IBM Security

IBM

IBM Resilient SOAR Platform

RSA NetWitness Functions Guide

V1.1.0

Date: March 2020

Resilient SOAR Platform
RSA NetWitness Functions Guide

Version	Publication	Notes
1.1.0	March 2020	Removed the dependency on fn_utilities function Utilities: String to Attachment in the example_netwitness_retrieve_log_file. Also updated the example workflow to retrieve log data handles creating the attachment within the rsa_netwitness integration.
1.0	March 2019	Initial publication.

Table of Contents

Overview

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.

The RSA NetWitness functions query for metadata and return pcap and log files for specific times and sessions.

Installation

Before you install the IBM Resilient RSA NetWitness functions, make sure that your environment meets the following prerequisites:

- Your Resilient platform version is 30 or later. If supporting the Resilient for MSSPs multi-organization feature, Resilient platform V33 or later is required.
- A Resilient integration server running Resilient Circuits V30 or later. To setup an integration server, see https://ibm.biz/res-int-server-guide.
- A dedicated Resilient account to use as the API user. This can be any account that has the permission to create incidents, and view and modify administrator and customization settings. You need to know the account username and password.

  NOTE: Should you later change the dedicated Resilient account to another user, the new user must also have the permission to edit incidents, in addition to the permission to create incidents and view and modify administrator and customization settings. The edit permission is necessary so that the integration can continue to modify or synchronize the incidents escalated by the original user account.

  If supporting the Resilient for MSSP feature, the Resilient account must have permission to access the configuration, global dashboard and all child organizations.

Perform the following procedure to install the IBM Resilient RSA NetWitness package.

1. Download the IBM Resilient RSA NetWitness .zip file from the IBM Security App Exchange.
2. Copy the zip file to your Integration Server nd SSH into it.
3. Unzip the package:

   ```
   unzip fn_rsa_netwitness-x.x.x.zip
   ```

4. Change directory into the unzipped directory:

   ```
   cd fn_rsa_netwitness-x.x.x
   ```

5. Install the package:

   ```
   pip install fn_rsa_netwitness-x.x.x.tar.gz
   ```

6. Import the configurations into your file:

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

7. Import the fn_rsa_netwitness customizations into your Resilient platform:

   ```
   resilient-circuits customize -y -l fn_rsa_netwitness
   ```

8. Open the config file, scroll to the bottom and edit your [fn_rsa_netwitness] configurations:

   ```
   nw_packet_server_url=<http://test.nw_packet_server.com:50104>
   nw_packet_server_user=<nw_packet_server_username>
   nw_packet_server_password=<nw_packet_server_password>
   nw_packet_server_verify=[true|false]

   nw_log_server_url=<http://test.nw_log_server.com:50102>
   nw_log_server_user=<nw_log_server_username>
   nw_log_server_password=<nw_log_server_password>
   nw_log_server_verify=[true|false]
   ```

9. Save and close the app.config file.
10. Optionally, run selftest to test the integration you configured:

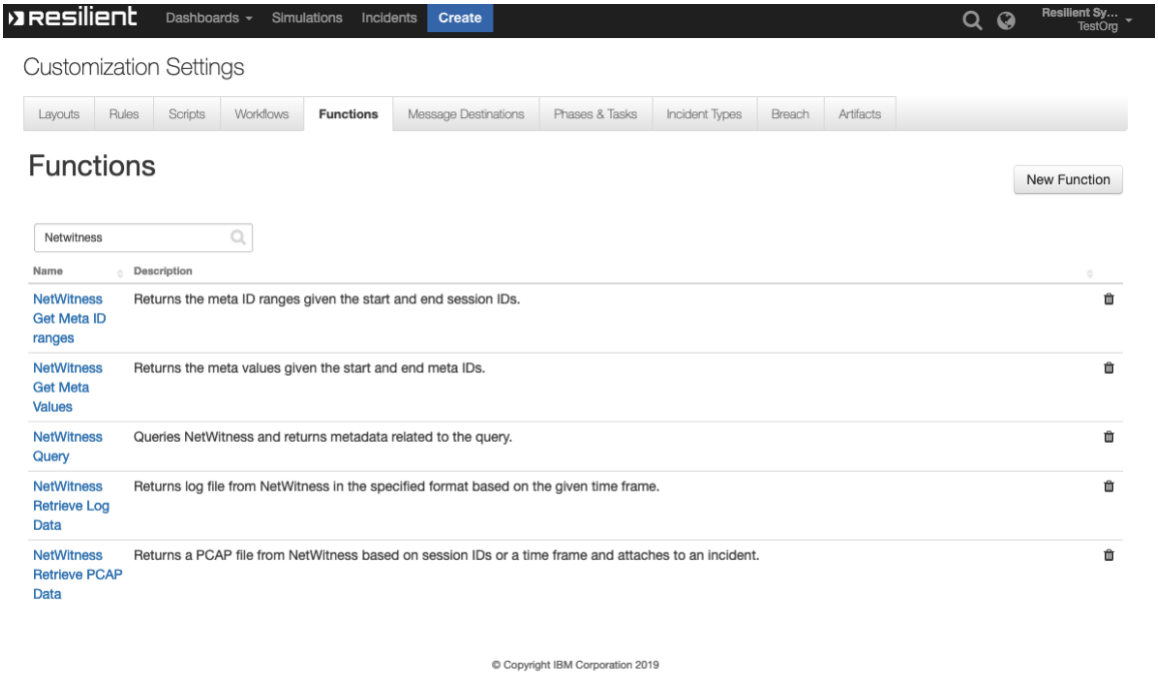
```
resilient-circuits selftest -l fn_rsa_netwitness
```

