RSA® NETWITNESS®
Security Operations Implementation Guide

EclecticIQ Platform 2.3

Daniel R. Pintal, RSA Partner Engineering
Last Modified: December 17, 2018

RSA READY

Solution Summary

Through its use of open standards and technologies, EclecticIQ's integration provides threat intelligence to enrich metadata (meta) gathered by RSA NetWitness. If an IP or DNS contained within EclecticIQ's threat intelligence is found within meta collected by RSA NetWitness, EclecticIQ's meta is appended to the event.

RSA NetWitness Features	
EclecticIQ Platform 2.3	
Feed format	STIX xml
Collection method	http, local file
Feed Collection Frequency	Hourly, Daily, Weekly

Partner Product Configuration

Before You Begin

This section provides instructions for configuring the EclecticIQ Platform 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 to install the required components.

All EclecticIQ 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. For more information, please refer to the EclecticIQ Platform documentation or website.

EclecticIQ Platform Configuration

EclecticIQ Platform integrates with RSA NetWitness via STIX XML files. Structured Threat Information Expression (STIX) is a structured language for describing cyber threat information it can be shared, stored, and analyzed in a consistent manner. RSA NetWitness supports the import of STIX Indicators and STIX Observables to improve threat detection through hunting and ESA notification.

STIX files can either be manually downloaded from EclecticIQ Platform or can be generated via the Trusted Automated eXchange of Indicator Information (TAXII) API. For more information on using the TAXII API, consult the EclecticIQ Platform product documentation here:

https://docs.eclecticiq.com/configure-data-sources/outgoing-feeds/types-of-outgoing-feeds/taxii-poll-feed

To integrate with RSA NetWitness, create at least one Outgoing Feed on the EclecticIQ Platform with Transport **TAXII Poll** and Content type **STIX 1.2**.

> **Important: STIX files with multiple observables or indicators must have only one </stix:STIX_Package> element in the XML. In RSA NetWitness, a STIX (.xml) feed of type Indicator or Observable which contains properties such as the IP addresses, File hashes, Domain names, and URLs are supported.**

EclecticIQ recommends creating a separate outgoing feed for each observable, ex. **hashes, URIs, Domain Names, and emails**. In addition, create two IP feeds, one to alert RSA NetWitness and a second to generate sightings which are sent back to EclecticIQ via a script. The feeds will have the same content but different TAXII collection Names, one for source IPs alerting and the second for destination IPs. Finally, to send all available data from EclecticIQ Platform to RSA NetWitness configure six different outgoing feeds with different TAXII collection Names: **ip-source, ip-destination, URI, domains, hashes and emails**.

To limit the types of indicators or observables supported, use either the **Dataset** or **Observable and Enrichment Observable types** option or both for the outgoing feed settings. It is recommended to create specific **Outgoing Feed** for each type of object: **IP addresses, File hashes, Domain names, and URLs**. Consult the EclecticIQ Platform product documentation here:

https://docs.eclecticiq.com/configure-data-sources/datasets/create-a-dataset

https://docs.eclecticiq.com/configure-data-sources/outgoing-feeds/set-observable-filters-for-outgoing-feeds

EclecticIQ Platform Sightings User Configuration

For sightings generation purposes you must be authorized in EclecticIQ Platform and therefore you need to create a user with specific Roles and Groups. For security reasons it is not recommended to use generic users for this, especially users with admin/full rights.

Process of EclecticIQ Platform user creation described here in details:

https://docs.eclecticiq.com/configure-data-sources/account-policies

1. Create a role. Go to **Settings -> User Management -> Roles**, select **Create Role**, type name for example **rsa_sightings_role** and add permissions: **read groups**, **modify entities** and **read entities**.
2. Create a group. Go to **Settings -> User Management -> Groups**, select **Create Group**, type name for example **rsa_sightings_group**.
3. Create a user. Go to **Settings -> User Management -> Users**, select **Create User**, fill the form, assign Group: **rsa_sightings_group** and assign Role: **rsa_sightings_role**. Do not set Administrator checkbox for the user.
4. If the Partner Product can update RSA NetWitness Intel feeds, provide instructions on the feed format (.csv or STIX) and how the file is transported to NetWitness.

RSA NetWitness Configuration

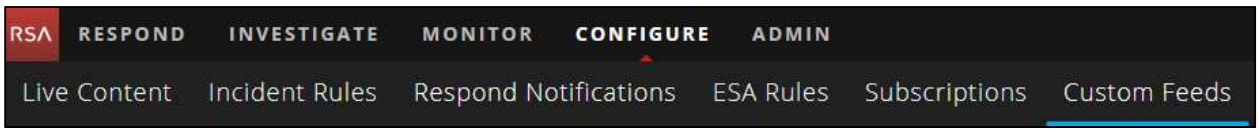
RSA NetWitness Custom Feed Configuration

Depending on your deployment and if you have elected to add an RSA NetWitness Log Decoder and/or Packet Decoder follow the steps below for your integration.

