

# **RSA** | NetWitness Suite

IBM QRadar Integration

## **Trademarks**

RSA, the RSA Logo and EMC are either registered trademarks or trademarks of EMC Corporation in the United States and/or other countries. All other trademarks used herein are the property of their respective owners. For a list of EMC trademarks, go to [www.emc.com/legal/emc-corporation-trademarks.htm](http://www.emc.com/legal/emc-corporation-trademarks.htm).

## **License Agreement**

This software and the associated documentation are proprietary and confidential to EMC, are furnished under license, and may be used and copied only in accordance with the terms of such license and with the inclusion of the copyright notice below. This software and the documentation, and any copies thereof, may not be provided or otherwise made available to any other person.

No title to or ownership of the software or documentation or any intellectual property rights thereto is hereby transferred. Any unauthorized use or reproduction of this software and the documentation may be subject to civil and/or criminal liability. This software is subject to change without notice and should not be construed as a commitment by EMC.

## **Third-Party Licenses**

This product may include software developed by parties other than RSA.

## **Note on Encryption Technologies**

This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when using, importing or exporting this product.

## **Distribution**

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license. EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." EMC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

# Contents

---

<b>RSA NetWitness Suite Integration with IBM QRadar .....</b>	<b>4</b>
<b>Forwarding Alerts: NetWitness to QRadar .....</b>	<b>5</b>
Event Stream Analysis CEF Template .....	5
Reporting Engine Alert Syslog CEF Template .....	11
QRadar configuration for accepting RSA Alerts .....	16
Install a single DSM .....	16
Add a Universal CEF log source on the QRadar console .....	16
Import RSA NetWitness QID map entries into QRadar .....	18
Configure event mapping for Universal CEF events .....	19
Add Custom Event Properties .....	22
<b>RSA Qlink for RSA NetWitness Suite .....</b>	<b>23</b>
Introduction .....	23
Deployment .....	23
Deploy to IBM QRadar Appliance .....	24
Deploy to Separate CentOS Instance .....	24
Configuration .....	24
Right-Click Properties File .....	25
Required Elements .....	25
Useful Elements .....	26
QLink Script Contents .....	27
QRadar Admin Console .....	27
License and Support for QLink Configuration .....	29
RSA NetWitness Right Click to QRadar .....	29
Right Click from QRadar to NetWitness .....	29
<b>Appendix: Source Code .....</b>	<b>32</b>
QLink Configuration Script Source Code .....	32
Right-Click Properties Source Code .....	34

# RSA NetWitness Suite Integration with IBM QRadar

---

IBM QRadar is an enterprise security information and event management (SIEM) product. It collects log data from an enterprise, its network devices, host assets and operating systems, applications, vulnerabilities, user activities and behaviors.

The IBM QRadar package is delivered as a ZIP archive that contains the context actions code. You download the file from RSA Link. You can find the package on the [RSA Link Downloads](#) space [here](#).

There are several NetWitness integrations with IBM QRadar. Currently, the following integration points are supported:

- [Forwarding Alerts: NetWitness to QRadar](#)
- RSA Qlink for NetWitness
- Right Click from NetWitness QRadar
- Right Click from QRadar to NetWitness

Additionally, the source is supplied in this document, in the [Appendix: Source Code](#) section.

## Forwarding Alerts: NetWitness to QRadar

---

NetWitness can forward Syslog CEF alerts to QRadar, both from Event Stream Analysis and the Reporting Engine . It is a two-step process.

1. NetWitness needs to configure the alerts in CEF format to be sent to QRadar via syslog.
  - Use the Event Stream Analysis CEF template to forward from ESA: [Event Stream Analysis CEF Template](#)
  - Use the Reporting Engine Alert CEF template to forward from the Reporting Engine: [Reporting Engine Alert Syslog CEF Template](#)
2. Set up QRadar so it can parse these logs coming in from NetWitness, rather than categorizing them as unknown.
  - a. Install a single DSM.
  - b. Add a Universal CEF log source on the QRadar console.
  - c. Import the RSA NetWitness Suite QID map entries into QRadar.
  - d. Configure event mapping for Universal CEF events.
  - e. Add custom event properties.

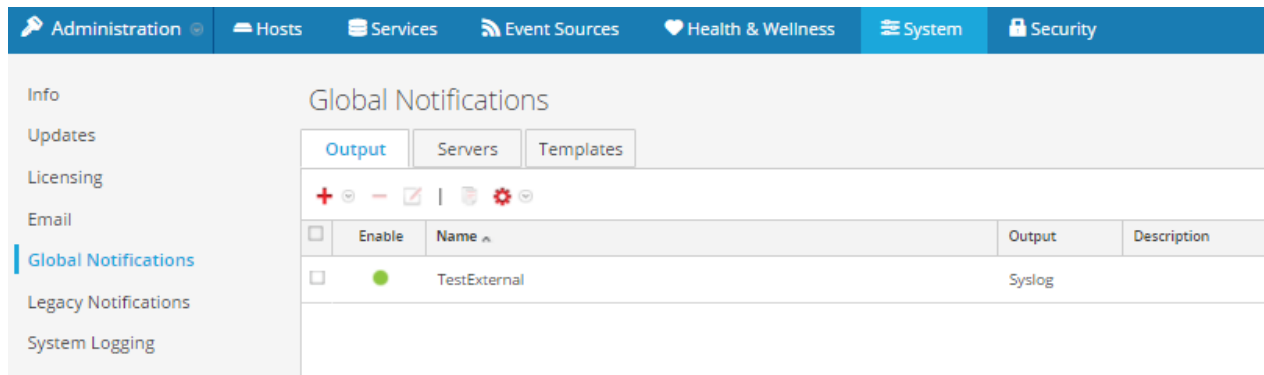
### Event Stream Analysis CEF Template

In this section, we add a global notification output, server, and template, then set up an Alert to use them.

