

# fn\_relations

---

## Table of Contents

- [Release Notes](#)
  - [Overview](#)
    - [Key Features](#)
  - [Requirements](#)
    - [SOAR platform](#)
    - [Cloud Pak for Security](#)
    - [Python Environment](#)
  - [Installation](#)
    - [Install](#)
    - [App Configuration](#)
    - [Custom Layouts](#)
  - [Function - Relations: Assign Parent](#)
  - [Function - Relations: Auto Close Child Incidents](#)
  - [Function - Relations: Remove Child Relation](#)
  - [Function - Relations: Sync Child Table Data](#)
  - [Function - Relations: Sync Notes](#)
  - [Data Table - Relations Child Incidents](#)
  - [Custom Fields](#)
  - [Custom Artifact Types](#)
  - [Rules](#)
  - [Troubleshooting & Support](#)
- 

## Release Notes

Version	Date	Notes
1.0.0	07/2020	Initial Release
1.0.1	05/2021	Add AppHost Support Patch: Verification of Parent Incident before creating relation
1.0.2	04/2023	Support for Python 3.9 Support for CP4S Patch: Verification of Parent and Child Incidents are different

---

## Overview

The Relations function is meant to provide the ability to allow parent/child relations levels within Resilient to link incidents “manually”. This function allows users sync notes from child to parent and vice versa, close all child incidents under a parent, and update the parent when the status of the child changes.

This document outlines the functionality of function as it relates to Resilient.

## Builds Relationships of Incidents within IBM Security SOAR



This package consists of 5 Functions, 5 Workflows, and 6 Rules along with 2 new fields and 1 new data table.

### 1. Assign Parent:

Used to assign children incidents to a parent by supplying the incident ID of the parent. The function configures both parent incident and child incident appropriately.

### 2. Remove Child Relation:

Used to remove child relationship with parent by removing all related content created in the relationship.

### 3. Sync Notes:

This is used to sync notes between the parent and child incidents with 2 different rules.

### 4. Sync Child Table Data:

Used to update child information within the table if the incident name or status changes.

### 5. Auto Close Child Incidents:

Allows the closing of all child incidents listed in the parent data table.

## Key Features

- Assign Parent incident to Child
- Remove Child Relation if established incorrectly
- Sync Notes between Parent and Child
  - Child Notes sync automatically
  - Parent Notes sync manually
- Parent incident contains Data Table of Children incidents
- Sync Child incident data automatically to Parent Data Table
- Auto-close Child incidents on Parent incident closure

---

## Requirements

No application specific configuration settings are required.

This app supports the IBM Security QRadar SOAR Platform and the IBM Security QRadar SOAR for IBM Cloud Pak for Security.

### SOAR platform

The SOAR platform supports two app deployment mechanisms, App Host and integration server.

If deploying to a SOAR platform with an App Host, the requirements are:

- SOAR platform  $\geq$  45.0.7899.
- The app is in a container-based format (available from the AppExchange as a zip file).

If deploying to a SOAR platform with an integration server, the requirements are:

- SOAR platform  $\geq$  45.0.7899.
- The app is in the older integration format (available from the AppExchange as a zip file which contains a tar.gz file).
- Integration server is running resilient-circuits $\geq$ 45.0.0.
- If using an API key account, make sure the account provides the following minimum permissions:

Name	Permissions
Org Data	Read
Function	Read
All Incidents	Read
All Incidents Status	Edit
All Incidents Fields	Edit
All Incidents Notes	Edit

The following SOAR platform guides provide additional information:

- *App Host Deployment Guide*: provides installation, configuration, and troubleshooting information, including proxy server settings.
- *Integration Server Guide*: provides installation, configuration, and troubleshooting information, including proxy server settings.
- *System Administrator Guide*: provides the procedure to install, configure and deploy apps.

The above guides are available on the IBM Documentation website at [ibm.biz/soar-docs](https://ibm.biz/soar-docs). On this web page, select your SOAR platform version. On the follow-on page, you can find the *App Host Deployment Guide* or *Integration Server Guide* by expanding **Apps** in the Table of Contents pane. The System Administrator Guide is available by expanding **System Administrator**.

## Cloud Pak for Security

If you are deploying to IBM Cloud Pak for Security, the requirements are:

- IBM Cloud Pak for Security  $\geq$  1.4.
- Cloud Pak is configured with an App Host.
- The app is in a container-based format (available from the AppExchange as a zip file).

The following Cloud Pak guides provide additional information:

- *App Host Deployment Guide*: provides installation, configuration, and troubleshooting information, including proxy server settings. From the Table of Contents, select Case Management and Orchestration & Automation > **Orchestration and Automation Apps**.

- *System Administrator Guide*: provides information to install, configure, and deploy apps. From the IBM Cloud Pak for Security IBM Documentation table of contents, select Case Management and Orchestration & Automation > **System administrator**.