Log Decoder Configuration

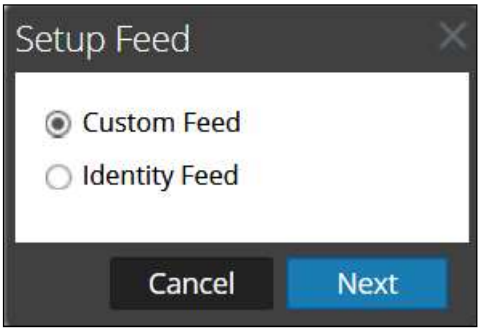
RSA NetWitness Feed Configuration

1. From the RSA NetWitness Dashboard Select **CONFIGURE** > **CUSTOM Feeds**.

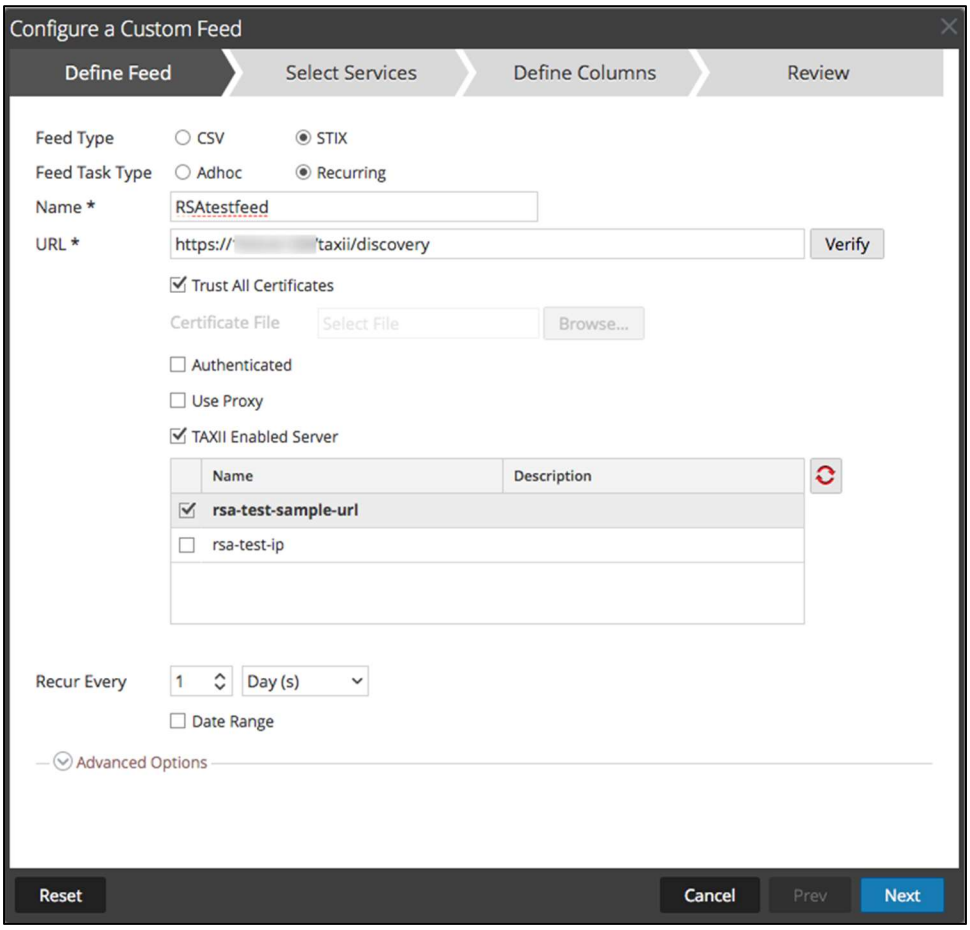


2. Select the + in the Feeds Window to setup the feed.

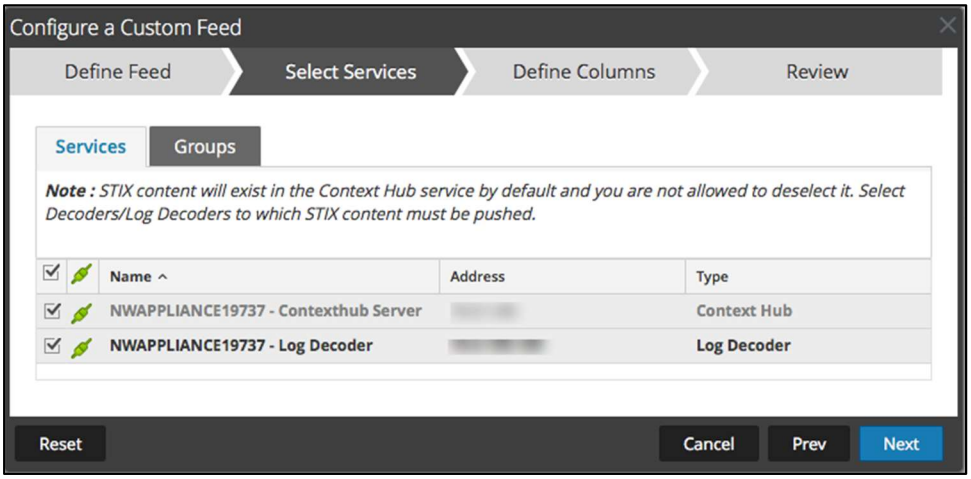
3. Select the **Custom Feed** radio button within the Setup Feed pop-up window and select **Next**.



4. Select **Adhoc** if you are uploading the file once **or** select the **Recurring** radio button if you plan to automate the feed. Enter the URL of the Feed provider and select how often to pull the feed by setting the Recur Every option and select **Next**. For using TAXII transport enter the **URL*** for the **Discovery Service** of EclecticIQ Platform. If you have specific permissions, select the **Authenticated** checkbox and enter credentials. You can use only one Collection in each feed, but you can create multiple feeds with different collections from one TAXII server.



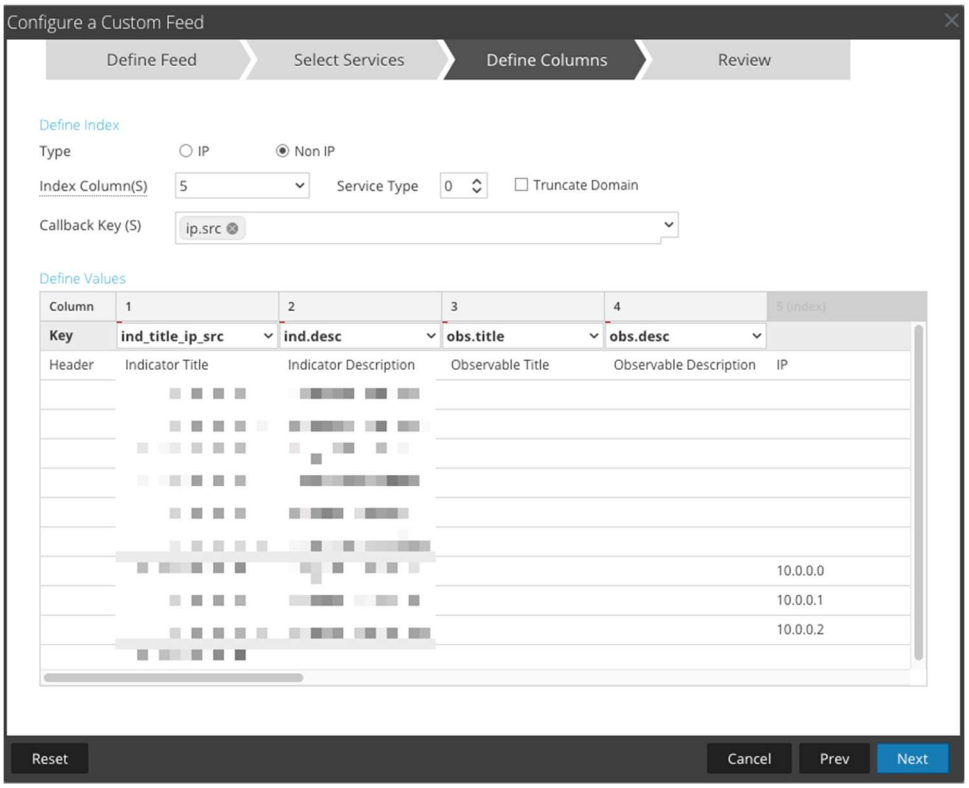
5. Select the **RSA NetWitness Log Decoder Service checkbox** and select **Next**.



Important: Due to the ESA alerting workflow and sightings generation EclecticIQ recommends using "Non IP" type for any feed.

6. The example below is for Source IP matching and alerting. Define the Index as Type **Non IP**, **Index Column 5** (the one with IP value from indicator/observable), **Service Type 0** and use **Callback Key (S)** to select which fields you wish to check for indicator, for source IP it is ip.src. For ESA alerting set the name of first column connected with Callback key, **ind_title_ip_src**. For additional fields see table below. Set the header of the other columns as needed and select **Next**.

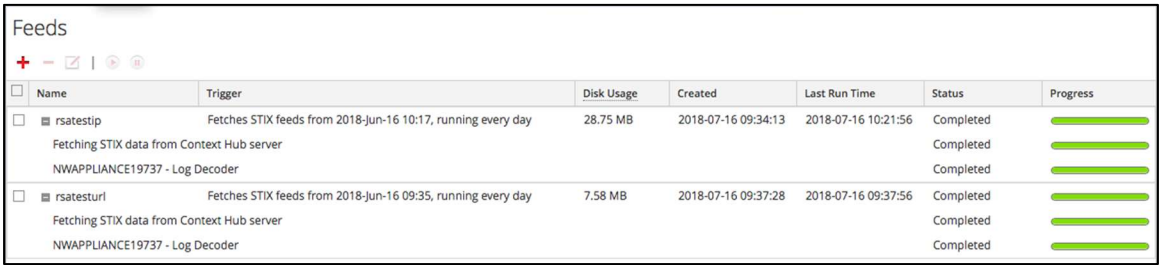
Important: The graphic below displays a configuration using additional fields added manually (obs.title, ind.desc etc), sample configuration provided in Appendix A.



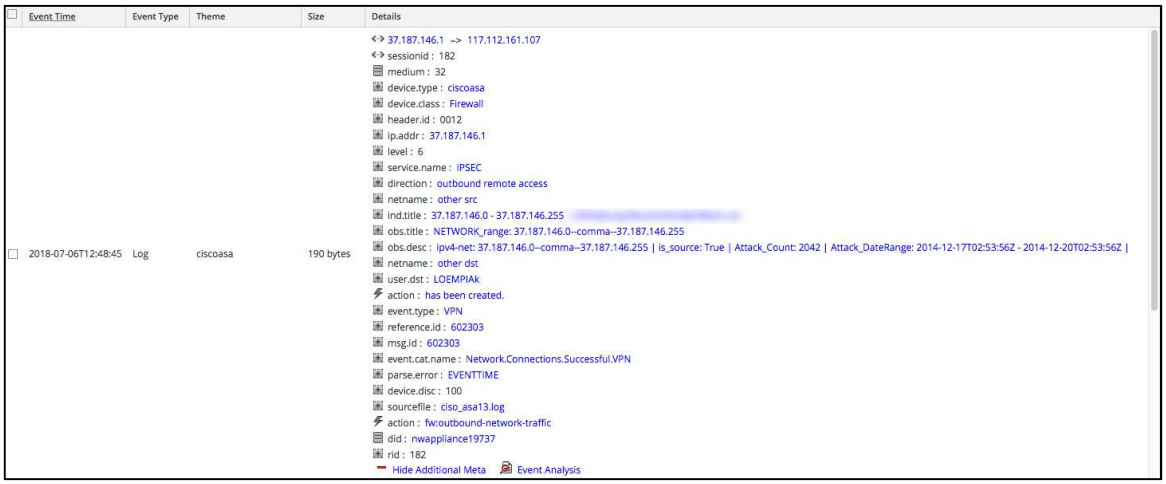
EclecticIQ Outgoing Feed contains	RSA NetWitness Callback Key(s)	ESA alerting field	Index Column
IP (source)	ip.src	ind_title_ip_src	IP (#5)
IP (destination)	ip.dst	ind_title_ip_dst	IP (#5)
Hashes	Select custom field which is used	ind_title_hash	File Hash Value (#17)
URI	uri	ind_title_uri	URI (#7)
Domain	domain.dst, domain.src, domain	ind_title_domain	Domain (#6)
Email	email	ind_title_email	Email (#8)

7. Select **Finish** to complete the setup of the Feed Integration. Repeat steps 2-7 with outgoing feeds in EclecticIQ Platform.