#### To add the global output, server and template:

1. Log into the NetWitness GUI with Administrative Credentials
2. In the **NetWitness** menu, select **Administration > System**.
3. In the options panel, select **Global Notifications**.

The **Global Notifications** panel is displayed.



4. Add a Notification Output.

- a. From the **Output** tab, click  and choose Syslog from the drop-down menu.

The **Define Syslog Notification** dialog box is displayed.

- b. Fill in the screen as it appears here:

The 'Define Syslog Notification' dialog box is shown. It includes a descriptive paragraph about syslog. Below the text are several configuration fields:
 

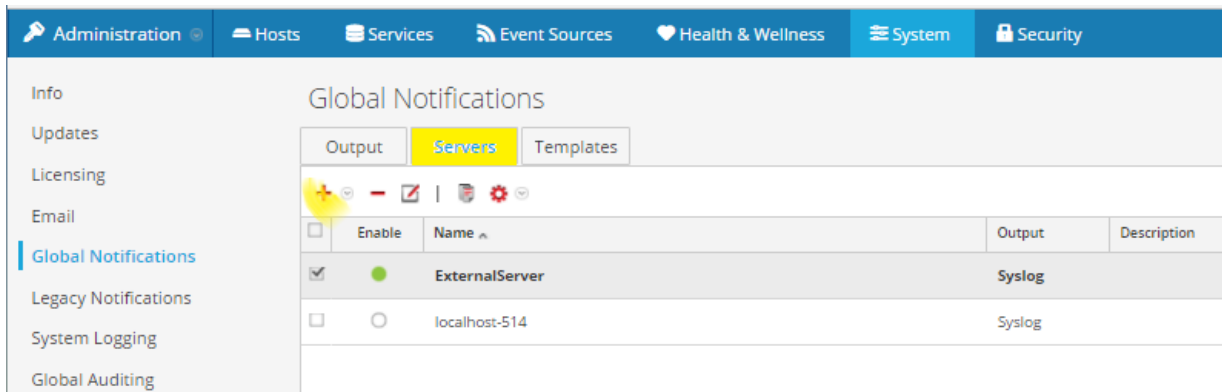
- Enable:** A checked checkbox.
- Name \*:** A text field containing 'TestExternal'.
- Description:** An empty text field.
- Severity:** A dropdown menu set to 'Alert'.
- Encoding:** A text field set to 'UTF-8'.
- Max Length:** A text field set to '10240', which is highlighted in yellow.
- Include Local Timestamp:** An unchecked checkbox.
- Include Local Hostname:** An unchecked checkbox.
- Identity String:** An empty text field.

 At the bottom right, there are 'Cancel' and 'Save' buttons.

- c. Click **Save**.

5. Add a Notification Server.

- a. Select the **Servers** tab, click  , and choose Syslog from the drop-down menu.



- b. Fill in the screen as it appears here:

### Define Syslog Notification Server

Provides auditing through the use of the RFC 5424 syslog protocol. Regulations, such as SOX, PCI DSS, HIPAA, and many others are requiring organizations to implement comprehensive security measures, which often include collecting and analyzing logs from many different sources. Syslog has proven to be an effective format to consolidate logs, as there are many open source and proprietary tools for reporting and analysis.

