User Guide: fn_scheduler_v1.0.0

Table of Contents

- Key Features
- Layout
- Function Create a Scheduled Rule
- Function List Scheduled Rules
- Function Pause a Scheduled Rule
- Function Resume a Scheduled Rule
- Function Remove a Scheduled Rule
- Rules

Key Features

This package of functions allows an enterprise to schedule a rule to run in the future associated with a incident, task, artifact, and datatable. Schedule times to run can be specified in the following ways:

- 1. cron (ex. * 0 * * * for every night at midnight)
- 2. interval (ex. 5h for every 5 hours, 2d for every 2 days. Valid values are s(econd), m(inute), h(our), d(ay), w(eek), M(onth))
- 3. date (ex. 2019/10/23 12:00:00 or 2019-10-23 12:00:00)
- 4. delta (ex. 1h for one hour in the future, the same values as interval are supported)

Scheduled rules using **cron** and **interval** are reocurring whereas **date** and **delta** are single event schedules. Scheduled rules are persisted so that restarts of resilient-circuits will resume already scheduled rules.

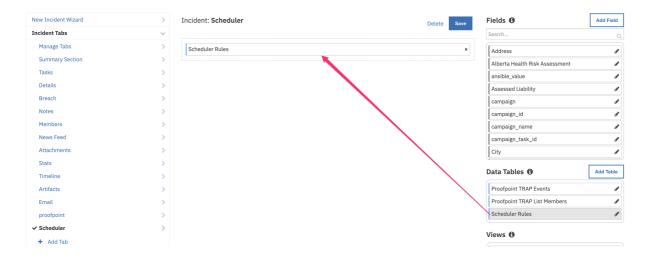
Functions available include:

- · Create a scheduled rule
- List scheduled rule
- · Remove a scheduled rule

Layout

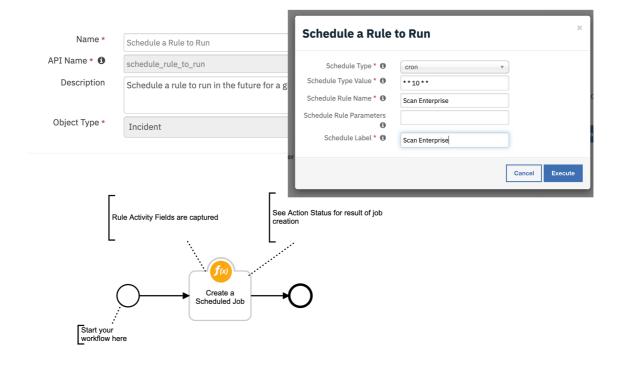
A datatable is used to display scheduled rules and to take actions such as pause, resume and remove a rule. This datatable can be added to your incident layout by adding a new tab and by dragging the Scheduler Rules datatable to the new tab. Remember to save the layout change.

2019-10-08



Function - Create a Scheduled Rule

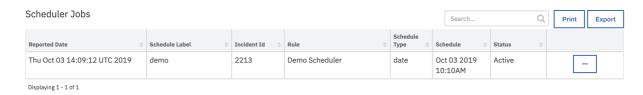
Schedule a rule to run on a schedule. This rule will be executed for a given incident, artifact, task, etc.



- ► Inputs:
- ► Outputs:
- ► Example Pre-Process Script:
- ► Example Post-Process Script:

Function - List Scheduled Rules

List the schedules presently defined

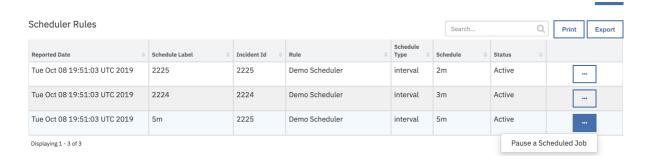


2019-10-08

- ▶ Inputs:
- ▶ Outputs:
- ► Example Pre-Process Script:
- ► Example Post-Process Script:

Function - Pause a Scheduled Rule

Pause an existing scheduled rule



- ► Inputs:
- ► Outputs:
- ► Example Pre-Process Script:
- ► Example Post-Process Script:

Function - Resume a Scheduled Rule

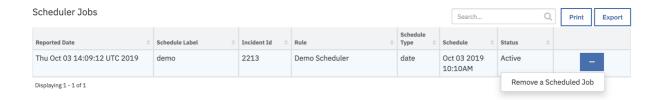
Resume an existing scheduled rule



- ► Inputs:
- ► Outputs:
- ► Example Pre-Process Script:
- ► Example Post-Process Script:

Function - Remove a Scheduled Rule

Stop a schedule



- ► Inputs:
- ► Outputs:
- ► Example Pre-Process Script:
- ► Example Post-Process Script:

Rules

Rule Name	Object	Workflow Triggered
List Scheduled Rules	incident	list_schedules
Remove a Scheduled Rule	scheduler_rules	remove_a_schedule
Schedule a Rule to Run	incident	schedule_rule_to_run
Schedule a Rule to Run - Artifact	artifact	schedule_a_rule_to_run_artifact
Schedule a Rule to Run - Task	task	schedule_a_rule_to_runtask

Considerations

Rules

- Rules must be enabled to be scheduled and are again checked when the scheduled rule is triggered.
- Rules scheduled must match the invoking Rule. For instance, to create a scheduled artifact rule, use the rule Create a Schedule Artifact.
- All schedules must be in the future.
- Disabled rules will not execute but the scheduled rule will continue to trigger.
- Rules triggered on closed incidents will not run and the scheduled rule will be removed.
- Incident notes are created each time a scheduled rule is excuted documenting the rule invocation.
- Scheduled rules will not show up under Action Status and Workflow Status. Refer instead
 to the incident notes.

Datatables

- Datatable scheduled rules are not part of this package, but can be easily created for a specific Datatable.
- Datatable scheduled rules cannot currently reference the invoking datatable row in the pre-processing script. However, a rule's activity field can be defined to prompt for it.

Persistence of Scheduled Rules

- Labels for scheduled rules need to be unique. Attempting to create a duplicate scheduled rule label will fail.
- Sqlite is used to persist scheduled rules. Restarting resilient-circuits will continue with the scheduled rules already defined.

2019-10-08

Integrations

• A function executed from a scheduled rule is free to perform any operation against Resilient. Even through a scheduled rule runs from a specific Incident, Resilient API calls can collect and operate on other incidents. For example, a scheduled rule can call a function which queries Resilient for all open tasks with due dates to review any overdue.

2019-10-08