11. Run Resilient Circuits or restart the service on Linux or Windows.

```
resilient-circuits run
```

Function Descriptions

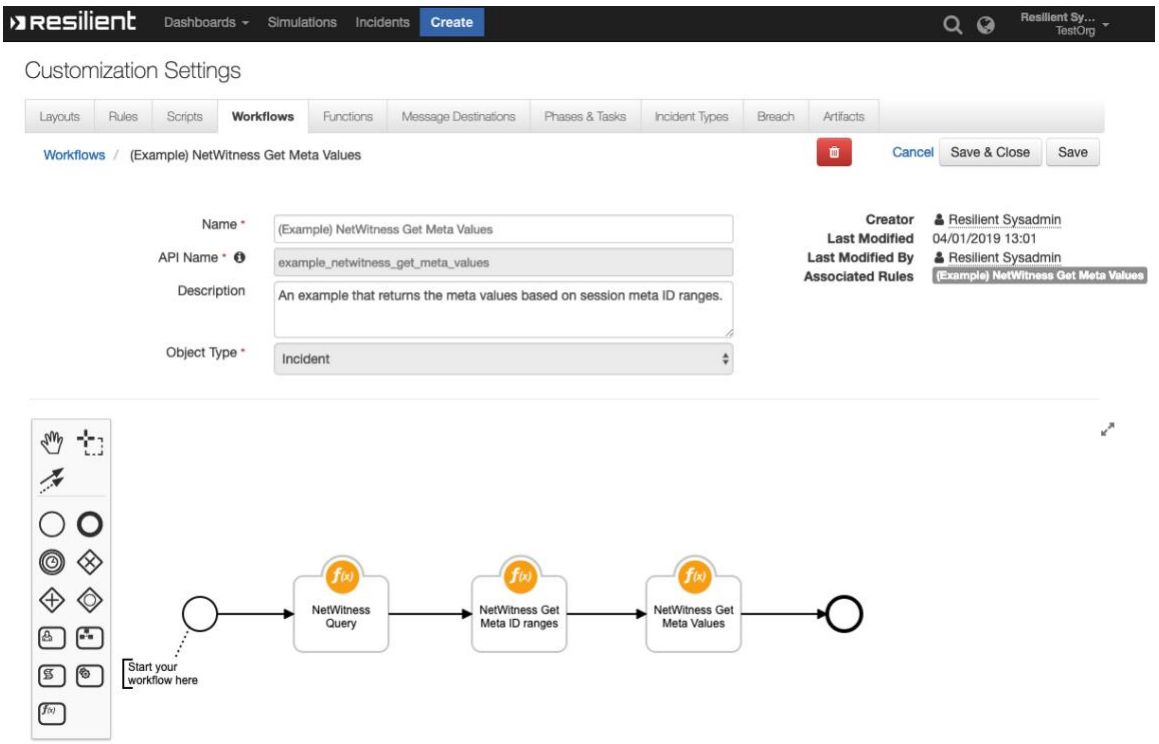
Once the function package deploys the functions, you can view them in the Resilient platform Functions tab, as shown below.