**Enable** ☒

**Name\***

**Description**

**Server IP Or Hostname\***

**Server Port**

**Protocol**

**Facility**

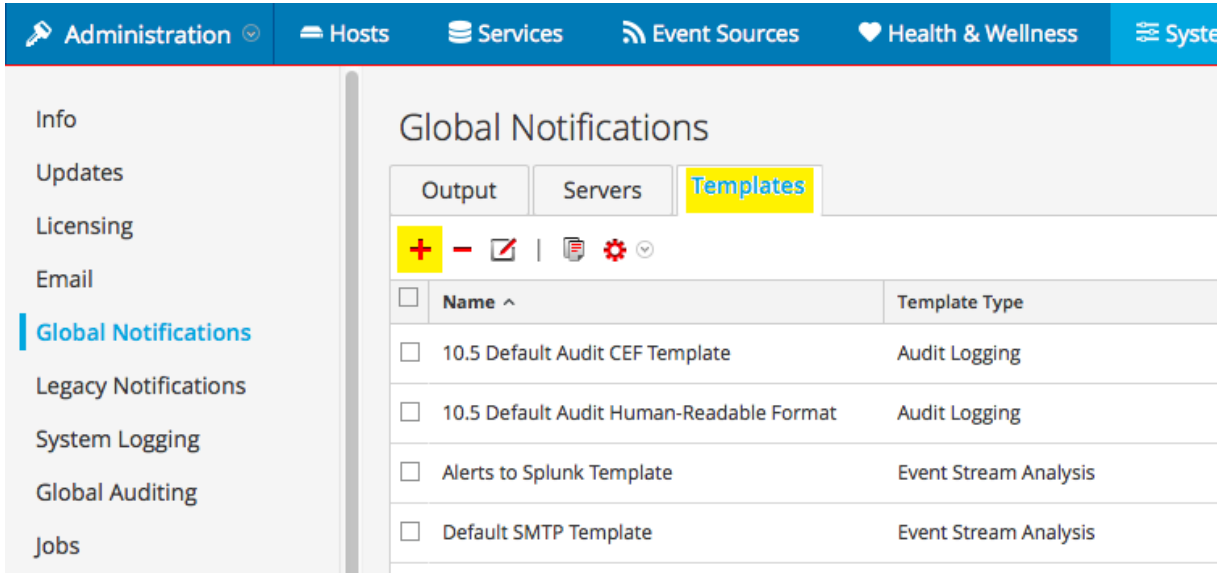
**Max Alerts Per Minute**

**Max Alert Wait Queue Size:**

**Note:** The Server IP address should be the IP Address of your QRadar.

- c. Click **Save**.
6. Add a Template.

- a. Select the **Templates** tab and click .



	Name ^	Template Type
<input type="checkbox"/>	10.5 Default Audit CEF Template	Audit Logging
<input type="checkbox"/>	10.5 Default Audit Human-Readable Format	Audit Logging
<input type="checkbox"/>	Alerts to Splunk Template	Event Stream Analysis
<input type="checkbox"/>	Default SMTP Template	Event Stream Analysis

The **Define Template** dialog box is displayed.

- b. Enter a **Name** (such as QRadar Template).
- c. From the **Template Type** drop-down menu, select **Event Stream Analysis**.
- d. Copy and paste in the text listed below (or copy it from the file that you downloaded from RSA Link) into the Template text field:



```
#include "macros.ftl"
<#list events as x>
CEF:0|RSA|NetWitness|10.6.3|${x.event_
type!""}|${moduleName}|${x.severity!""}|src=${x.ip_src!""} dst=${x.ip_dst!""} act="<#if
x.action?has_content><@value_of x.action /></#if>" app=${x.protocol!""}
destinationDnsDomain=${x.domain_dst!""} destinationServiceName="${x.client!""}"
dmac=${x.eth_dst!""} sntdom=${x.ad_domain_src!""} dproc="${x.process!""}" dpt=${x.tcp_
dstport!""} dst=${x.ip_dst!""} duid='${x.user_dst!""}' dvc=${x.device_ip!""}
dvchost=${x.device_host!""} endTime=${time?datetime} externalId=${x.rid!""}
fileType=${x.filetype!""} fileName="${x.filename!""}" msg="${x.event_desc!""}"
transportProtocol=${x.service!""} reason="${x.result_code!""}"
requestClientApplication="${x.user_agent!""}" requestMethod="<#if x.action?has_
content><@value_of x.action /></#if>" sourceHostName=${x.host_src!""} src=${x.ip_
src!""} smac=${x.eth_src!""} sourceDnsDomain=${x.domain_src!""} suid='${x.user_src!""}'
type=${x.medium!""} deviceCustomDate1=${x.event_time!""} deviceCustomDate1Label="Event
Time" cs2=${time?datetime?iso_m_nz("GMT+01") } cs2Label="Custom Time String plus 1 Hour"
cs1=${time?datetime?iso_m_nz("GMT-01") } cs1Label="Custom Time String minus 1 Hour"
cat="${x.event_cat_name!""}" spriv="${x.group!""}" cs3="${x.alert_id!""}"
cs3Label="Alert ID" cs4="${x.msg_id!""}" cs4Label="Message ID" cs5="${x.risk_info!""}-
${x.risk_suspicious!""}-${x.risk_warning!""}" cs5Label="Risk Categories"
cs6="${x.category!""}" cs6Label="NW Category" suser='${x.ad_username_src!""}'
deviceExternalId=${x.did!""} dhost=<#if x.alias_host?has_content><@value_of x.alias_
host/></#if> spt=${x.tcp_srcport!""} duser='${x.ad_username_dst!""}'
fileSize=${x.size!""} fileHash=${x.checksum!""} outcome="${x.ec_outcome!""}"
cnl=${x.sessionid!""} cnlLabel="SessionID" </#list>
```

Your dialog box should look like this:

Define Template

Name \*

QRadar Template

Template Type

Event Stream Analysis

Description

Template \*

```

<#include "macros.ftl">
<#list events as x>
CEF:0|RSA|NetWitness|10.6.3|${x.event_type!""}|${(moduleName)}|${x.severi
ty!""}|src=${x.ip_src!""}|dst=${x.ip_dst!""}|act="<#if x.action?has_content>
<@value_of x.action /></#if>" app=${x.protocol!""}
destinationDnsDomain=${x.domain_dst!""}
destinationServiceName="${x.client!""}" dmac=${x.eth_dst!""}
sntdom=${x.ad_domain_src!""} dproc="${x.process!""}"
dpt=${x.tcp_dstport!""}|dst=${x.ip_dst!""}|duid='${x.user_dst!""}'
dvc=${x.device_ip!""}|dvchost=${x.device_host!""}|endTime=${(time?datetime)}
externalId=${x.rid!""}|fileType=${x.filetype!""}|fileName="${x.filename!""}"
msg="${x.event_desc!""}" transportProtocol=${x.service!""}
reason="${x.result_code!""}" requestClientApplication="${x.user_agent!""}"
requestMethod="<#if x.action?has_content><@value_of x.action /></#if>"
sourceHostName=${x.host_src!""}|src=${x.ip_src!""}|smac=${x.eth_src!""}

```


Cancel

Save

e. Click **Save**.

