IBM Resilient

resilient

Incident Response Platform Integrations

Outbound Email Workflow Functions V1.0.6

Release Date: February 2020

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 Outbound Email workflow functions.

Overview

The Outbound Email workflow function provides a way of sending email from the Resilient platform.

The Outbound Email integration package provides the following functionality:

- Send a plain text or HTML-formatted email by triggering a Resilient action
- Add incident data to the email body as well as incident attachments to the outgoing email

This document describes the function, workflow, and rule included in the package.

Installation

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

- Resilient platform is version 34 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 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 are 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.

Complete the following steps to install the Python components:

1. Ensure that the environment is up-to-date, as follows:

   ```
   sudo pip install --upgrade pip
   sudo pip install --upgrade setuptools
   sudo pip install --upgrade resilient-circuits
   ```

2. Run the following command to install the package:

```
    sudo pip install fn_outbound_email-1.0.6.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.

Complete the following steps to configure and run the integration:

1. Using sudo, switch to the integration user, as follows:

   ```
   sudo su - integration
   ```

2. Use one of the following commands to create or update the resilient-circuits configuration file. Use `–c` for new environments or `–u` for existing environments.

   ```
   resilient-circuits config -c
   ```

   or

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

3. Edit the resilient-circuits configuration file, as follows:

   a. In the [resilient] section, ensure that you provide all the information required to connect to the Resilient platform.

   b. In the [fn_outbound_email] section, edit the settings are required:

   ```
   [fn_outbound_email]

   # SMTP SERVER (IP ADDRESS OR FQDN)

   smtp_server=xxx.xxx.xxx.xxx

   smtp_user=xxx

   smtp_password=xxx

   # SMTP PORT NUMBER: 25 or 587

   smtp_port=25

   # SMTP CONNECTION TIMEOUT IN SECONDS

   smtp_conn_timeout=20

   # SMTP SSL MODE = (starttls, ssl, None)

   smtp_ssl_mode=None
   ```

```
# SSL Cert
# If your email server uses a self-signed SSL/TLS certificate, or some
# other certificate that is not automatically trusted by your machine,
# specify the file below, e.g. 'path/to/certificate.pem' OR
# set to true if using system cert store OR
# set to false if disabling SSL verification
#smtp_ssl_cafile=~/path/to/email_cert.cer
```

Deploy customizations to the Resilient platform

The package contains function definitions that you can use in workflows.

1. Use the following command to deploy these customizations to the Resilient platform:

   ```
   resilient-circuits customize
   ```

2. Respond to the prompts to deploy functions.

Run the integration framework

To test the integration package before running it in a production environment, you must 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.

Configure Resilient Circuits for restart

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 RHEL system, this is done using a systemd unit files to define services. The configuration file defines the following properties:

- OS user account to use.
- Directory from where it should run.
- Any required environment variables.
- Command to run the integrations, such as resilient-circuits run.
- Dependencies.

You may need to change the paths to your working directory and app.config.

1. The unit file must be named resilient_circuits.service. To create the file, enter the following command:

   ```
   sudo vi /etc/systemd/system/resilient_circuits.service
   ```

2. Add the following contents to the file and change as necessary. If you are not running on the Resilient appliance, then the “After” and “Requires” lines in the [Unit] section should be removed:

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

```
[Service]
Type=simple
User=integration
WorkingDirectory=/usr/share/integration
ExecStart=/usr/local/bin/resilient-circuits run
Restart=always
TimeoutSec=10
Environment=APP_CONFIG_FILE=/usr/share/integration/.resilient/app.config
Environment=APP_LOCK_FILE=/usr/share/integration/.resilient/resilient_circuits.lock

