

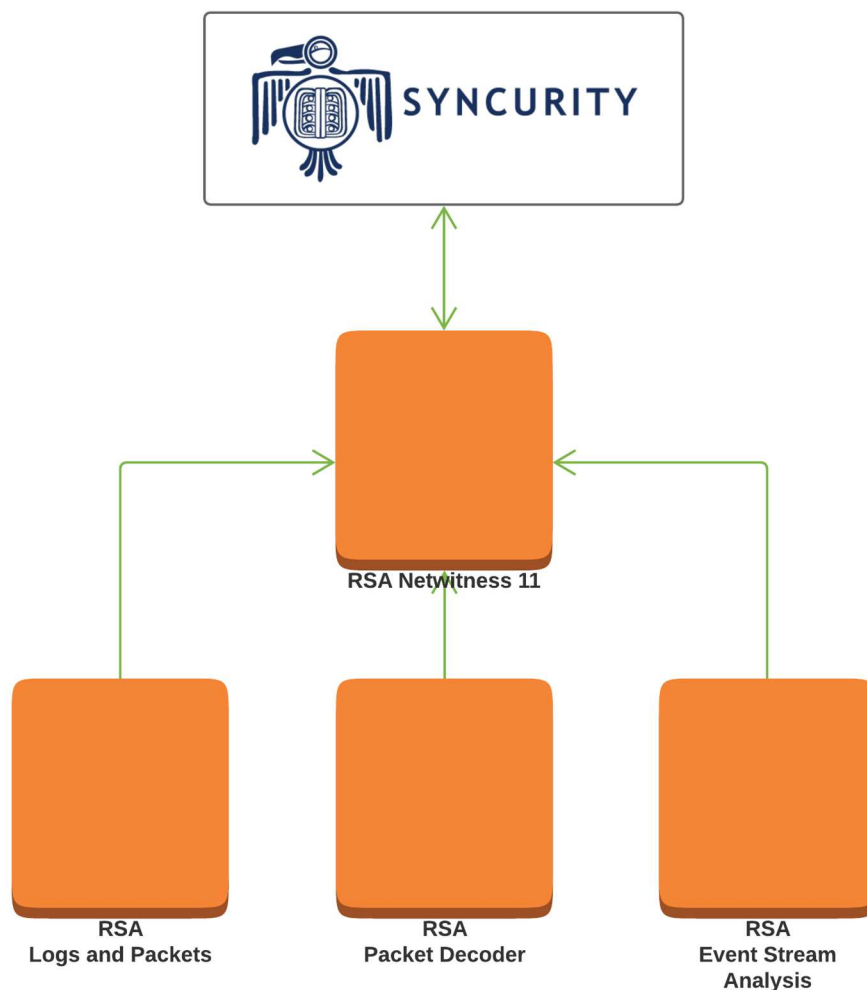
RSA® NETWITNESS®
Security Operations
Implementation Guide

Syncurity IR Flow 5.x

Daniel R. Pintal, RSA Partner Engineering
Last Modified: October 22, 2018

Solution Summary

Syncurity has integrated with RSA NetWitness to provide a single pane of glass to monitor, investigate and remediate security events within the enterprise. The integration utilizes RSA NetWitness API to gather incidents and alerts administrators to address critical security related events within the network infrastructure.



Partner Product Configuration

Before You Begin

This section provides instructions for configuring the Syncurity IR Flow with RSA NetWitness. This document is not intended to suggest optimum installations or configurations.

It is assumed that the reader has both working knowledge of all products involved, and the ability to perform the tasks outlined in this section. Administrators should have access to the product documentation for all products in order to install the required components.

All Syncurity IR Flow components must be installed and working prior to the integration. Perform the necessary tests to confirm that this is true before proceeding.

!> Important: The configuration shown in this Implementation Guide is for example and testing purposes only. It is not intended to be the optimal setup for the device. It is recommended that customers make sure Syncurity IR Flow is properly configured and secured before deploying to a production environment. For more information, please refer to the Syncurity IR Flow documentation or website.

Syncurity IR Flow Prerequisites

Prior to any configuration steps for this integration please check that your configuration meets the minimum requirements for operation:

1. RSA NetWitness instance is updated to version 11.1.0.1 or greater.
2. Syncurity IR-Flow instance is updated to version 5.1.0 or greater.
3. Any network configurations are complete for communication between RSA NetWitness and Syncurity IR-Flow. (e.g. Open Ports, Routes, etc.)
4. A user has been created in RSA NetWitness with proper privileges allowing Syncurity IR-Flow to ingest information.