Next, configure your ESA Alert to use the new template.

### To configure a rule to use the QRadar output, server, and template:

1. In the **NetWitness** menu, select **Alerts > Configure**.
2. From the Rule Library, select a rule and click the **Edit** icon (.

The Rule Builder screen appears.

3. In the Notifications section, click   > **Syslog** to configure Syslog notifications for the rule.

A notification row is added to the Notifications section.

4. Select the notification, notification server, and template that you created earlier.

Trial Rule

☒

Severity \*

Low

Conditions \*

+

-

✕

Investigation

	Statement	Occurs	Connector	Correlation Type	Meta	Meta
<input type="checkbox"/>	C1	1				

Group By

Occurs Within

minutes

Notifications

+

⊖

-

Global Notifications

	Output	Notification	Notification Server	Template
<input type="checkbox"/>	SYSLOG	TestExternal	ExternalServer	Qradar Template

☐ Output Suppression of every

minutes

Enrichments

+

⊖

-

Settings

	Output	Enrichment Source	ESA Event Stream Meta	Enrichment Source Column Name
No parameters to edit.				

Debug

☐

Save

Close

Show Syntax

\* = required field

5. Click **Save** to save your changes and close the Rule Builder screen.

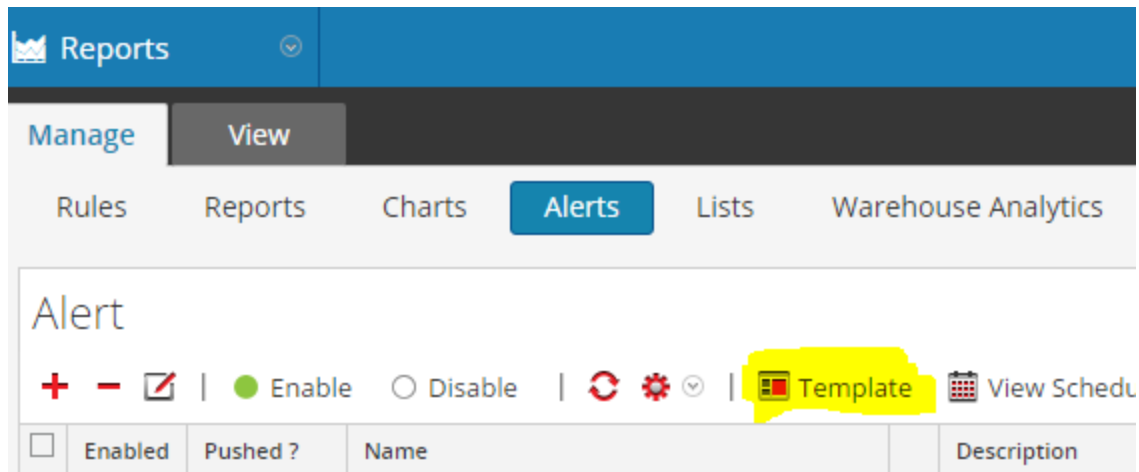
**Note:** Remember to re-deploy the rule to push the new notification templates

## Reporting Engine Alert Syslog CEF Template

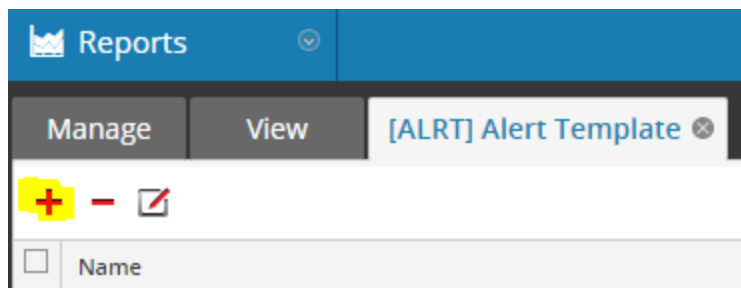
In this section, we add a Reporting Engine template, then set up an Alert to use it.

### To add a Reporting Engine template:

1. Log into the NetWitness GUI with Administrative Credentials
2. In the **NetWitness** menu, select **Reports > Alerts**.