[Install]
WantedBy=multi-user.target
```

3. Ensure that the service unit file is correctly permissioned, as follows:

   ```
   sudo chmod 664 /etc/systemd/system/resilient_circuits.service
   ```

4. Reload and enable the new service:

   ```
   sudo systemctl daemon-reload

   sudo systemctl enable resilient_circuits.service
   ```

You can use the systemctl command to manually start, stop, restart and return status on the service:

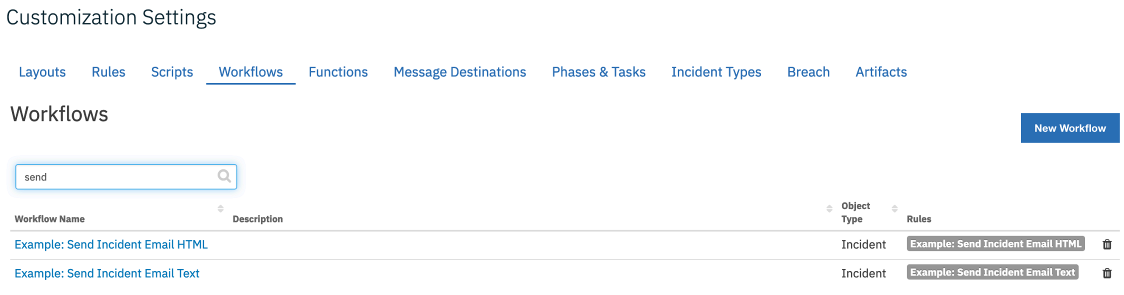
```
sudo systemctl [start|stop|restart|status] resilient_circuits
```

You can view log files for systemd and the resilient-circuits service using the journalctl command, as follows:

```
sudo journalctl -u resilient_circuits --since "2 hours ago"
```

Workflow Description

Once the function package deploys the workflows, you can view them in the Resilient platform Workflow tab, as shown below. The rules will also be deployed and may be viewed in the Rules tab.



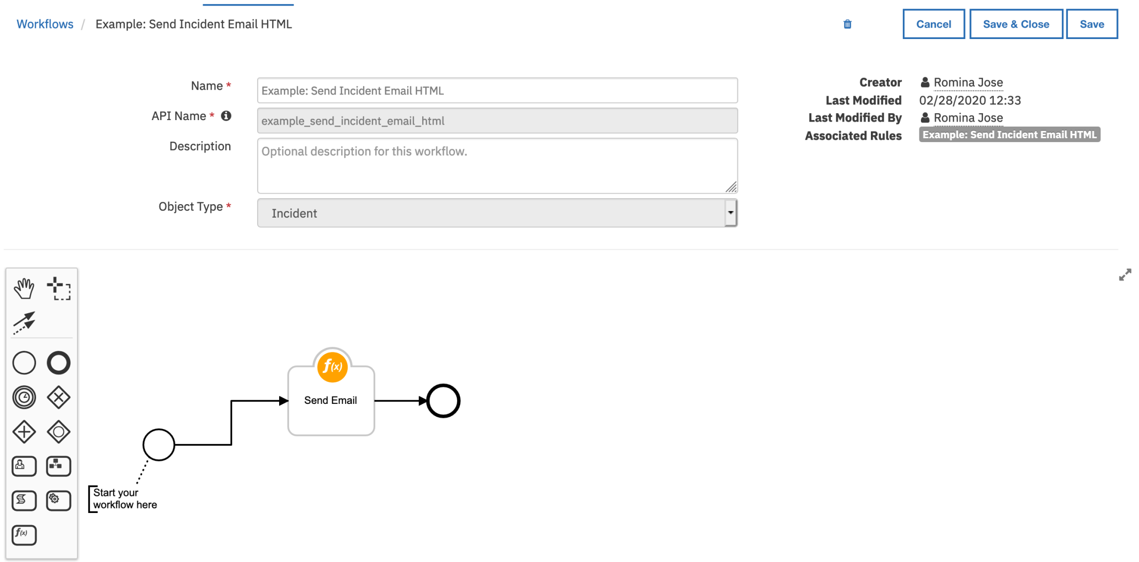
Workflows

send

Workflow Name	Description	Object Type	Rules
Example: Send Incident Email HTML		Incident	Example: Send Incident Email HTML
Example: Send Incident Email Text		Incident	Example: Send Incident Email Text

The workflows themselves will be as shown below.

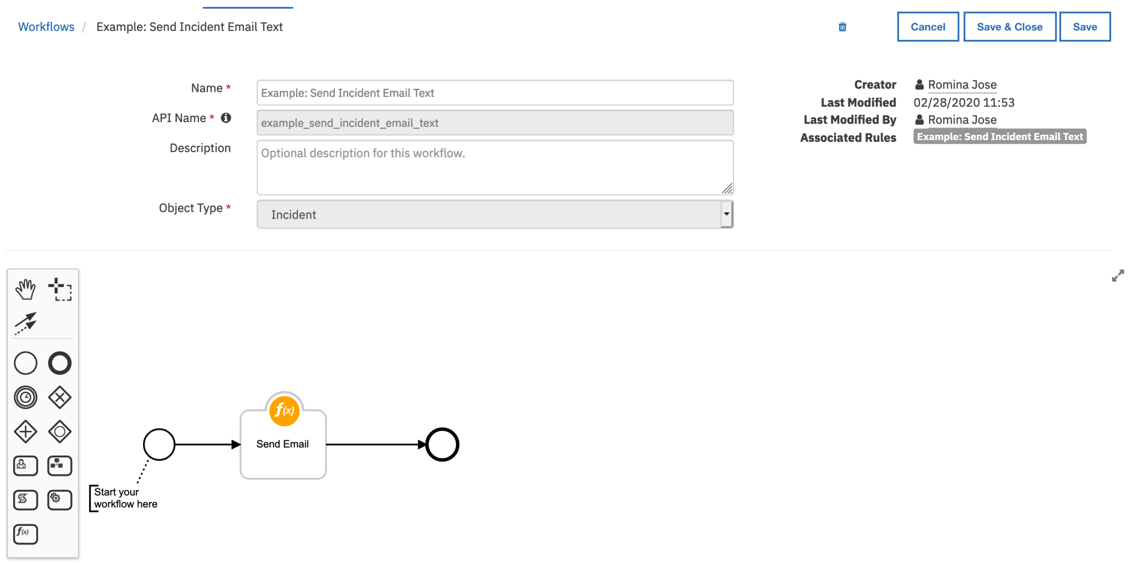
Example: Send Incident Email HTML



Name *	Example: Send Incident Email HTML
API Name *	example_send_incident_email_html
Description	Optional description for this workflow.
Object Type *	Incident

Creator	Romina Jose
Last Modified	02/28/2020 12:33
Last Modified By	Romina Jose
Associated Rules	Example: Send Incident Email HTML

Example: Send Incident Email Text



Name * Example: Send Incident Email Text

API Name * example_send_incident_email_text

Description Optional description for this workflow.

Object Type * Incident

Creator Romina Jose

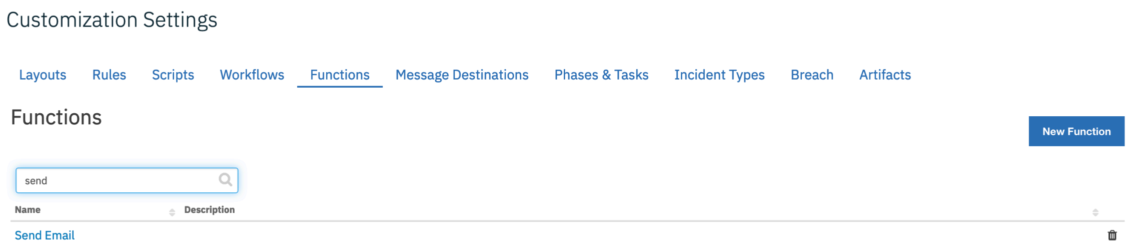
Last Modified 02/28/2020 11:53

Last Modified By Romina Jose

Associated Rules Example: Send Incident Email Text

Function Descriptions

Once the function package deploys the function, you can view it in the Resilient platform Functions tab, as shown below.