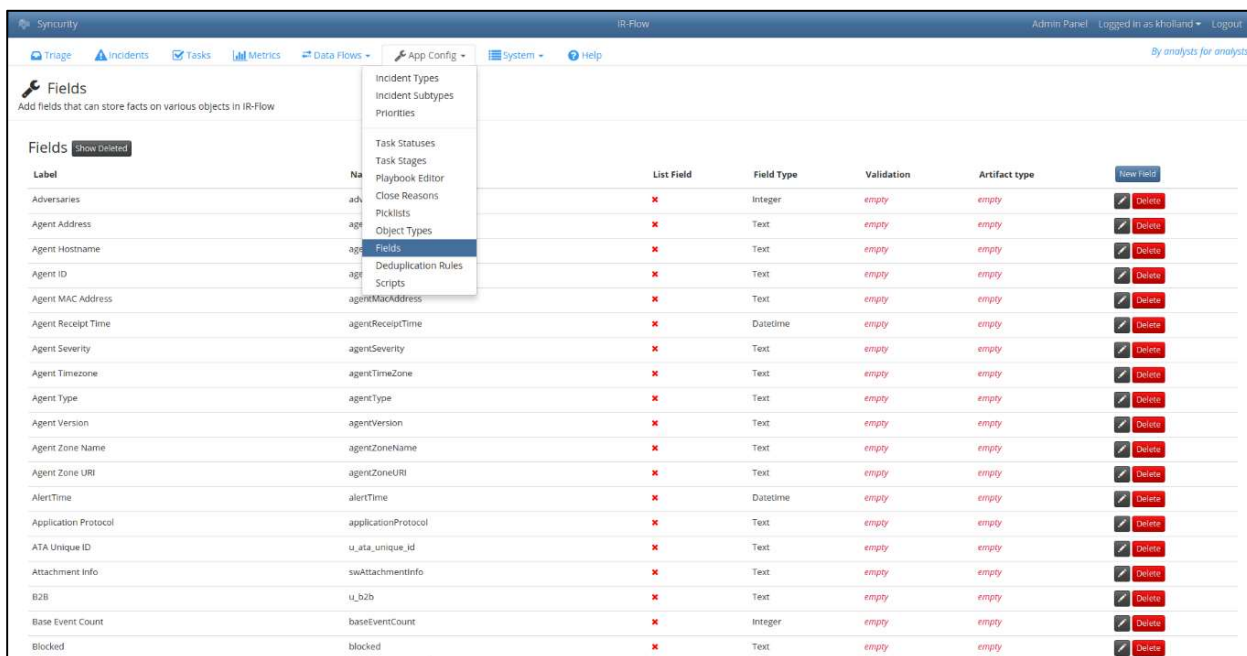
Syncurity IR Flow Configuration

Alert Object Setup

In order for this integration to submit data to IR-Flow, there must be a properly configured Alert object along with an Incoming Data Field Group. Before either of these can be configured, ensure that all required fields are present, and create any that are not. The table below contains the Label, Name, and Field Type for each field needed; none of the fields are of the list type, or require validation or an artifact type.

Label	Name	Field Type
Incident Alerts	alerts	JSON
Incident ID	incidentId	Text
Original Alert URL	originalAlertURL	URL
Results	results	JSON
Title	title	Text