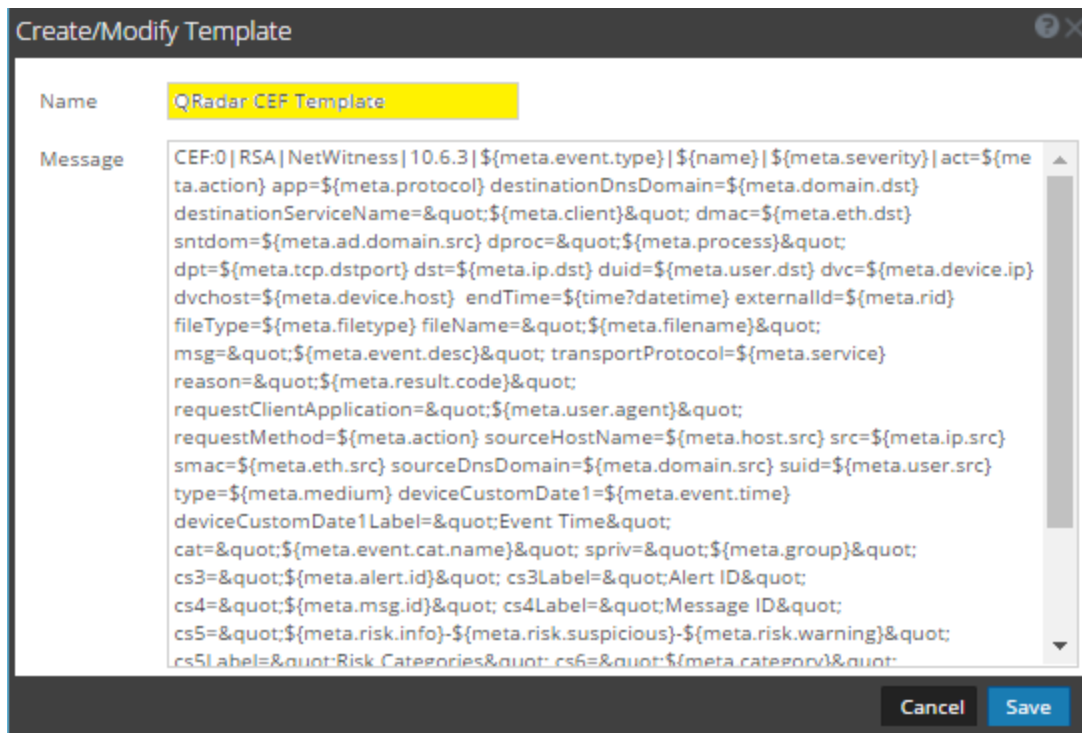
3. Select **Template** and click **+** to create a new template.



4. Name the template, for example **QRadar CEF Template**.
5. Copy and paste in the text listed below (or copy it from the file that you downloaded from RSA Link) into the Template text field:

```
CEF:0|RSA|NetWitness|10.6.3|${meta.event.type}|${name}|${meta.severity}|act=${meta.action}
app=${meta.protocol} destinationDnsDomain=${meta.domain.dst}
destinationServiceName=&quot;${meta.client}&quot; dmac=${meta.eth.dst}
sntdom=${meta.ad.domain.src} dproc=&quot;${meta.process}&quot; dpt=${meta.tcp.dstport}
dst=${meta.ip.dst} duid=${meta.user.dst} dvc=${meta.device.ip} dvchost=${meta.device.host}
endTime=${time?datetime} externalId=${meta.rid} fileType=${meta.filetype}
fileName=&quot;${meta.filename}&quot; msg=&quot;${meta.event.desc}&quot;
transportProtocol=${meta.service} reason=&quot;${meta.result.code}&quot;
requestClientApplication=&quot;${meta.user.agent}&quot; requestMethod=${meta.action}
sourceHostName=${meta.host.src} src=${meta.ip.src} smac=${meta.eth.src}
sourceDnsDomain=${meta.domain.src} suid=${meta.user.src} type=${meta.medium}
deviceCustomDate1=${meta.event.time} deviceCustomDate1Label=&quot;Event Time&quot;
cat=&quot;${meta.event.cat.name}&quot; spriv=&quot;${meta.group}&quot;
cs3=&quot;${meta.alert.id}&quot; cs3Label=&quot;Alert ID&quot;
cs4=&quot;${meta.msg.id}&quot; cs4Label=&quot;Message ID&quot;
cs5=&quot;${meta.risk.info}-${meta.risk.suspicious}-${meta.risk.warning}&quot;
cs5Label=&quot;Risk Categories&quot; cs6=&quot;${meta.category}&quot; cs6Label=&quot;NW
Category&quot; suser=${meta.ad.username.src} deviceExternalId=${meta.did}
dhost=${meta.alias.host} spt=${meta.tcp.srcport} duser=${meta.ad.username.dst}
fileSize=${meta.size} fileHash=${meta.checksum} outcome=&quot;${meta.ec.outcome}&quot;
cn1=${meta.sessionid} cn1Label=&quot;SessionID
```

Your dialog box should look like this:

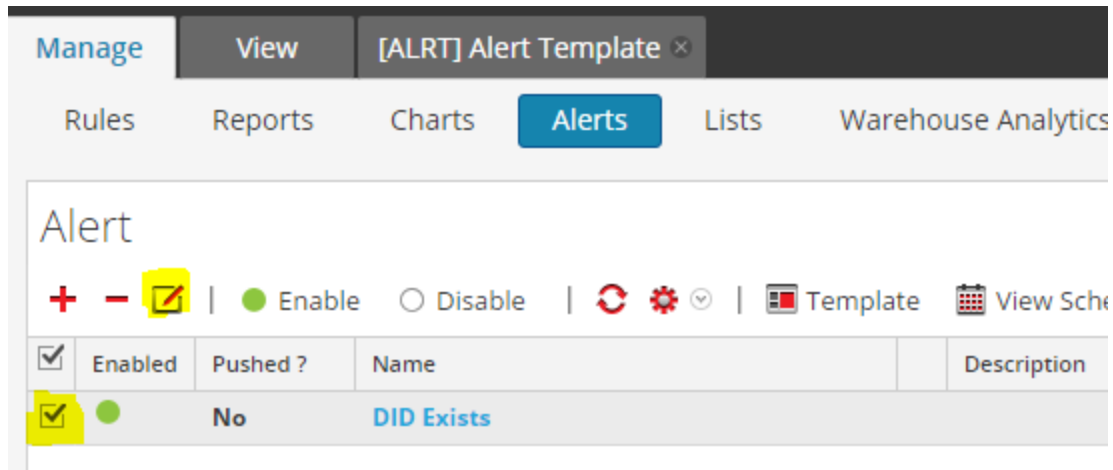


6. Click **Save**.

For each alert that you want to send to QRadar, you need to edit the Alert and configure it to send a Syslog Notification using the new template.