8. During the feed deployment, initially the status will appear as **Waiting** and the Progress will be **yellow** until RSA NetWitness completes the transfer of the Feed. Once deployed the Status will display **Completed** and the Progress will be **green**. Depending on the size of the feed it may take some time for RSA NetWitness to download all Threat Intel from EclecticIQ Platform.



9. Once completed and if you have any IOC's, the EclecticIQ Feed meta will appear within RSA NetWitness Investigator appended to the collected event.



RSA NetWitness ESA Alerting configuration

RSA NetWitness ESA alerting can be implemented with data ingested from the EclecticIQ Platform.

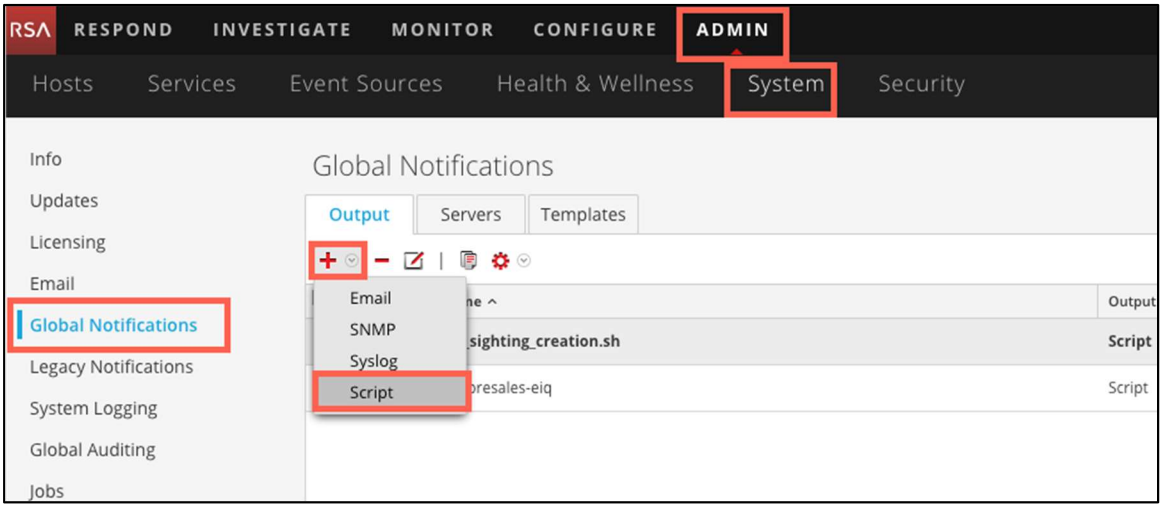
When NetWitness collects an event matching an IOC field contained within the EclecticIQ threat intelligence feed, NetWitness enriches the collected event by merging associated meta contained within the EclecticIQ Feed.

Upon detection of the IOC, NetWitness ESA alerting can be configured based on these meta fields, for example **ind_title_ip_src** to perform an ESA Notification such as sending an Email, SNMP Trap, Syslog Event or run a script.

The following instructions provide an example of how to deploy the EclecticIQ sightings script in RSA NetWitness.

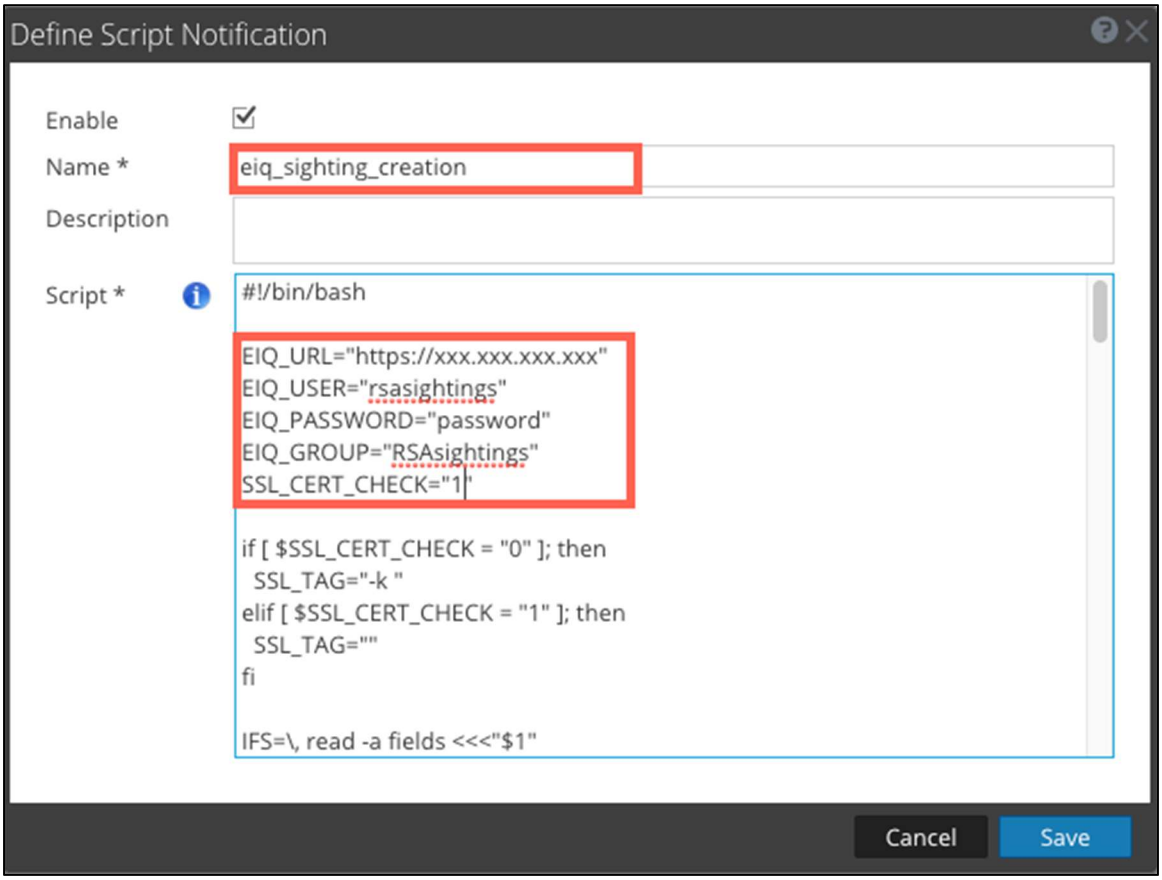
Adding Sightings generation script

1. To add script for sightings generation in NetWitness select, **Admin** > **System** > **Global Notifications** > **Output**. Select + > **Script**.