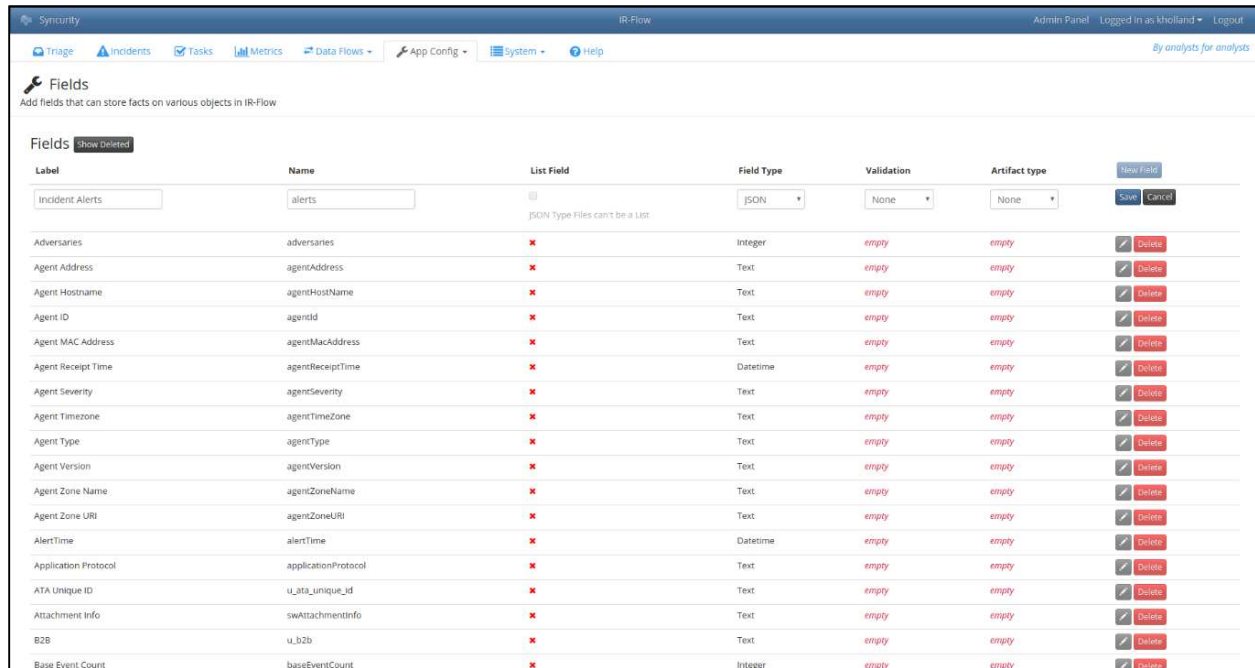
1. Locate the list of fields under **App Config > Fields** in the IR-Flow web interface. New fields can be created with the New Field button as needed.



The screenshot shows the 'Fields' configuration page in the Syncurity IR-Flow web interface. The page title is 'Fields' and it includes a sub-header 'Add fields that can store facts on various objects in IR-Flow'. A navigation menu on the left includes 'Fields', 'Incident Types', 'Incident subtypes', 'Priorities', 'Task Stages', 'Task Stages', 'Playbook Editor', 'Close Reasons', 'Picklists', 'Object Types', 'Deduplication Rules', and 'Scripts'. The main content area displays a table of fields with the following columns: Label, Name, List Field, Field Type, Validation, and Artifact type. A 'New Field' button is located in the top right corner of the table. The table contains the following data:

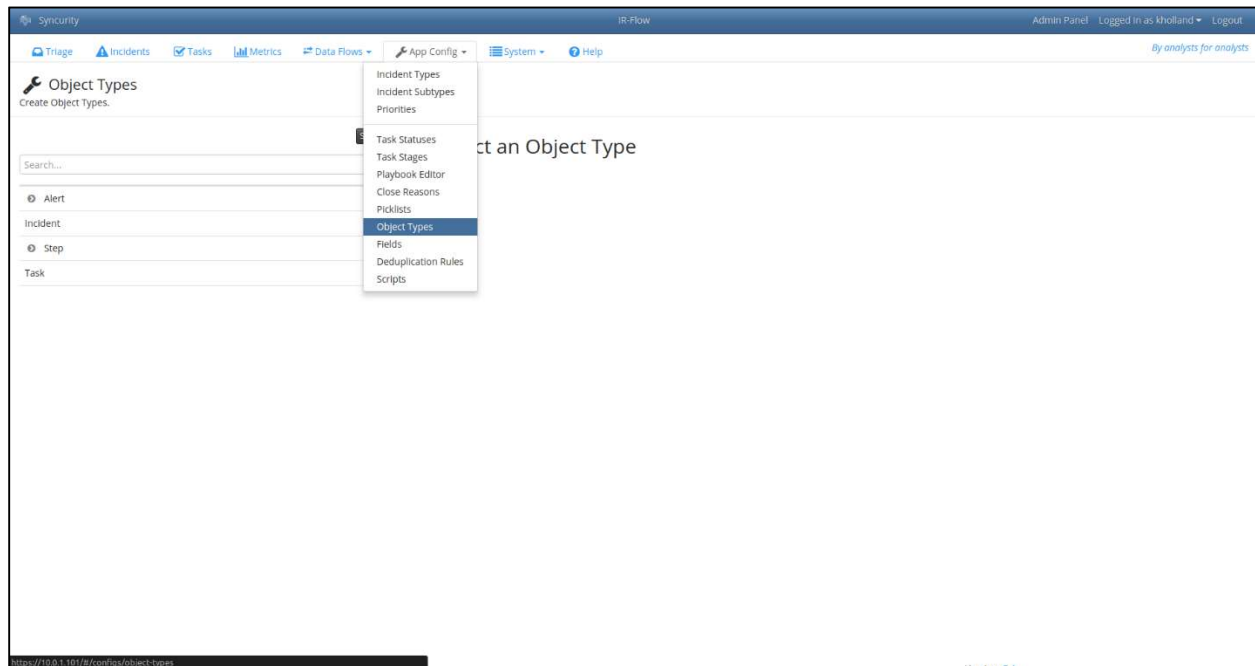
Label	Name	List Field	Field Type	Validation	Artifact type	
Adversaries	adv	x	Integer	empty	empty	Delete
Agent Address	agt	x	Text	empty	empty	Delete
Agent Hostname	agn	x	Text	empty	empty	Delete
Agent ID	agt	x	Text	empty	empty	Delete
Agent MAC Address	agentMacAddress	x	Text	empty	empty	Delete
Agent Receipt Time	agentReceiptTime	x	Datetime	empty	empty	Delete
Agent Severity	agentSeverity	x	Text	empty	empty	Delete
Agent Timezone	agentTimezone	x	Text	empty	empty	Delete
Agent Type	agentType	x	Text	empty	empty	Delete
Agent Version	agentVersion	x	Text	empty	empty	Delete
Agent Zone Name	agentZoneName	x	Text	empty	empty	Delete
Agent Zone URI	agentZoneURI	x	Text	empty	empty	Delete
Alert Time	alertTime	x	Datetime	empty	empty	Delete
Application Protocol	applicationProtocol	x	Text	empty	empty	Delete
ATA Unique ID	u_ata_unique_id	x	Text	empty	empty	Delete
Attachment Info	sauAttachmentInfo	x	Text	empty	empty	Delete
B2B	u_b2b	x	Text	empty	empty	Delete
Base Event Count	baseEventCount	x	Integer	empty	empty	Delete
Blocked	blocked	x	Text	empty	empty	Delete

2. Once everything is filled out for a new field to be created, simply click the **Save** button.



Label	Name	List Field	Field Type	Validation	Artifact type	
Incident Alerts	alerts	<input type="checkbox"/>	JSON	None	None	New Field Save Cancel
Adversaries	adversaries	<input checked="" type="checkbox"/>	Integer	empty	empty	Delete
Agent Address	agentAddress	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
Agent Hostname	agentHostName	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
Agent ID	agentId	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
Agent MAC Address	agentMacAddress	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
Agent Receipt Time	agentReceiptTime	<input checked="" type="checkbox"/>	Datetime	empty	empty	Delete
Agent Severity	agentSeverity	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
Agent Timezone	agentTimezone	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
Agent Type	agentType	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
Agent Version	agentVersion	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
Agent Zone Name	agentZoneName	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
Agent Zone URI	agentZoneUri	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
AlertTime	alertTime	<input checked="" type="checkbox"/>	Datetime	empty	empty	Delete
Application Protocol	applicationProtocol	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
ATA Unique ID	u_ata_unique_id	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
Attachment Info	swAttachmentInfo	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
B2B	u_b2b	<input checked="" type="checkbox"/>	Text	empty	empty	Delete
Base Event Count	baseEventCount	<input checked="" type="checkbox"/>	Integer	empty	empty	Delete

3. Configure the alert object in **App Config > Object Types**.



Object Types
Create Object Types.

Search...