**To configure a rule to use the QRadar Reporting Engine template:**

1. In the **NetWitness** menu, select **Reports**.
2. From the Manage tab, click **Alerts**.



3. Select an alert and click the **Edit** icon (pencil icon).
4. From the Rule Library, select a rule and click the **Edit** icon (pencil icon).  
The Rule Builder screen appears.
5. In the Notification section, select **Syslog**.

## Create/Modify Alert

Enable ☒

Rule Basis:

Data Sources:  ☐ Push to decoders

Description

Severity:

Notification: ☒ Record ☐ SMTP ☐ SNMP ☒ Syslog

6. Click .

The New Syslog Configuration dialog box is displayed.

7. Enter parameters as follows.

- For **Syslog Configs**, select your Syslog Destination. This Destination was configured under **Global Notifications**, in the [Event Stream Analysis CEF Template](#) section.
- For **Body**, select the RE template you created earlier ([Create RE template step](#)).
- For the remainder of the fields, accept the default values.

### New Syslog Configuration

Syslog Configs:

Execute:

Facility:

Severity:

Body:

Body Template:

Repeat this procedure for each alert you want to export to QRadar.

## QRadar configuration for accepting RSA Alerts

Typically, the Universal CEF DSM components are installed by default and are part of IBM's regular update schedule for DSM updates when the **automatic updates** setting is enabled. If the Universal CEF components are not installed, you need to install them. Download and install the most recent version of the following RPMs on your QRadar Console.

### Install a single DSM

**Note:** The IBM support website contains individual DSMs that you can download and install using the command line.

1. Download the DSM file to your system hosting QRadar.
2. Using SSH, log onto QRadar as the root user (username is **root**).
3. Navigate to the directory that includes the downloaded file.
4. Type the following command:  

```
- rpm -Uvh <filename> i
```

where **<filename>** is the name of the downloaded file. For example:  

```
rpm -Uvh DSM-UniversalCEF-7.2-953671.noarch.rpm
```

or  

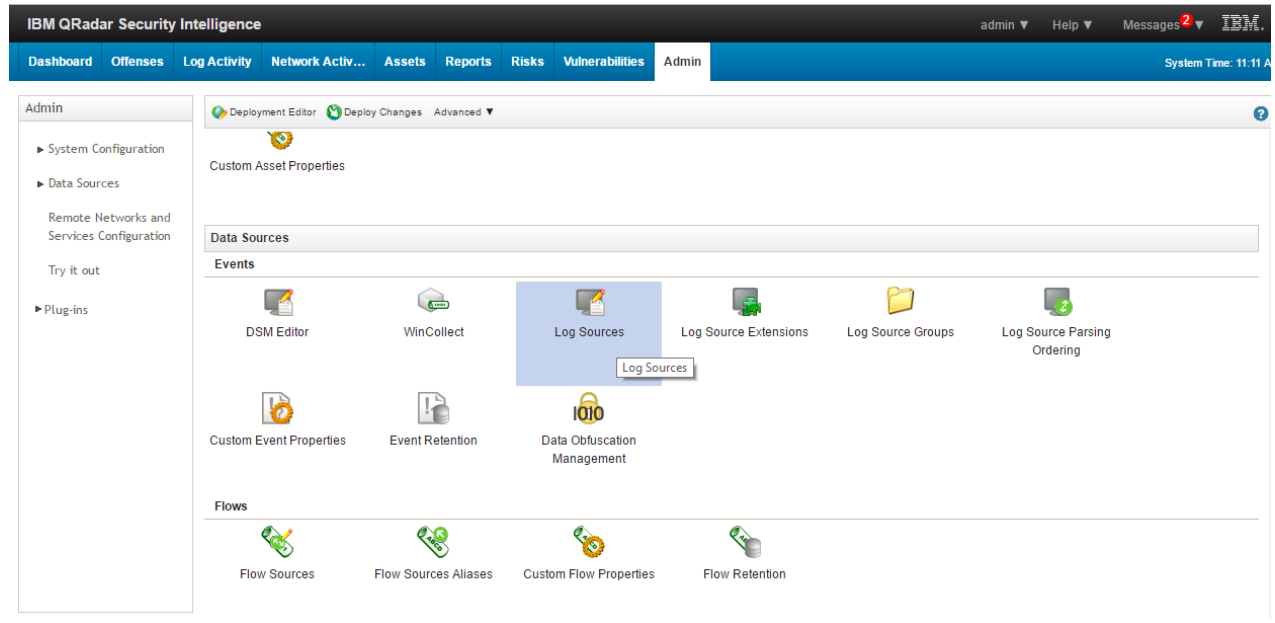
```
rpm -Uvh DSM-UniversalCEF-7.2-953671.noarch.rpm
```
5. Log in to QRadar: `https://<IP Address> i`, where **<IP Address>** is the IP address of the QRadar console or event collector.
6. On the **Admin** tab, click **Deploy Changes**.