Customization Settings

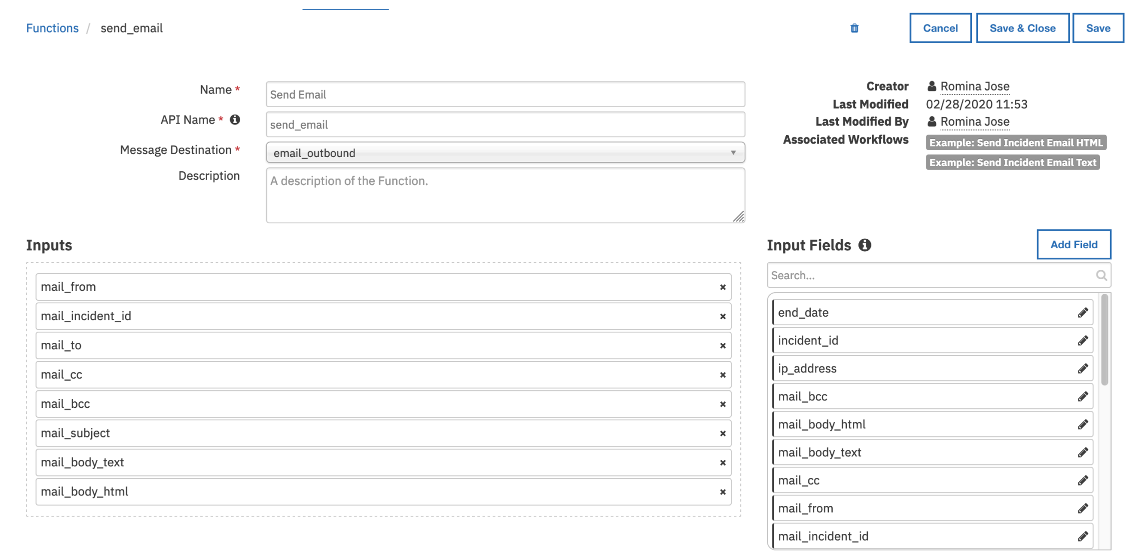
Functions

send

Name	Description
Send Email	

The function itself will be as shown below.

Send Email



Functions / send_email

Name * Send Email

API Name * send_email

Message Destination * email_outbound

Description A description of the Function.

Creator Romina Jose

Last Modified 02/28/2020 11:53

Last Modified By Romina Jose

Associated Workflows Example: Send Incident Email HTML
Example: Send Incident Email Text

Inputs

- mail_from
- mail_incident_id
- mail_to
- mail_cc
- mail_bcc
- mail_subject
- mail_body_text
- mail_body_html

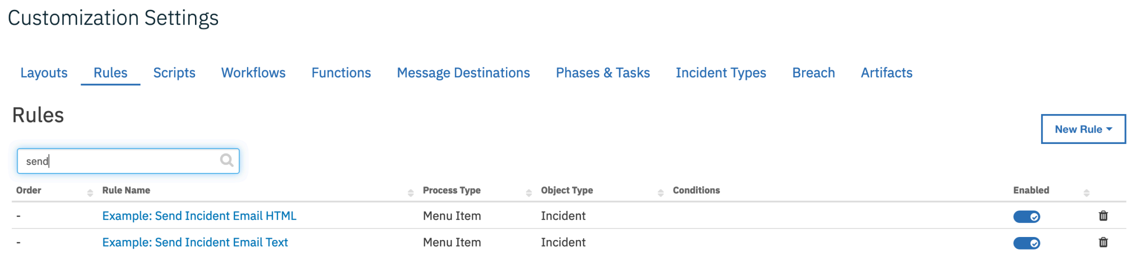
Input Fields

Search...

- end_date
- incident_id
- ip_address
- mail_bcc
- mail_body_html
- mail_body_text
- mail_cc
- mail_from
- mail_incident_id

Rules

Once the function package deploys the rules, you can view them in the Resilient platform Rules tab, as shown below.



Customization Settings

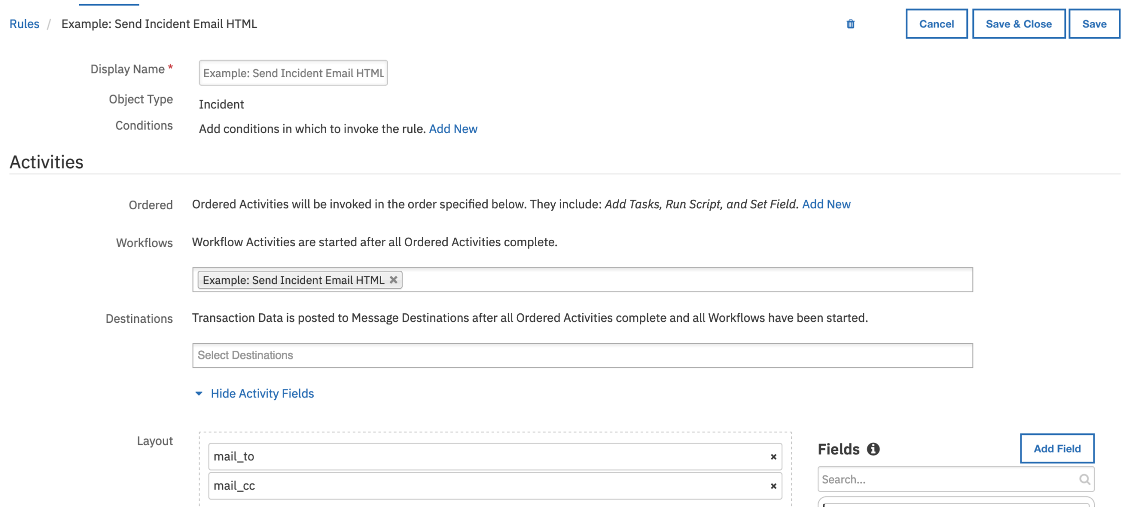
Rules

send

Order	Rule Name	Process Type	Object Type	Conditions	Enabled	
-	Example: Send Incident Email HTML	Menu Item	Incident		●	🗑
-	Example: Send Incident Email Text	Menu Item	Incident		●	🗑

The rules themselves will be as shown below.

Example: Send Incident Email HTML



Rules / Example: Send Incident Email HTML

Display Name * Example: Send Incident Email HTML