fn_rsa_netwitness: NetWitness Get Meta ID Ranges

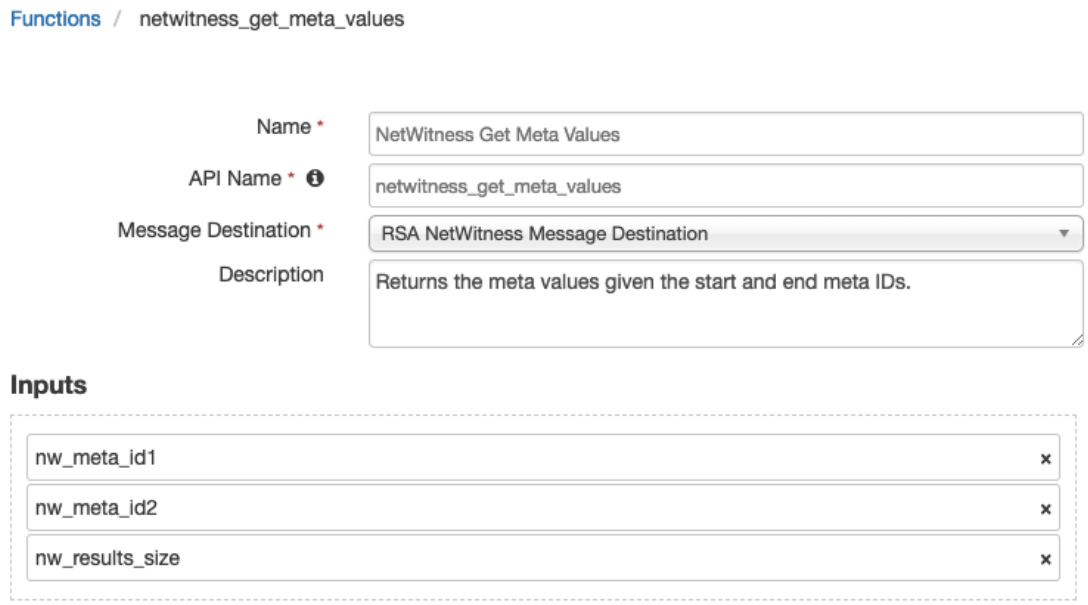
The NetWitness Get Meta ID Ranges function returns the first and last meta ID fields when given the session IDs. You can also specify the size of the results returned by setting nw_results_size.

This function works well when paired with the NetWitness Query and NetWitness Get Meta Values functions. The Query function provides the session ID range, this function uses that output to get the meta ID range, and the Get Meta Values function uses its output to get all the meta values. All this is accomplished by the example workflow, (Example) NetWitness Get Meta Values, shown below.



fn_rsa_netwitness: NetWitness Get Meta Values

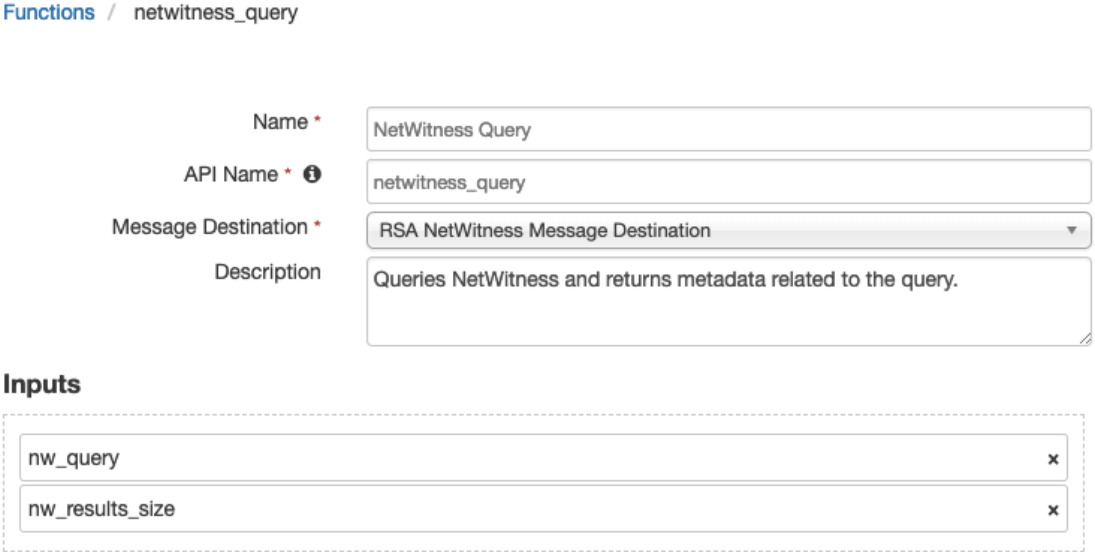
The NetWitness Get Meta Values function returns the meta values between the first and last meta ID fields. You can also specify the size of the results returned by setting nw_results_size.



This is included in the workflow, (Example) NetWitness Get Meta Values. See the NetWitness Get Meta ID Ranges for more information.