This completes the installation.

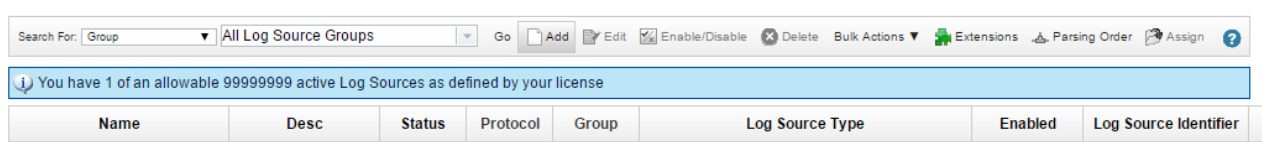
### Add a Universal CEF log source on the QRadar console

1. Log into the QRadar GUI with Administrative Credentials.
2. Navigate to the **Admin** tab, then click **Log Sources**.






3. Click **Add**.



4. Select log source as **Universal CEF** and fill in the details as rest as shown in the diagram.

 Note that the connection information for this log source is shared amongst one or more other log sources.

Log Source Name	<input type="text" value="RSA_Alert"/>
Log Source Description	<input type="text" value="RSA Alert"/>
Log Source Type	<input type="text" value="Universal CEF"/>
Protocol Configuration	<input type="text" value="Syslog"/>
Log Source Identifier	<input type="text" value="10.31.125.59"/>
Enabled	<input checked="" type="checkbox"/>
Credibility	<input type="text" value="10"/>
Target Event Collector	<input type="text" value="eventcollector0 :: qradar"/>
Coalescing Events	<input checked="" type="checkbox"/>
Incoming Payload Encoding	<input type="text" value="UTF-8"/>
Store Event Payload	<input checked="" type="checkbox"/>

Please select any groups you would like this log source to be a member of:

5. Click **Save**.
6. Click **Deploy Changes**.

## Import RSA NetWitness QID map entries into QRadar

1. Create a text file named **RSA\_QID.txt**.
2. Copy and paste the following entry into **RSA\_QID.txt**:  

```
,RSA NetWitness, ALERTS,3,8001
```
3. Using SSH, log in to QRadar SIEM as the root user (username is **root**).
4. Place the file on your system hosting QRadar in the following directory: `/opt`
5. Change to the following directory: `opt/qradar/bin`
6. To import the QID map file, execute the following command:  

```
./opt/qradar/bin/qidmap_cli.sh -i -f /opt/RSA_QID.txt
```

**Note:** If any of the entries in the file cause an error, none of the entries in the file are enforced.

The output from the command should look similar to the following (this sample output has been truncated from the actual output):

```
importing from file: /opt/RSA_QID.txt
importing new: ,RSA NetWitness,ALERTS,3,8001
Created entry:
  qid: 2000005
  name: RSA NetWitness
  description: ALERTS
  severity: 3
  low level category id: 8001
  ratethreshold: 0
  catpipename: Golf
  rateshortwindow: 0
  ratelongwindow: 0
  reverseip: false
  rateinterval: 0
,RSA NetWitness,ALERTS,3,8001

Invoking operation: forceNotification ( )
Result: true

Invoking operation: forceNotification ( )
Result: true
```

7. Restart the QRadar **hostcontext** service:

```
/etc/init.d/service hostcontext restart
```

QID map file importing is complete.

## Configure event mapping for Universal CEF events

**Note:** The following procedure and details are reproduced here from IBM Support site:  
[https://www.ibm.com/support/knowledgecenter/en/SS42VS\\_7.3.0/com.ibm.dsm.doc/t\\_dsm\\_guide\\_universal\\_cef\\_event\\_mapping.html](https://www.ibm.com/support/knowledgecenter/en/SS42VS_7.3.0/com.ibm.dsm.doc/t_dsm_guide_universal_cef_event_mapping.html)

By default, the Universal CEF DSM categorizes all events as unknown. All Universal CEF events display a value of unknown in the Event Name and Low-Level Category columns on the Log Activity tab.


Event Name	Log Source	Event Count	Time	Low Level Category	Source IP	Source Port	Destination IP	Destination Port	Username	Magnitude
Information Message	System Notification-2 :: qradar	1	Jun 15, 2017, 12:54:14 PM	Information	10.31.244.188	0	127.0.0.1	0	N/A	
Information Message	System Notification-2 :: qradar	1	Jun 15, 2017, 12:54:14 PM	Information	10.31.244.188	0	127.0.0.1	0	N/A	
Information Message	System Notification-2 :: qradar	1	Jun 15, 2017, 12:54:14 PM	Information	10.31.244.188	0	127.0.0.1	0	N/A	
Information Message	System Notification-2 :: qradar	1	Jun 15, 2017, 12:54:13 PM	Information	10.31.244.188	0	127.0.0.1	0	N/A	
Unknown CEF Event	RSA_Alert	3	Jun 15, 2017, 12:54:05 PM	Unknown	9.9.9.1	0	20.20.20.2	0		

You must modify the QID map to individually map each event for your device to an event category in QRadar. Mapping events allows QRadar to identify, coalesce and track events from your network devices.

