm\_get\_list\_of\_cases.FunctionComponent' function 'mcafee\_\_esm\_get\_list\_of\_cases' registered to 'mcafee\_esm\_message\_destination'

2018-04-10 12:05:10,310 INFO [actions\_component] 'fn\_mcafee\_esm.components.mcafee\_esm\_get\_case\_events\_detail.FunctionComponent' function 'mcafee\_esm\_get\_case\_evenets\_detail' registered to 'mcafee\_esm\_message\_destination'

2018-04-10 12:05:10,310 INFO [actions\_component] 'fn\_mcafee\_esm.components.mcafee\_esm\_edit\_case.FunctionComponent' function 'mcafee\_esm\_edit\_case' registered to 'mcafee\_esm\_message\_destination'

2018-04-10 12:05:10,310 INFO [actions\_component] 'fn\_mcafee\_esm.components.mcafee\_esm\_get\_triggered\_alarms.FunctionComponent' function 'mcafee\_esm\_get\_triggered\_alarms' registered to 'mcafee\_esm\_message\_destination'

2018-04-10 12:05:10,310 INFO [actions\_component] 'fn\_mcafee\_esm.components.mcafee\_esm\_query.FunctionComponent' function 'mcafee\_esm\_query' registered to 'mcafee\_esm\_message\_destination'

2018-04-10 12:05:10,310 INFO [app] Components loaded

2018-04-10 12:05:10,312 INFO [app] App Started

2018-04-10 12:05:10,414 INFO [actions\_component] STOMP attempting to connect

2018-04-10 12:05:10,414 INFO [stomp\_component] Connect to Stomp...

2018-04-10 12:05:10,437 INFO [client] Connection established

2018-04-10 12:05:10,537 INFO [client] Connected to stomp broker [session=ID:resilient.localdomain-40775-1523276401752-5:3, version=1.2]

2018-04-10 12:05:10,538 INFO [stomp\_component] Connected to failover:(ssl://9.108.163.130:65001)?maxReconnectAttempts=1,startupMaxReconnectAttempts=1

2018-04-10 12:05:10,538 INFO [stomp\_component] Client HB: 0 Server HB: 15000

2018-04-10 12:05:10,538 INFO [stomp\_component] No Client heartbeats will be sent

2018-04-10 12:05:10,538 INFO [stomp\_component] Requested heartbeats from server.

2018-04-10 12:05:10,539 INFO [actions\_component] STOMP connected.

[mcafee\_esm\_case\_polling] Polling for cases in ESM is occurring

2018-04-10 12:05:10,641 INFO [actions\_component] Subscribe to message destination 'mcafee\_esm\_message\_destination'

2018-04-10 12:05:10,642 INFO [stomp\_component] Subscribe to message destination actions.<orgID>.mcafee\_esm\_message\_destination

Configuration of resilient-circuits for restartability

For normal operation, resilient-circuits must run continuously. The recommend way to do this is to configure it to automatically run at startup. On a Red Hat appliance, this is done using a systemd unit file such as the one below. You may need to change the paths to your working directory and app.config.

The unit file should be named ‘resilient\_circuits.service’:

sudo vi /etc/systemd/system/resilient\_circuits.service

The contents:

[Unit]  
Description=Resilient-Circuits Service  
After=resilient.service  
Requires=resilient.service

[Service]  
Type=simple  
User=integration  
WorkingDirectory=/home/integration  
ExecStart=/usr/local/bin/resilient-circuits run  
Restart=always  
TimeoutSec=10  
Environment=APP\_CONFIG\_FILE=/home/integration/.resilient/app.config  
Environment=APP\_LOCK\_FILE=/home/integration/.resilient/resilient\_circuits.lock

[Install]  
WantedBy=multi-user.target

Ensure that the service unit file is correctly permissioned:

sudo chmod 664 /etc/systemd/system/resilient\_circuits.service

Use the systemctl command to manually start, stop, restart and return status on the service:

sudo systemctl resilient\_circuits [start|stop|restart|status]

Log files for systemd and the resilient-circuits service can be viewed through the journalctl command:

sudo journalctl -u resilient\_circuits --since "2 hours ago"

Case Polling Description

When loaded and set to poll, the case polling integration spawns a new thread to handle all the polling and creation of incidents. For this to happen, within the app.config file esm\_polling must be set to True and incident\_template must be set to the location of a jinja template used to create incident data. The esm\_polling\_interval can be changed to represent how long (in seconds) to wait in-between polling intervals.