These guides are available on the IBM Documentation website at [ibm.biz/cp4s-docs](https://ibm.biz/cp4s-docs). From this web page, select your IBM Cloud Pak for Security version. From the version-specific IBM Documentation page, select Case Management and Orchestration & Automation.

## Python Environment

Both Python 2.7 and Python 3.6 are supported. Additional package dependencies may exist for each of these packages:

- resilient-circuits>=45.0.0
- resilient-lib>=45.0.0

---

## Installation

### Install

- To install or uninstall an App or Integration on the *SOAR platform*, see the documentation at [ibm.biz/soar-docs](https://ibm.biz/soar-docs).
- To install or uninstall an App on *IBM Cloud Pak for Security*, see the documentation at [ibm.biz/cp4s-docs](https://ibm.biz/cp4s-docs) and follow the instructions above to navigate to Orchestration and Automation.

### App Configuration

No application specific configuration settings are required.

### Custom Layouts

- Import the Data Tables and Custom Fields like the screenshot below:

#### **If Relation Level is: Parent**

New Tab: Child Incidents

- Add Relations Child Incidents data table

Summary Section:

- Add Relation Level

Tasks

Details

Child Incidents

QRadar

SEP

Areas/Users Involved

Artifacts

Notes

Attachments

News Feed

Timeline

Incident ID

Incident Name

Incident Status

5146

Child Incident 1

Closed

5147

Child Incident 2

Closed

5166

Child Incident 4

Closed

Search...

Print

Export

Relation Level

Status

Phase

Severity

Incident Disposition

Data Compromised

Resolution

Incident Classification

Incident Type

FQUAL Association

NIST Attack Vectors

Parent

Closed

Initial

High

Confirmed

Unknown

Resolved

Security

Reported Phish

—

E-mail

Edit

People

If Relation Level is: Child

Summary Section:

- Add Relation Level
- Parent ID

Summary

ID

5146

Relation Level

Child

Parent ID

5169

Status

Closed

Phase

Initial

Severity

Info

Function - Relations: Assign Parent

Create a parent/child relationship between the 2 incidents provided.

Name \*

API Name \* ⓘ

Message Destination \*

Description

Relations: Assign Parent

relations\_assign\_parent

fn\_relations

Create a parent/child relationship between the 2 incidents provided.

Inputs

relations\_child\_incident\_id

relations\_parent\_incident\_id

► Inputs:

Name	Type	Required	Example	Tooltip
relations_child_incident_id	number	Yes	-	-
relations_parent_incident_id	number	Yes	-	-

► Outputs:

**NOTE:** This example might be in JSON format, but `results` is a Python Dictionary on the SOAR platform.

```
results = {
    'child_artifact_results': {"success": True, "content": {'description': 'Parent Incident ID in Relationship',
                                                            'type': 'related_parent_incident',
                                                            'value': 2345}},
    'table_addition_results': {"success": True, "content": {'cells':
{'relations_incident_id': {'value': '<div class="rte"><div><a href="#incidents/1234" target="_blank">1234</a></div></div>',
                            'relations_incident_name': {'value': "Child Incident Name"},
                            'relations_incident_status': {'value': 'A'}
                        }}}},
    'parent_artifact_results': {"success": True, "content": {'description': 'Parent Incident ID in Relationship',
                                                            'type': 'related_parent_incident',
                                                            'value': 2345}},
    'notes_synced': 5
}
```

► Example Pre-Process Script:

```
inputs.relations_child_incident_id = incident.id
inputs.relations_parent_incident_id = rule.properties.relations_parent_incident
```

► Example Post-Process Script:

```
None
```

## Function - Relations: Auto Close Child Incidents

Close child incidents when the parent incident is closed.

Name \*

API Name \* ⓘ

Message Destination \*

Description

Relations: Auto Close Child Incidents

relations\_auto\_close\_child\_incidents

fn\_relations

Close child incidents when the parent incident is closed.

Inputs

relations\_parent\_incident\_id

► Inputs:

Name	Type	Required	Example	Tooltip
relations_parent_incident_id	number	Yes	-	-

► Outputs:

**NOTE:** This example might be in JSON format, but `results` is a Python Dictionary on the SOAR platform.

```
results = {
    "success": True
}
```

► Example Pre-Process Script:

```
inputs.relations_parent_incident_id = incident.id
```

► Example Post-Process Script:

```
None
```

## Function - Relations: Remove Child Relation

Used to remove the relation child relation from a Child incident as well as removing the parent relation from the Parent incident if it no longer has children.

Name \*

Relations: Remove Child Relation

API Name \* ⓘ

relations\_remove\_child\_relation

Message Destination \*

fn\_relations

Description

Used to remove the relation child relation from a Child incident as well as removing the parent relation from the Parent incident if it no longer has children.