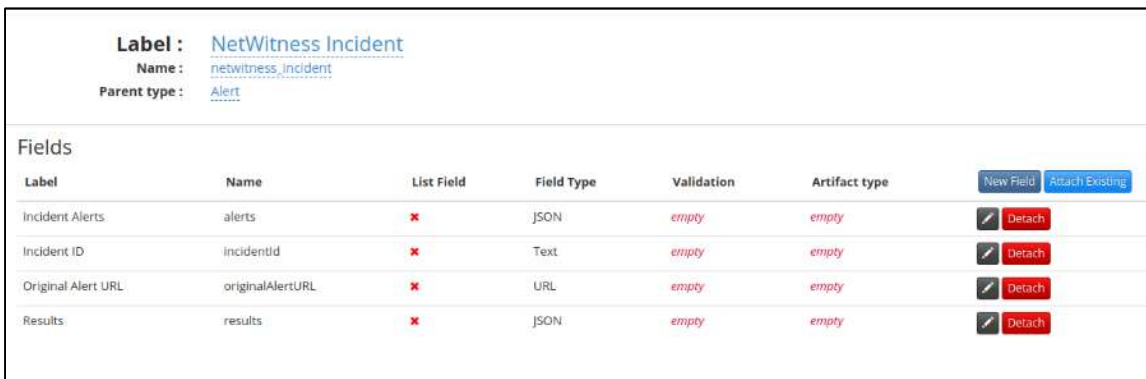
- Alert
- Incident
- Step
- Task

App Config menu items:
Incident Types
Incident Subtypes
Priorities
Task Statuses
Task Stages
Playbook Editor
Close Reasons
Picklists
Object Types
Fields
Deduplication Rules
Scripts

- Click the **+** button next to **Alert** to create a new **Alert** object.



- Complete the **Label field** with NetWitness Incident, and the **Name** field with NetWitness Incident.

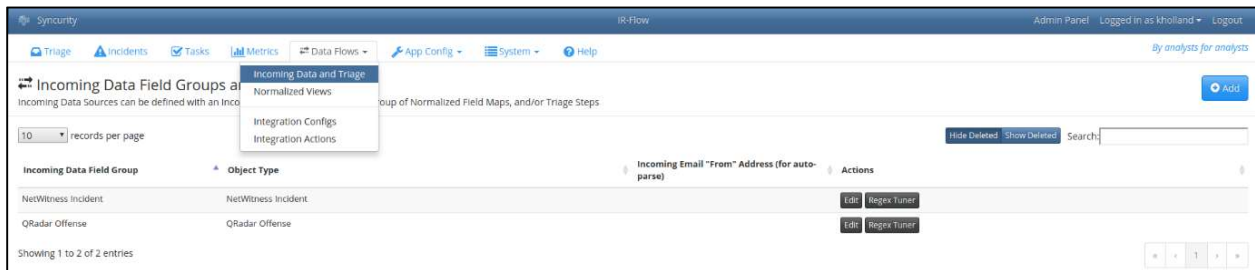


Label	Name	List Field	Field Type	Validation	Artifact type	
Incident Alerts	alerts	✗	JSON	empty	empty	✗ Detach
Incident ID	incidentid	✗	Text	empty	empty	✗ Detach
Original Alert URL	originalAlertURL	✗	URL	empty	empty	✗ Detach
Results	results	✗	JSON	empty	empty	✗ Detach

- Attach the appropriate fields using the **Attach Existing** button, and navigating to each of the required fields. The Save button will add the selected field to the Alert object.



- Navigate to **Data Flows > Incoming Data and Triage** to setup the Incoming Data Field Group for the integration.



- Use the **Add** button and fill out the **Data Source Name** with NetWitness Incident, and select the alert object we created previously for the **Alert Type** field. Click Save once these are filled out, and everything is set.

Create Data Source ✕

Data Source Name

Alert Type

Incoming Email "From" Address to Auto-Parse

Download PCAP and Import to IR-Flow

NetWitness, with the Packet Decoder (or simply Decoder) configured in your infrastructure allows NetWitness to capture network traffic in the form of a PCAP. This PCAP can be attached to enrich an alert in IR-Flow. This section will only cover the required arguments. For a complete list of options please refer to the IR-Flow Integrations Documentation under NetWitness.

1. Login to IR-Flow as the **irflow** user
2. Navigate to the **irflow_integrations/integrations/netwitness**
3. Run the **netwitness_download_pcap.py** file with the required arguments

```
python3 netwitness_download_pcap.py <arguments>
```

Required Arguments

-u / --plugin_user

RSA NetWitness API user's username

-p / --plugin_pass

RSA NetWitness API user's password

-H / --target_host

RSA NetWitness host instance address

-d / --decoder_host

Host address of the decoder from which to fetch the PCAP file

-o / --decoder_port

REST API port over which to communicate with the decoder

-s / --session_id

Session ID for which to pull the relevant PCAP file

-a / --alert_num

IR-Flow alert number to submit results to

Example Command

```
python3 netwitness_download_pcap.py -U rsa_user -p p@55w0rd -H 10.0.0.1 -d 10.0.0.2 -o 50004 -s 12345 -a 54321
```

1. Reload the alert in IR-Flow to see the PCAP attachment
2. Once you have confirmed this is working, you can configure this as an action in IR-Flow. See IR-Flow documentation for more info.