When the component is loaded, if esm\_polling is set to True a new thread is created. This thread reaches out to the ESM server and using the caseGetCaseList endpoint returns back a list of cases which are open and assigned to the logged-in ESM user. From here this list is cross referenced with active incidents within Resilient, if the case already exists as an incident in Resilient it moves on to the next case, otherwise a new incident is created in the Resilient platform based on the case data from the caseGetCaseDetail endpoint. To ensure the connection between cases and incidents, when the incident is created the custom field McAfee ESM Case ID is set to the ID of the case in ESM.

The incident template file can be edited to meet custom needs. The suggested way of accomplishing this is coping the default template that comes with the integration to a new directory and editing it from there, this template can be found at <python\_env>/lib/<python\_version>/site-packages/fn\_mcafee\_esm/data/templates/. The template utilizes jinja, and the documentation can be found here: <http://jinja.pocoo.org/docs/2.10/>. Once the custom template is finished its location will need to be be set in the config file at incident\_template=<location\_of\_template>.

Function Descriptions

The McAfee ESM integration comes packaged with 6 different Functions. These can be configured in Workflows how the user determines needed. Each Function comes with a corresponding Workflow to show how an example how it can be used and are explained below.

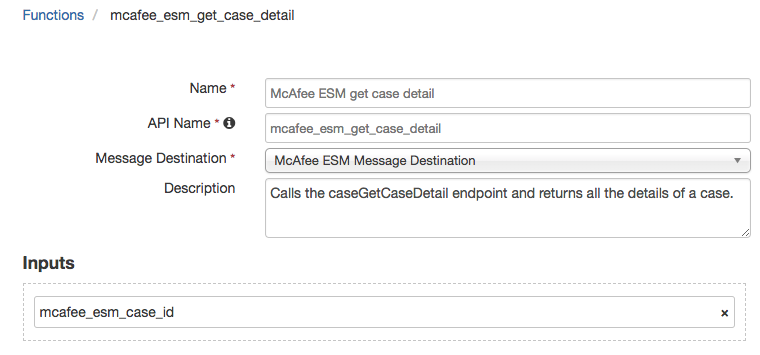
McAfee ESM Edit Case



The McAfee ESM Edit Case Function calls the caseEditCase ESM endpoint. This enables the Function to edit any incident. The Function accepts two inputs, mcafee\_esm\_case\_id, which is the numeric ID of the case in ESM, and mcafee\_esm\_edit\_case\_json which is a string of the JSON that will be used to edit the case. This is a text with value input type and comes with a few example input JSON strings that can be used. The case ID is need since the Function first does a call to the caseGetCaseDetail endpoint in ESM and then combines the results from that call with the string set as the mcafee\_esm\_edit\_case\_json input. Any value that is set in the edit case JSON input will override the results from the caseGetCaseDetail can when combined.

The default Workflow for this Function is an Incident Workflow which closes the case in ESM when the case is closed in Resilient.

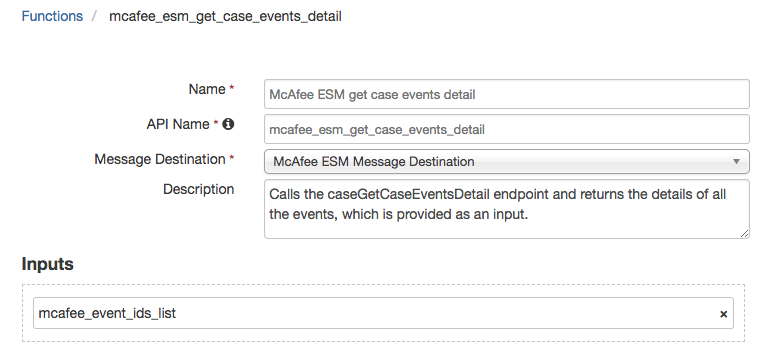
McAfee ESM Get Case Detail



The McAfee ESM Get Case Detail Function calls the caseGetCaseDetail endpoint in ESM and returns back all the information on that specific case. The function, takes one input, mcafee\_esm\_case\_id, this is the numeric ID of the case represented in ESM.

The default Workflow is an Incident Workflow that adds a note to the Incident containing additional details about the case in ESM.