Inputs

relations\_child\_incident\_id

relations\_remove\_notes

► Inputs:

Name	Type	Required	Example	Tooltip
relations_child_incident_id	number	Yes	-	-
relations_remove_notes	boolean	Yes	-	-

► Outputs:

**NOTE:** This example might be in JSON format, but `results` is a Python Dictionary on the SOAR platform.

```
results = {
    "success": True
}
```

► Example Pre-Process Script:

```
inputs.relations_child_incident_id = incident.id
inputs.relations_remove_notes = rule.properties.relations_remove_notes
```

► Example Post-Process Script:

```
None
```

## Function - Relations: Sync Child Table Data

Update data within the Parent Table if the Child data changes.



Name \*

Relations: Sync Child Table Data

API Name \* ⓘ

relations\_sync\_child\_table\_data

Message Destination \*

fn\_relations

Description

Update data within the Parent Table if the Child data changes.

Inputs

relations\_child\_incident\_id

relations\_parent\_incident\_id

► Inputs:

Name	Type	Required	Example	Tooltip
relations_child_incident_id	number	Yes	-	-
relations_parent_incident_id	number	Yes	-	-

► Outputs:

**NOTE:** This example might be in JSON format, but `results` is a Python Dictionary on the SOAR platform.

```
results = {
    "Success": True,
    "response": {'cells': {'relations_incident_id': {'value': '<div class="rte">
<div><a href="#incidents/1234" target="_blank">1234</a></div></div>'},
                      'relations_incident_name': {'value': "Child
Incident Name"},
                      'relations_incident_status': {'value': 'A'}}
}
```

► Example Pre-Process Script:

```
import re

regex = re.compile(r'#incidents/(\d+)')

inputs.relations_parent_incident_id =
int(re.findall(regex,incident.properties.relations_parent_id['content'])[0])
inputs.relations_child_incident_id = incident.id
```

► Example Post-Process Script:

None

## Function - Relations: Sync Notes

Sync notes from the incident where the note is currently to the parent or child.

Name \*

API Name \* ⓘ

Message Destination \*

Description

Relations: Sync Notes

relations\_sync\_notes

fn\_relations

Sync notes from the incident where the note is currently to the parent or child.

Inputs

incident\_id

relations\_note\_id

► Inputs:

Name	Type	Required	Example	Tooltip
incident_id	number	Yes	-	-
relations_note_id	number	Yes	-	-

► Outputs:

**NOTE:** This example might be in JSON format, but `results` is a Python Dictionary on the SOAR platform.

```
results = {
    "success": True
}
```

► Example Pre-Process Script:

```
inputs.relations_note_id = note.id
inputs.incident_id = incident.id
```

► Example Post-Process Script:

None

Data Table - Relations Child Incidents

Relations Child Incidents

Search...

Print

Export

Incident ID	Incident Name	Incident Status	
5146	Child Incident 1	Closed	
5147	Child Incident 2	Closed	
5166	Child Incident 4	Closed	

Displaying 1 - 3 of 3

API Name:

dt\_relations\_child\_incidents

Columns:

Column Name	API Access Name	Type	Tooltip
Incident ID	relations_incident_id	textarea	-
Incident Name	relations_incident_name	text	-
Incident Status	relations_incident_status	select	-

Custom Fields

Label	API Access Name	Type	Prefix	Placeholder	Tooltip
Relation Level	relations_level	select	properties	-	Is this incident considered a Parent or Child incident?
Parent ID	relations_parent_id	textarea	properties	-	Incident Number of the Parent Incident

Custom Artifact Types

Display Name	API Access Name	Description
--------------	-----------------	-------------

Display Name	API Access Name	Description
Related Parent Incident	<code>related_parent_incident</code>	Incident ID of the parent of all related incidents to create a relation within Resilient incidents manually.

## Rules

Rule Name	Object	Workflow Triggered
Example: Relations - Assign Parent Incident	incident	<code>example_relations_assign_parent</code>
Example: Relations - Close Child Incidents	incident	<code>example_relations_auto_close_child_incidents</code>
Example: Relations - Remove Child Relation	incident	<code>example_relations_remove_child_relation</code>
Example: Relations - Sync Notes with Child	note	<code>example_relations_sync_notes_to_parentchild</code>
Example: Relations - Sync Notes with Parent	note	<code>example_relations_sync_notes_to_parentchild</code>
Example: Relations - Update Child Incident Parent Data Table	incident	<code>example_relations_update_child_table_data</code>

## Troubleshooting & Support

Refer to the documentation listed in the Requirements section for troubleshooting information.

### For Support

This is a IBM Community provided App. Please search the Community [ibm.biz/soarcommunity](https://ibm.biz/soarcommunity) for assistance.