Service Setup

Setting up the integration service to pull incidents into IR-Flow is handled from within the IR-Flow instance itself, generally through an SSH connection. Once signed in as the **irflow** user:

1. Navigate into the `irflow-integrations/integrations/netwitness` RSA NetWitness integration directory
2. Run the `first-time-setup.sh` script, and ensure that the newly created `netwitness.conf` and `last.conf` files are readable and writable by all users – `chmod` may be used as needed.
3. Fill in the **netwitness.conf** file's **Host** field with the host address of your RSA NetWitness instance, and **username** and **password** fields with your username and password on said instance respectively. The **disable_ssl** field should default to **False**, but can be set to **True** to disable SSL verification with the RSA NetWitness instance.

Service Setup

Setting up the integration service to pull incidents into IR-Flow is handled from within the IR-Flow instance itself, generally through an SSH connection. Once signed in as the `irflow` user:

1. Navigate into the **irflow-integrations/integrations/netwitness** RSA NetWitness integration directory
2. Run the **first-time-setup.sh** script, and ensure that the newly created **netwitness.conf** and **last.conf** files are readable and writable by all users – `chmod` may be used as needed.
3. Fill in the **netwitness.conf** file's **Host** field with the host address of your RSA NetWitness instance, and **username** and **password** fields with your username and password on said instance respectively. The `disable_ssl` field should default to `False`, but can be set to `True` to disable SSL verification with the RSA NetWitness instance.

Running, Stopping, and Checking on the Service

The incident pulling service can be started by running the `enable-netwitness.sh` script as root within the RSA NetWitness integration directory. It will run once when this script is run, and then continuously thereafter until stopped. Stopping the service is as simple as running the `disable-netwitness.sh` script as root within the same directory. The status of the service can be checked on through the following command:

```
systemctl status -l irflow-netwitness.service
```

IR-Flow Alert Output

Back in the IR-Flow web interface Triage, once the service is running, you will find new alerts appearing with Source and Description values of NetWitness Incident. Within these, you will find fields representing the incident's ID and title, a URL link to the incident on the RSA NetWitness instance, and expandable data fields filled with data for the incident itself, and its list of associated alerts.

Alert Facts		Show Empty Facts (0) ^
Incident Alerts	▶ array [18]	
Incident ID	INC-39	
Original Alert URL	https://10.0.1.36/respond/incident/INC-39	
Results	▶ object {24}	
Title	High Risk Alerts: ESA for 70.0	

Alert Facts		Show Empty Facts (0) ^
Incident Alerts	▼ array [18] <ul style="list-style-type: none"> ▶ 0 {8} ▶ 1 {8} ▶ 2 {8} ▶ 3 {8} ▶ 4 {8} ▶ 5 {8} ▶ 6 {8} ▶ 7 {8} ▶ 8 {8} ▶ 9 {8} ▶ 10 {8} ▶ 11 {8} ▶ 12 {8} ▶ 13 {8} ▶ 14 {8} ▶ 15 {8} ▶ 16 {8} ▶ 17 {8} 	
Incident ID	INC-39	
Original Alert URL	https://10.0.1.36/respond/incident/INC-39	
Results	▶ object {24}	
Title	High Risk Alerts: ESA for 70.0	

Certification Checklist for RSA NetWitness

Date Tested: October 22, 2018

Certification Environment		
Product Name	Version Information	Operating System
RSA NetWitness	11.1	Virtual Appliance
Syncurity IR Flow	5.x	AWS

RSA NetWitness Test Case	Result
Inline Query/Enrichment	
Query NetWitness for IP Info (source/destination IP)	N/A
Query NetWitness for User Info (usernames, user behavior)	N/A
Query NetWitness for Specific Meta (Other)	N/A
Retrieve NetWitness Log/Packet Data	N/A
Retrieve NetWitness PCAP files	N/A
Alerting / Incident Creation	
NetWitness alert via syslog	N/A
NetWitness alert via email	N/A
NetWitness alert via ESA/scripting	N/A
Send alert to NetWitness (Syslog, CEF, or custom parser)	N/A
Alerts and Incident Retrieval	
Collect NetWitness Alert	✓
Collect NetWitness Incident	✓
RSA NetWitness Intel Feeds	
Update NetWitness Intel Feed (CSV, STIX)	N/A

✓ = Pass ✗ = Fail N/A = Non-Available Function