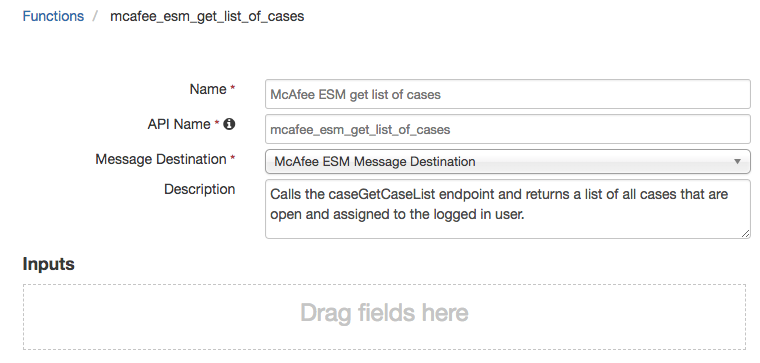
McAfee ESM Get Case Events Detail



The McAfee ESM Get Case Events Detail Function calls the caseGetCaseEventsDetail ESM endpoint and returns back all information of each of the events. This function takes one input, mcafee\_event\_eds\_list, this is a string and represents the comma separated list of events ID’s to be passed to the API.

The default Workflow for this function is a Data Table Workflow which when triggered on a row from the McAfee ESM Event List Data Table with an event ID will add additional details about the event to the Data Table row.

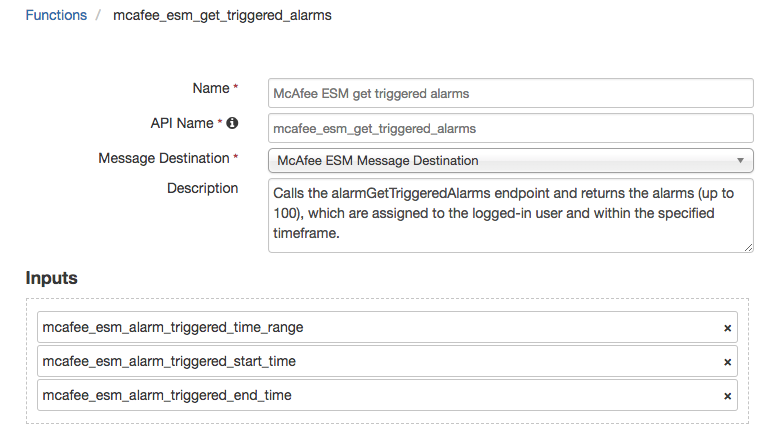
McAfee ESM Get List of Cases



The McAfee ESM Get List of Cases Function calls the caseGetCaseList endpoint and returns the list of all open cases in ESM which are also assigned to the logged-in ESM user. This Function does not take any inputs.

The default Workflow for this function is an Incident Workflow which adds a Note to an Incident stating how many open cases assigned to the logged-in user there is in ESM in addition to populating McAfee ESM Event List data table with event ID’s.

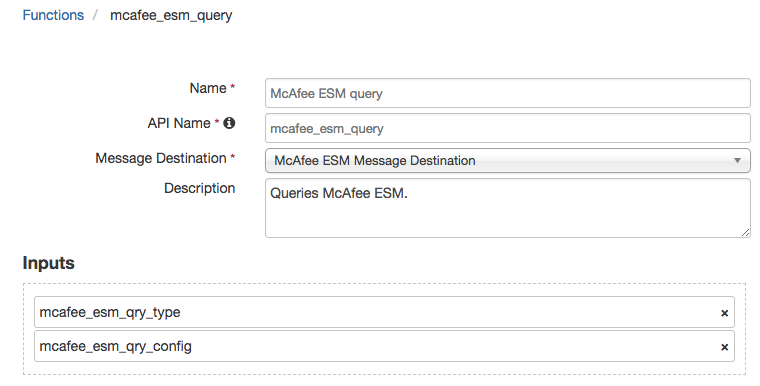
McAfee ESM Get Triggered Alarms



The McAfee ESM Get Triggered Alarms Function uses the alarmGetTriggeredAlarms ESM endpoint to return back up to the last 100 alarms assigned to the logged-in ESM user within the designated time frame. This Function accepts three different inputs, the first mcafee\_esm\_alarm\_triggered\_time\_range, is a select input which allows for quick and easy decision making when setting the specified time range of when to return alarms back. The other two, mcafee\_esm\_alarm\_triggered\_start\_time and two, mcafee\_esm\_alarm\_triggered\_end\_time when set, override the previous input and set the time range for alarms to be custom between the designated start and end date/times.

The default Workflow for this Function is an Incident Workflow that when triggered returns back alarms assigned to the logged-in ESM user within the last 30 days and populates the McAfee ESM Triggered Alarms data table.

McAfee ESM Query



The McAfee ESM Query Function queries ESM based on the inputs and returns the results back. This function takes two inputs, the first mcafee\_esm\_qry\_type is a select input which is used to specify the type of query it is. The second, mcafee\_esm\_qry\_config, is a string input which represents the JSON of the query config.

The default Workflow for this Function is an Incident Workflow which queries ESM for a specific event ID within the last 30 days and returns the number of occurrences as a note to the Incident.

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 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 will create progress information. Failures will 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.