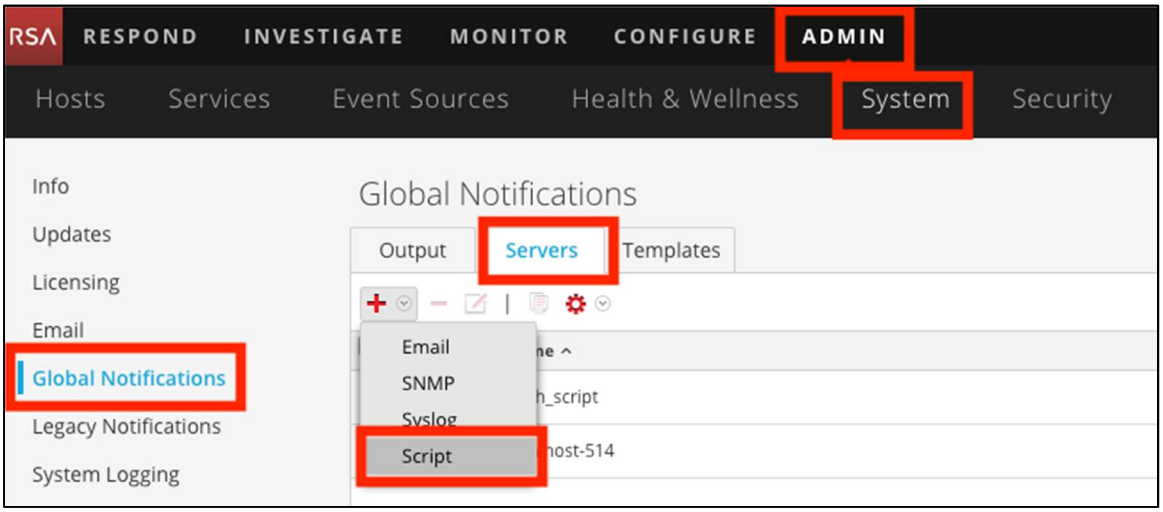
2. In the new window fill name for the script, for example **eiq_sighting_creation** and copy and paste script from **Appendix B** to the **Script*** field. If you have any issues with copying and pasting (new line symbols, spaces etc) from this document submit a request to EclecticIQ for the script at **support@eclecticiq.com**. In the beginning of script, replace the account parameters with your own.

```
EIQ_URL=https://xxx.xxx.xxx.xxx   // name or IP address of EclecticIQ Platform
EIQ_USER=" rsasightings "         // enter user name created in the Platform.
Described above
EIQ_PASSWORD="password"           // enter password for user in the Platform.
Described above
EIQ_GROUP=" RSAsightings "        // enter group name created in the Platform.
Described above
SSL_CERT_CHECK="1"                // enter 1 to check SSL cert or 0 to ignore
it.
```

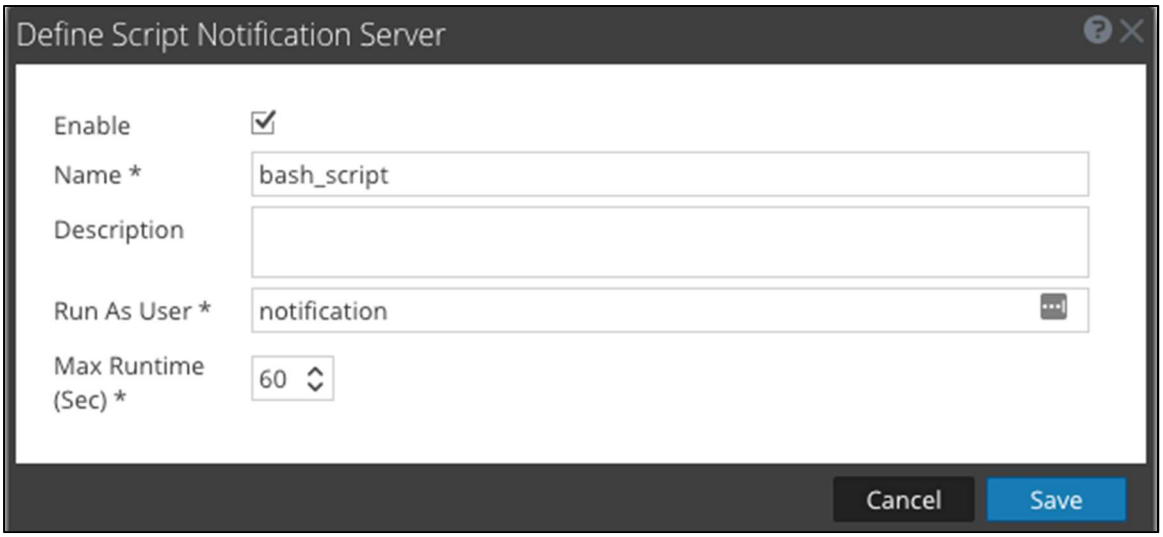


Adding Notification server

1. To add a **NetWitness Global Notifications server** for script execution select **Admin** > **System** > **Global Notifications** > **Servers,** select **add** and select **Script**.



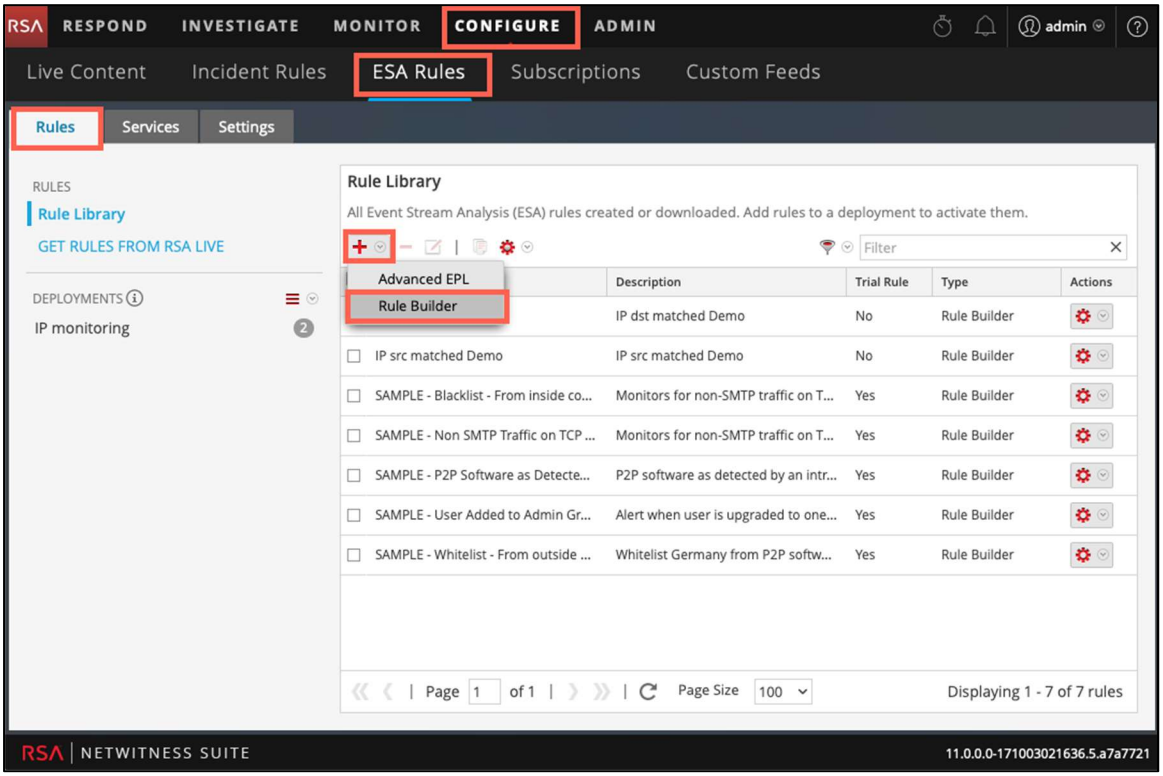
2. In the new window fill the Name of Script Server, for example **bash_script** and other parameters.



Creating ESA alert for Sightings generation

At this step we have Script for sightings generation and Notification Server in NetWitness now we can create ESA Rules for alerting with sightings generation.

1. To create **NetWitness ESA Rule** to select **Configure** > **ESA Rules** > **Add new Rule** > **Rule Builder**.



2. NetWitness Rule Builder supports complex rules however in this guide a simple rule is used to detect matching events with data from the EclecticIQ Threat Feed and to send Sightings back to the EclecticIQ Platform.
3. For basic functionality create an alert with following parameters:

   - **Rule name:** IP src matched.
   - **Description:** Threat Intelligence matching rule.
   - **Conditions:** event.ind_title_ip_src is not null.
     Select the field for condition based on the alert field, in this case for Source IP we use event.ind_title_ip_src.
   - **Notifications:** Select + >Script, select name of script imported before into Notification column, bash_script as Notification Server, template – Default Script Template.

4. **Save and apply** changes in ESA alerting. If you ingest all the possible data from EclecticIQ Platform and want to have alerts for different IOC's, create rules for the following: **ip source, ip destination, hash, email, domain and uri**.