Object Type Incident

Conditions Add conditions in which to invoke the rule. Add New

Activities

Ordered Ordered Activities will be invoked in the order specified below. They include: *Add Tasks, Run Script, and Set Field.* Add New

Workflows Workflow Activities are started after all Ordered Activities complete.

Example: Send Incident Email HTML ×

Destinations Transaction Data is posted to Message Destinations after all Ordered Activities complete and all Workflows have been started.

Select Destinations

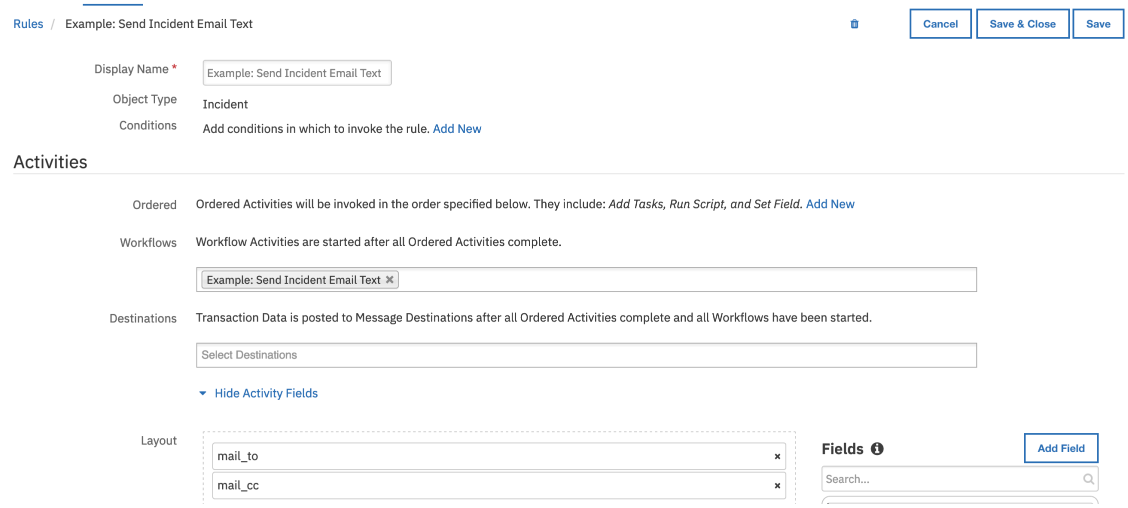
▾ Hide Activity Fields

Layout

mail_to ×

mail_cc ×

Example: Send Incident Email Text



Rules / Example: Send Incident Email Text

Display Name * Example: Send Incident Email Text

Object Type Incident

Conditions Add conditions in which to invoke the rule. Add New

Activities

Ordered Ordered Activities will be invoked in the order specified below. They include: *Add Tasks, Run Script, and Set Field.* Add New

Workflows Workflow Activities are started after all Ordered Activities complete.

Example: Send Incident Email Text ✕

Destinations Transaction Data is posted to Message Destinations after all Ordered Activities complete and all Workflows have been started.

Select Destinations

Layout

mail_to ✕

mail_cc ✕

Fields

Search...

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 https://ibm.com/mysupport.

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