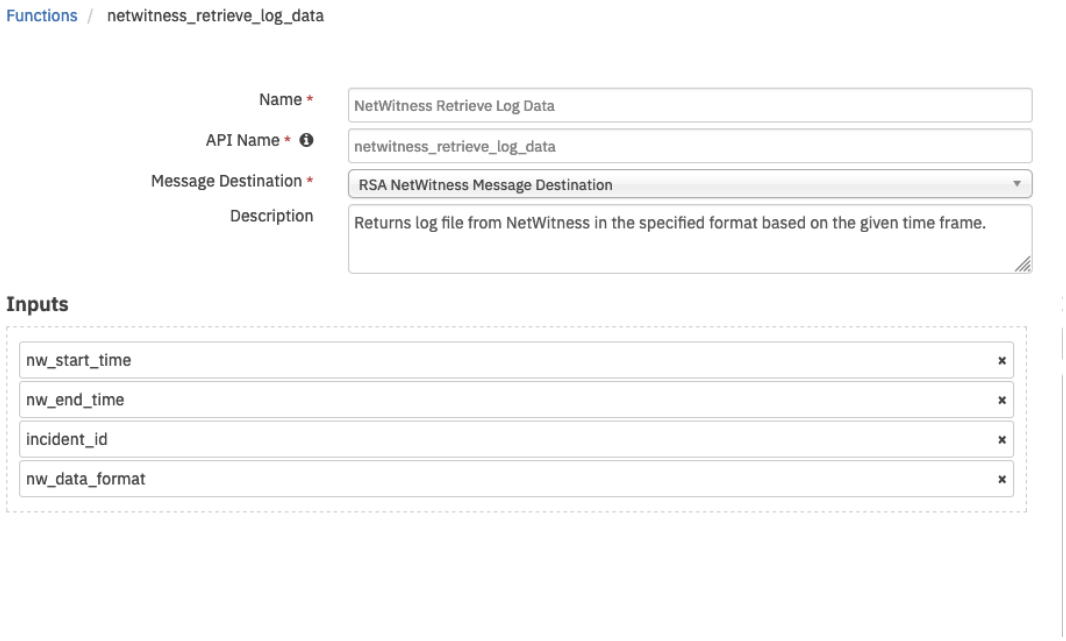
fn_rsa_netwitness: NetWitness Query

The NetWitness Query function takes a string query as an input and returns the query response as json. Setting the size of the results to be returned can also be set using nw_results_size. This function is used in the workflows (Example) NetWitness Get Meta Values and (Example) NetWitness Retrieve PCAP File.



fn_rsa_netwitness: NetWitness Retrieve Log Data

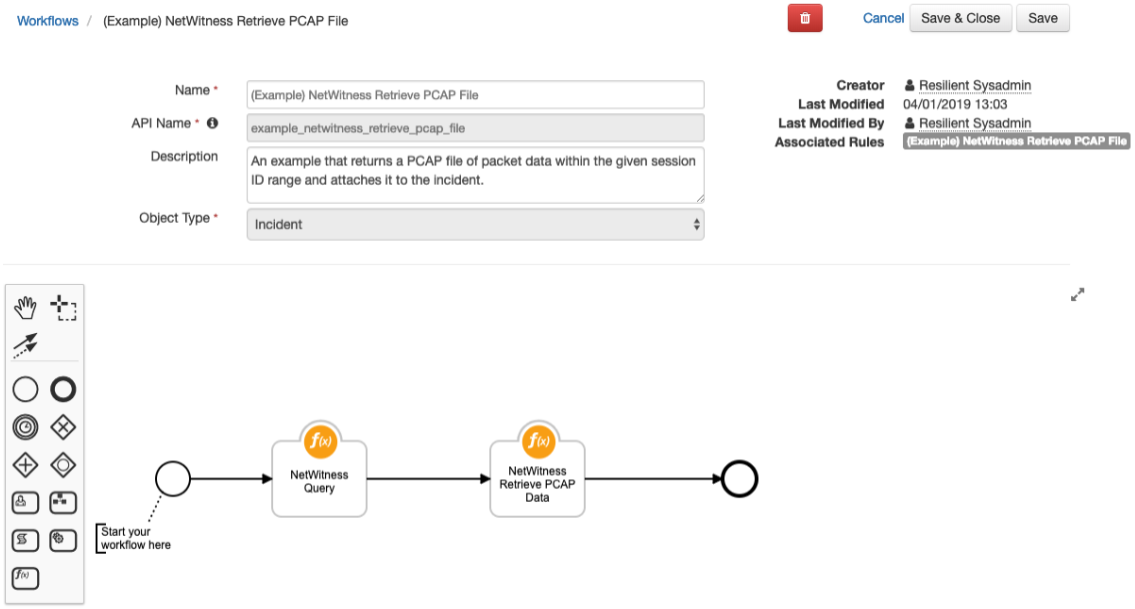
The NetWitness Retrieve Log Data function takes the incident id, and a start and end time and returns the log data in the specified format, which can be plain text, csv, xml, or json.



fn_rsa_netwitness: NetWitness Retrieve PCAP Data

The NetWitness Retrieve PCAP Data function returns a PCAP data file of the specific network data based on a given timeframe or comma separated list of session IDs.

This function automatically adds the PCAP data as an attachment to the incident. An example of this function being used in a workflow is shown below.



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 support, visit https://ibm.com/mysupport.

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