> **Important: If you don't see custom fields in ESA Rule Builder you need to update Schema in ESA, see details.**
>
> https://community.rsa.com/docs/DOC-78096
>
> **or** https://community.rsa.com/docs/DOC-78083

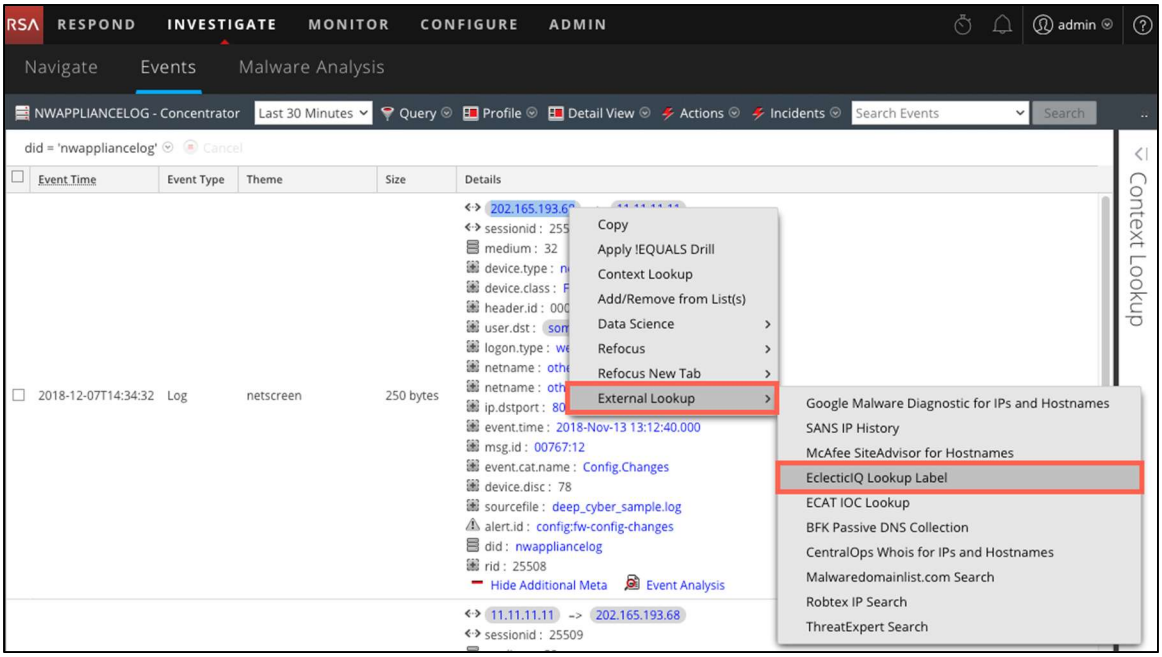
5. With these rules if RSA NetWitness collects a security event containing an IOC, ESA creates an alert and runs the script which in turn creates an EclecticIQ sighting.
6. To test the EclcecticIQ sightings script connect to the CLI of the NetWitness ESA server, open the **/opt/rsa/esa/scripts** folder and locate the appropriate script to be tested. NetWitness script names are randomly generated. Once the correct script to test has been identified, run it locally with the following parameters:

```
#./5bfbf363e4b0f6722f8e79ee '"eiq_alert_src_ip" : "10.10.0.10", "ip_src" :
"10.10.0.10"'
```

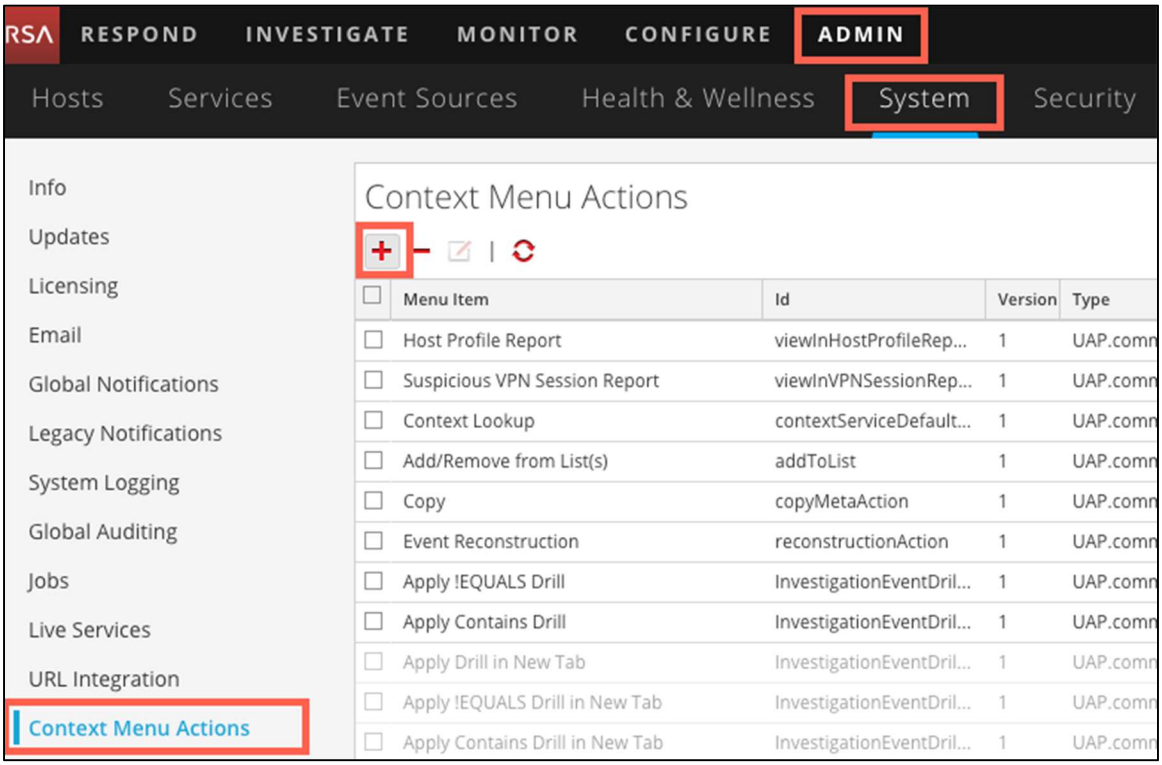
7. If successful, a detailed output will be sent to the EclecticIQ Platform and new sightings will be displayed in the Platform web UI.

RSA NetWitness Context Menu action configuration

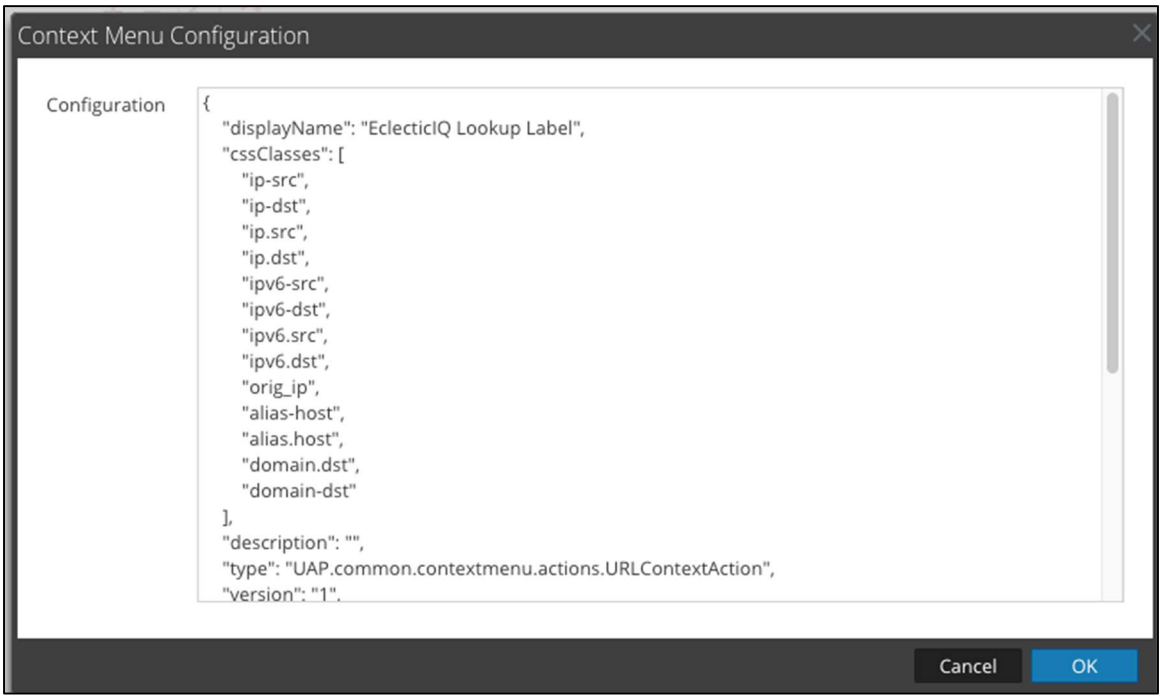
NetWitness Context Menu actions are used to link an IOC displayed within NetWitness Investigate to a third party and provide additional details of the threat indicator. The Context Menu action is initiated by right-clicking the IOC to open a web url or by initiating a script to run against the EclecticIQ Platform to display additional details of the IOC via a browser.



1. To add Context menu action, select **Admin** > **System** > **Context Menu Actions** select +.



2. Copy and paste the configuration code to text field in the new window. The configuration code located in **Appendix C** of this document. Edit the **urlFormat** field and replace its value with yours own.
3. For RSA NetWitness version 11.1+ users, the interface may be different, select **Switch to Advanced View**.



4. Select **OK**, and Context Menu/Right-Click actions are enabled for EcelecticIQ.

EclecticIQ Fusion Center integration with RSA NetWitness

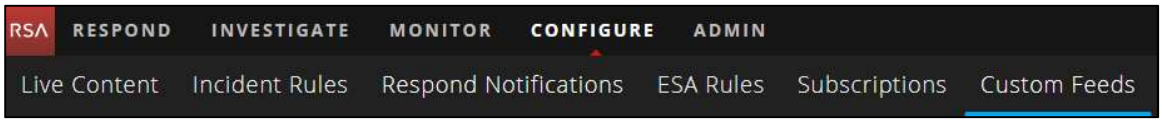
Depending on your deployment and if you have elected to add an RSA NetWitness Log Decoder and/or Packet Decoder follow the steps below for your integration.

To extend the functionality of RSA NetWitness Feeds for use with NetWitness rules and notifications please refer to **http://sadocs.emc.com/**.

Log Decoder Configuration

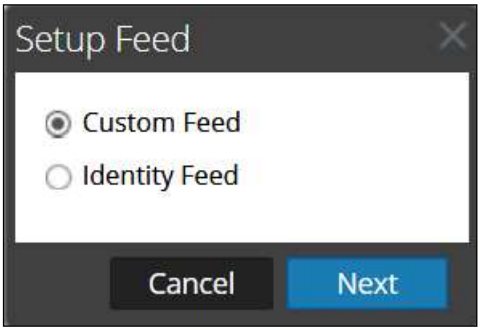
RSA NetWitness Feed Configuration

1. From the RSA NetWitness Dashboard Select **CONFIGURE** > **Custom Feeds**.

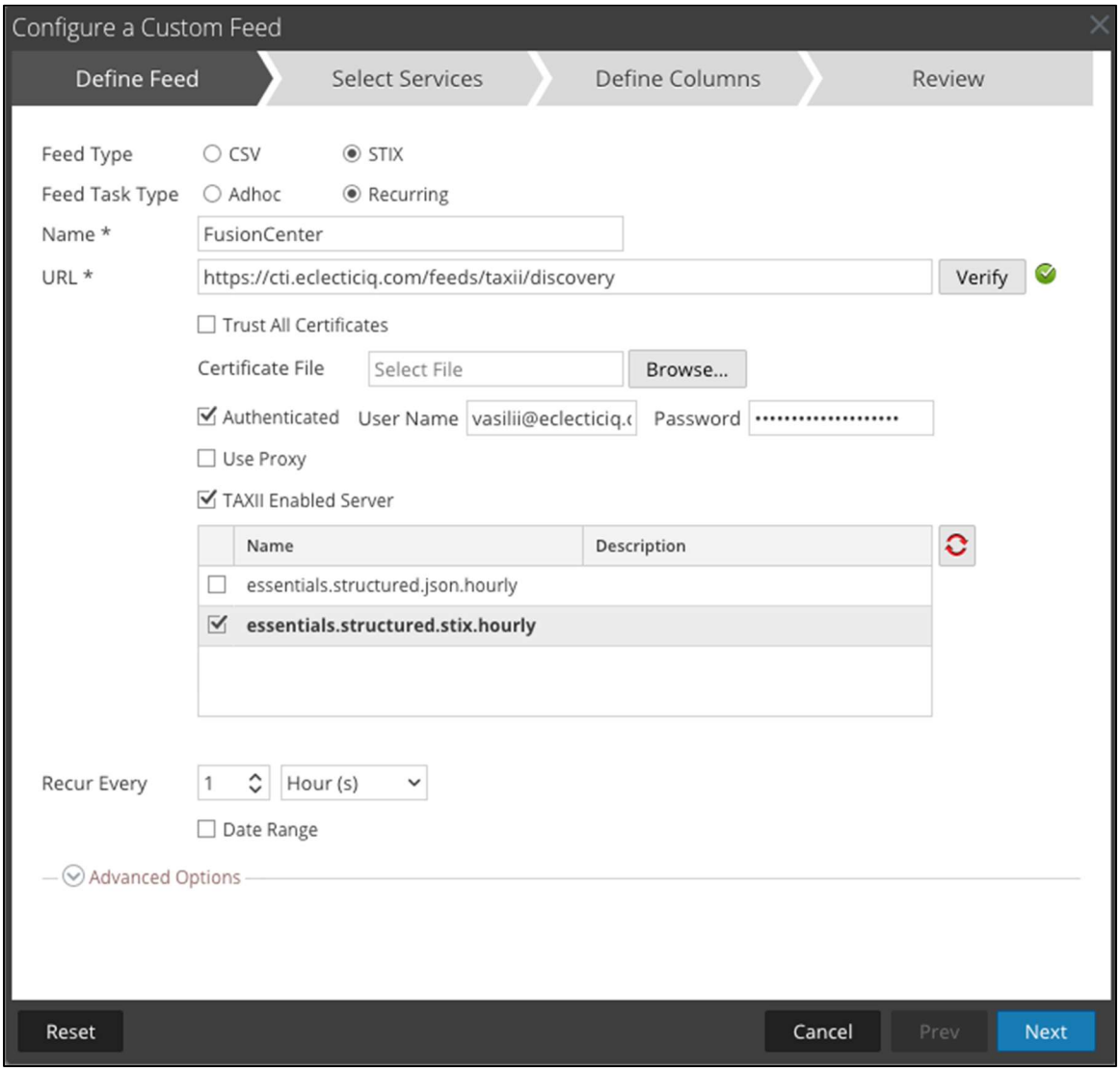


2. Select the + in the Feeds Window to setup the feed.

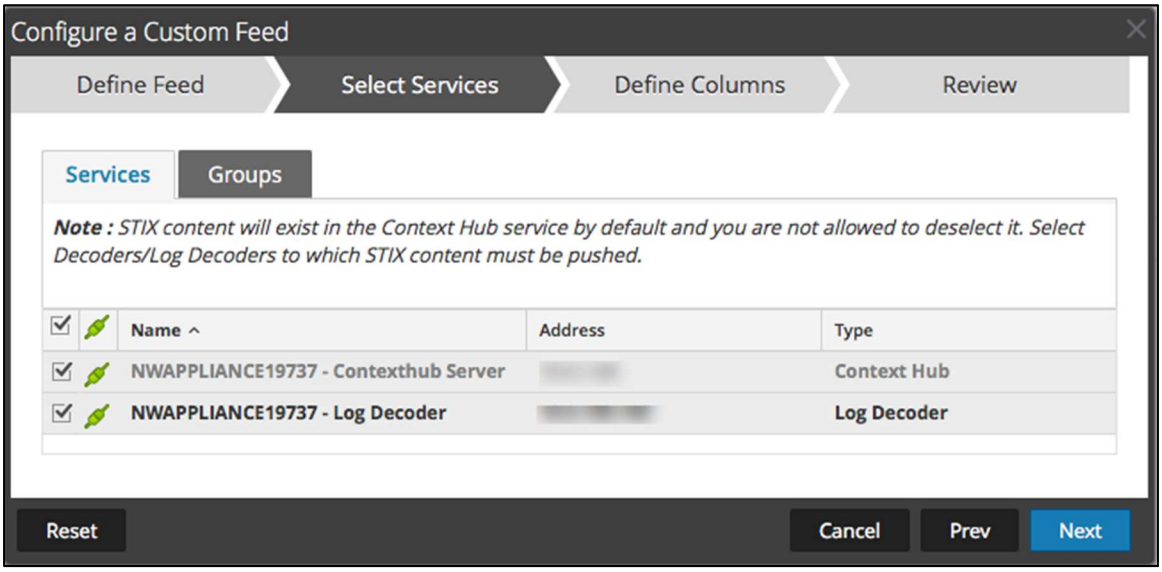
3. Select the **Custom Feed** radio button within the Setup Feed pop-up window and select **Next**.



4. Fill the parameters:
   - Feed Type: STIX
   - Feed Task Type: Recurring
   - URL: **https://cti.eclecticiq.com/feeds/taxii/discovery**
   - Deselect Trust All Certificates.
   - Select TAXII Enabled Server checkbox.
   - Fill your credentials to the Authenticated field.
   - Select stix feed in the list of available feeds.
   - Select how often to pull the feed by setting the **Recur Every** option and select **Next**.

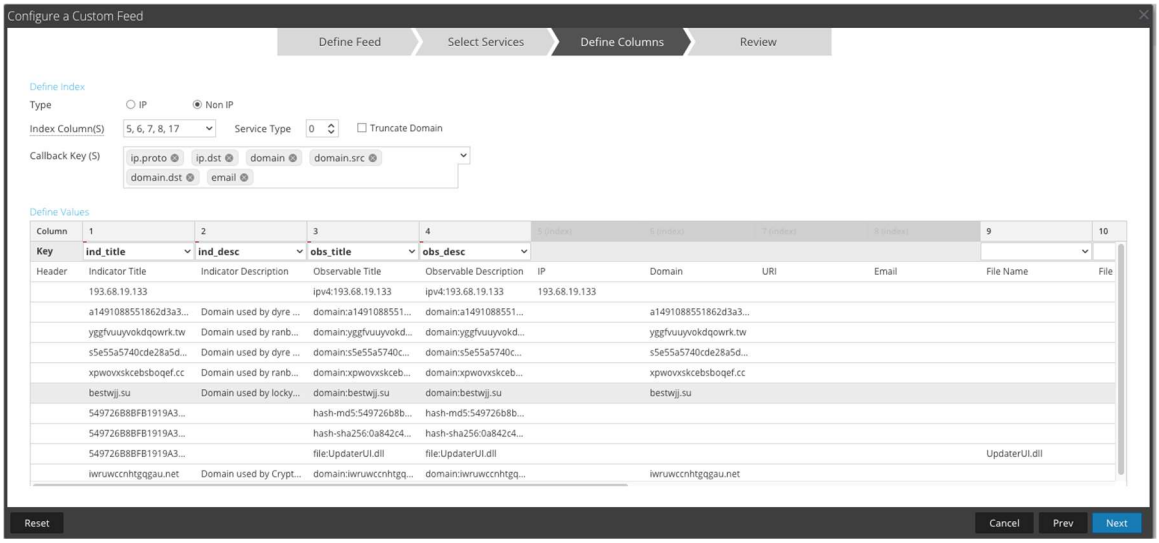


5. Select the **RSA NetWitness Log Decoder Service checkbox** and select **Next**.

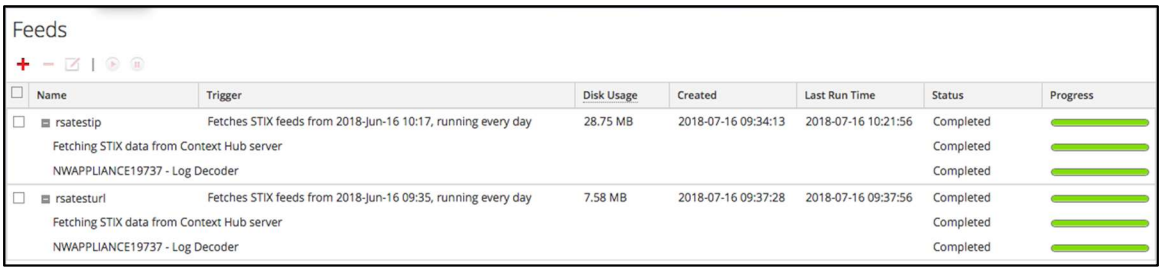


6. Select following parameters:
   - Type: Non IP
   - Index Column (S): IP (#5), Domain (#6), URI (#7), Email (#8), Hash(#17).
   - Callback Key (S): ip.src, ip.dst, domain, domain.src, domain.dst, email, uri and any other fields which you want to match with feed data.
7. Enter the following keys to the column headers:
   - Indicator Title: ind_title
   - Indicator Description: ind_desc
   - Observable Title: obs_title
   - Observable Description: obs_desc

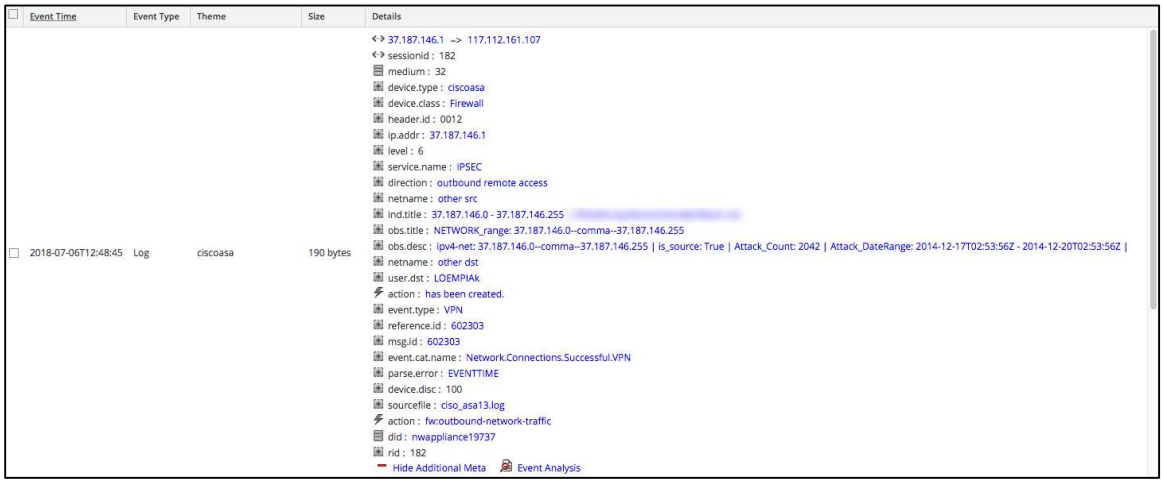
> **Important: The graphic below displays a configuration using additional fields added manually (obs.title, ind.desc etc), sample configuration provided in Appendix A.**



8. Select **Finish** to complete the setup of the Feed Integration.
9. Initially the status will appear as **Waiting** and the Progress will be **yellow** until RSA NetWitness completes the transfer of the Feed. Once completed the Status will display **Completed** and the Progress will be **green**. Depending on the size of the feed it may take some time for RSA NetWitness to download all Threat Intel from EclecticIQ Platform.



10. Once completed and if you have any threat events, the meta will appear within RSA NetWitness Investigator.



Certification Checklist for RSA NetWitness

Date Tested: December 17, 2018

Certification Environment		
Product Name	**Version Information**	**Operating System**
RSA NetWitness	11.2	Virtual Appliance
EclecticIQ Platform	2.3	Virtual Appliance

RSA NetWitness Test Case	Result
Inline Query/Enrichment Threat Intelligence Feed is received through Log Decoder Meta	✓
Alerting Alert Third Party via ESA/scripting	✓
Intelligence Feed Import Import Intelligence Feed (CSV, STIX)	✓
Context Menu Actions Right-Click Action to redirect for Third Party lookup/reference	✓

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

Appendix A - Sample Custom Meta Keys

RSA NetWitness provides a number of out of the box keys that can be integrated with a custom feed such as threat.source, threat.category, threat.description, etc. However, if you wish to create custom meta keys for use with a custom feed, such as EclecticIQ, you can do so following the instructions found here:

https://community.rsa.com/docs/DOC-80195

A sample snippet of entries into the **index-concentrator-custom.xml** file is provided below. Please note that this is provided as an example only, as additional fields can be included or excluded as needed.

```
<key description=" EclecticIQ Indicator title for src IP" format="Text"
level="IndexValues" name="ind_title_ip_src" valueMax="250000" defaultAction="Open"/>
<key description=" EclecticIQ Indicator title for dst IP" format="Text"
level="IndexValues" name="ind_title_ip_dst" valueMax="250000" defaultAction="Open"/>
<key description=" EclecticIQ Indicator title for hash" format="Text"
level="IndexValues" name="ind_title_hash" valueMax="250000" defaultAction="Open"/>
<key description=" EclecticIQ Indicator title for email" format="Text"
level="IndexValues" name="ind_title_email" valueMax="250000" defaultAction="Open"/>
<key description=" EclecticIQ Indicator title for URI" format="Text"
level="IndexValues" name="ind_title_uri" valueMax="250000" defaultAction="Open"/>
<key description=" EclecticIQ Indicator title for domain" format="Text"
level="IndexValues" name="ind_title_domain" valueMax="250000" defaultAction="Open"/>
<key description=" EclecticIQ Indicator title" format="Text" level="IndexValues"
name="ind_title" valueMax="250000" defaultAction="Open"/>
<key description=" EclecticIQ Indicator Description" format="Text" level="IndexValues"
name="ind_desc" valueMax="250000" defaultAction="Open"/>
<key description=" EclecticIQ Observables Title" format="Text" level="IndexValues"
name="obs_title" valueMax="250000" defaultAction="Open"/>
<key description=" EclecticIQ Observables Description" format="Text"
level="IndexValues" name="obs_desc" valueMax="250000" defaultAction="Open"/>
```

Appendix B – Alerting Script for ESA

In the following configuration you have to replace values at the beginning with yours. Additional description for parameters:

```
EIQ_URL="https://eiqtip.localdomain"   // URL or IP of EclecticIQ Platform
EIQ_USER="user1"                       // User created for sightings generation in the
                                       // EclecticIQ Platform. Do not use admin user!
EIQ_PASSWORD="Password"                // Password for EclecticIQ user
EIQ_GROUP="Testing"                    // Group name assigned to EclecticIQ user
SSL_CERT_CHECK="0"                     // Enter "0" to not to validate SSL cert, enter "1" for
                                       // validation
```

If you have any issues with script copying because of document format you can request it **support@eclecticiq.com**

Script text:

```
#!/bin/bash

EIQ_URL="https://eiqtip.localdomain"
EIQ_USER="user1"
EIQ_PASSWORD="P@ssw0rd123"
EIQ_GROUP="Testing"
SSL_CERT_CHECK="0"

if [ $SSL_CERT_CHECK = "0" ]; then
  SSL_TAG="-k "
elif [ $SSL_CERT_CHECK = "1" ]; then
  SSL_TAG=""
fi

IFS=\, read -a fields <<<"$1"

for x in "${fields[@]}";do
        if [[ "$x" =~ ind_title* ]]; then
                read eiq_alert_type <<<$(echo "$x" | awk '{ print $1 }' | sed s/\"//g)
        fi
    done

echo $eiq_alert_type

if [ $eiq_alert_type = "ind_title_ip_src" ]; then
  eiq_sighting_type="ipv4"
  for x in "${fields[@]}";do
    echo "$x"
        if [[ "$x" =~ \"ip_src\" ]]; then
                echo "$x"
                read eiq_sighting_value <<<$(echo "$x" | awk '{ print $3 }' | sed s/\"//g)
        fi
    done
elif [ $eiq_alert_type = "ind_title_ip_dst" ]; then
  eiq_sighting_type="ipv4"
  for x in "${fields[@]}";do
        if [[ "$x" =~ \"ip_dst\" ]]; then
                echo "$x"
                read eiq_sighting_value <<<$(echo "$x" | awk '{ print $3 }' | sed s/\"//g)
        fi
    done
elif [ $eiq_alert_type = "ind_title_hash" ]; then
  eiq_sighting_type="hash-md5"
  for x in "${fields[@]}";do
    if [[ "$x" =~ \"hash\" ]]; then
        echo "$x"
        read eiq_sighting_value <<<$(echo "$x" | awk '{ print $3 }' | sed s/\"//g)
    fi
    done
elif [ $eiq_alert_type = "ind_title_email" ]; then
  eiq_sighting_type="email"
  for x in "${fields[@]}";do
    if [[ "$x" =~ \"email\" ]]; then
        echo "$x"
        read eiq_sighting_value <<<$(echo "$x" | awk '{ print $3 }' | sed s/\"//g)
    fi
    done
```

```
elif [ $eiq_alert_type = "ind_title_uri" ]; then
  eiq_sighting_type="uri"
  for x in "${fields[@]}";do
    if [[ "$x" =~ \"uri\" ]]; then
        echo "$x"
        read eiq_sighting_value <<<$(echo "$x" | awk '{ print $3 }' | sed s/\"//g)
    fi
    done
elif [ $eiq_alert_type = "ind_title_domain" ]; then
  eiq_sighting_type="domain"
  for x in "${fields[@]}";do
    if [[ "$x" =~ \"domain_src\" ]]; then
        echo "$x"
        read eiq_sighting_value <<<$(echo "$x" | awk '{ print $3 }' | sed s/\"//g)
    elif [[ "$x" =~ \"domain_dst\" ]]; then
        echo "$x"
        read eiq_sighting_value <<<$(echo "$x" | awk '{ print $3 }' | sed s/\"//g)
    elif [[ "$x" =~ \"domain\" ]]; then
        echo "$x"
        read eiq_sighting_value <<<$(echo "$x" | awk '{ print $3 }' | sed s/\"//g)
    fi
    done
fi

read AUTH_TOKEN <<< $(curl $SSL_TAG -X POST "$EIQ_URL/api/auth" -H "accept: application/json" -H
"Content-Type: application/json" -d "{ \"username\": \"$EIQ_USER\",\"password\": \"$EIQ_PASSWORD\"}" |
grep 'token' | awk '{ print $2 }' | sed s/\"//g)
read GROUP_ID <<< $(curl $SSL_TAG -X GET -G "$EIQ_URL/private/groups/" -d "filter[name]=$EIQ_GROUP" -H
"Authorization:Bearer $AUTH_TOKEN" -H "accept: application/json" -H "Content-Type: application/json" |
grep -m 1 '\"source' | awk '{ print $2 }' | sed s/\"//g)
read CURRENT_TIME <<< $(date '+%Y-%m-%dT%H:%M:%S')
read DESCRIPTION <<< $(echo $1 | sed 's/"//g')

SIGHTING="{\"data\": {
                \"data\": {
                    \"confidence\": {
                        \"type\": \"confidence\",
                        \"value\": \"Medium\"
                    },
                    \"description\": \"Sighting automatically created from RSA NetWitness ESA scripted
action. $DESCRIPTION\",
                    \"related_extracts\": [
                        {
                            \"type\": \"eclecticiq-extract\",
                            \"kind\": \"$eiq_sighting_type\",
                            \"value\": \"$eiq_sighting_value\"
                        }
                    ],
                    \"description_structuring_format\": \"html\",
                    \"type\": \"eclecticiq-sighting\",
                    \"title\": \"NetWitness Sighting $eiq_sighting_type $eiq_sighting_value\",
                    \"security_control\": {
                        \"type\": \"information-source\",
                        \"identity\": {
                            \"name\": \"EclecticIQ Platform App for NetWitness\",
                            \"type\": \"identity\"
                        },
                        \"time\": {
                            \"type\": \"time\",
                            \"start_time\": \"start_time\",
                            \"start_time_precision\": \"second\"
                        }
                    }
                },
                \"meta\": {
                    \"taxonomy\": [],
                    \"tags\": [\"NetWitness Alert\"],
                    \"ingest_time\":  \"$CURRENT_TIME\"
                },
                \"sources\": [{
                    \"source_id\": \"$GROUP_ID\"
                }]
            }}"

curl $SSL_TAG -X POST "$EIQ_URL/private/entities/" -H "Authorization:Bearer $AUTH_TOKEN" -H "accept:
application/json" -H "Content-Type: application/json" -d "$SIGHTING"
```

Appendix C – Context Menu Configuration

In the following configuration you have to find urlFormat field and replace its value with yours. For example, if the EclecticIQ Platform address is https://local-eiq-platform.localdomain field value has to be: "**https://local-eiq-platform.localdomain/search/observable?q={0}**",

Configuration text:

```
{
    "displayName": "EclecticIQ Lookup",
    "cssClasses": [
        "ip-src",
        "ip-dst",
        "ip.src",
        "ip.dst",
        "ipv6-src",
        "ipv6-dst",
        "ipv6.src",
        "ipv6.dst",
        "orig_ip",
        "alias-host",
        "alias.host",
        "domain.dst",
        "domain-dst"
    ],
    "description": "",
    "type": "UAP.common.contextmenu.actions.URLContextAction",
    "version": "1",
    "modules": [
        "investigation"
    ],
    "local": "false",
    "groupName": "externalLookupGroup",
    "urlFormat": "https://eiqplatform.localdomain/search/observable?q={0}",
    "disabled": "",
    "id": "EclecticIQAction",
    "moduleClasses": [
        "UAP.investigation.navigate.view.NavigationPanel",
        "UAP.investigation.events.view.EventGrid"
    ],
    "openInNewTab": "true",
    "order": "15"
      }
```