1. Log in to QRadar.
2. Click the **Log Activity** tab.
3. Click **Add Filter**.
4. From the first list, select **Log Source**.
5. From the **Log Source Group** list, select **Other**.
6. From the **Log Source** list, select your Universal CEF log source.
7. Click **Add Filter**.
8. From the **View** list, select **Last Hour**.
9. Optional: Click **Save Criteria** to save your existing search filter.
10. On the **Event Name** column, double-click an unknown event for your Universal CEF DSM.
11. Scroll to the Payload information at the bottom and highlight and copy the name of the event contained in the sixth field of the CEF header.

```
<9>CEF:0|RSA|NetWitness|10.6.3||Multi Service Connection Attempts Log|src=9.9.9.1 dst=20.20.20.2 act="fw:inbound-network-traffic" app= destinationDnsDomain=
destinationServiceName="" dmac= sntdom= dproc="" dpt= dst=20.20.20.2 duid="" dvc=127.0.0.1 dvchost= endTime=2017-06-15T13:12Z externalId=61084 fileType=
fileName="" msg="connection denied" transportProtocol= reason="" requestClientApplication="" requestMethod="fw:inbound-network-traffic" sourceHostName=
src=9.9.9.1 smac= sourceDnsDomain= suid="" type=32 deviceCustomDate1= deviceCustomDate1Label="Event Time" cs2=2017-06-15T14:12 cs2Label="Custom Time String
plus 1 Hour" cs1=2017-06-15T12:12 cs1Label="Custom Time String minus 1 Hour" cat="Network.Denied Connections" spriv="" cs3="" cs3Label="Alert ID" cs4="106001"
cs4Label="Message ID" cs5="" cs5Label="Risk Categories" cs6="" cs6Label="NW Category" user="" deviceExternalId=nwappliance28448 dhost= spt= duser=""
fileSize=154 fileHash= outcome="" cn1=61084 cn1Label="SessionID" CEF:0|RSA|NetWitness|10.6.3|
```

12. Click **Map Event**.
13. From the Browse for QID pane, paste the event name in the **QID/Name** search box.


**Log Source Event**

Log Source Type: UniversalCEF

Log Source Event Category: UniversalCEF

Log Source Event ID: Multi Service Connection Attempts Log

Original QID: Unknown

*If you know the QID to associate this event to, enter it here*

Enter QIDs:

*Or browse for the desired QID below*

Browse for QID

High-Level Category: Any ▼

Low-Level Category: Any ▼

Log Source Type: Universal CEF ▼

QID/Name:

**Matching QIDs**

QID	Name ▲	Description	Severity
2000002	RSA Netwitness	ALERTS	3
2000005	RSA NetWitness	ALERTS	3
93750001	UniversalCEF Mes...	UniversalCEF Stored Event	3
93750003	unknown	unknown	3
93750002	Unknown CEF Event	An unknown CEF event was e...	3

14. Click **Search**.
  15. Select the QID that you want to associate to your unknown Universal CEF DSM event and click **OK**.
  16. Do this for all event types that come as Unknown from RSA NetWitness Suite.
- Event mapping is complete.

## Add Custom Event Properties

If you need to extract information from the CEF that is not available in the standard set of properties, you can add custom event properties.

Navigate to Admin > Custom Event Properties to configure custom properties.

You can use this method to extract any key=value pair within the CEF syslog message.

Add Edit Copy Enable/Disable Delete

rsa

Property Name	Type	Property Description	Log Source Type	Log Source	Event Name	Category	Expression	Username	Enabled
RSA NetWitness	Regex	For extraction...	Universal CEF	RSA_Alert	RSA NetWitn...	N/A	msg=("(.*?)"	admin	True

IBM QRadar Security Intelligence

admin Help Messages 3

Dashboard Offenses Log Activity Network Activ... Assets Reports Risks Vulnerabilities Admin

Return to Event List Offense Map Event False Positive Extract Property Previous Next Print Obfuscation

Extract a custom property from the payload of this event

Event Information

Event Name	RSA NetWitness										
Low Level Category	Unknown System Event										
Event Description	ALERTS										
Magnitude	<div></div> (4)			Relevance	3			Severity	3	Credibility	10
Username	"										
Start Time	Jun 15, 2017, 2:12:33 PM			Storage Time	Jun 15, 2017, 2:12:33 PM			Log Source Time	Jun 15, 2017, 2:12:43 PM		
RSA NetWitness (custom)	"connection denied"										
Domain	Default Domain										

Source and Destination Information

Source IP	9.9.9.1	Destination IP	20.20.20.2
Source Asset Name	N/A	Destination Asset Name	N/A
Source Port	0	Destination Port	0
Pre NAT Source IP		Pre NAT Destination IP	
Pre NAT Source Port	0	Pre NAT Destination Port	0

## RSA Qlink for RSA NetWitness Suite

---

### Introduction

The RSA NetWitness Suite packet capture platform is frequently deployed alongside the IBM QRadar SIEM product to augment traditional Log or Flow event capture with deep visibility and RSA's market leading Network Forensics investigation capabilities.

QLink is a PHP script. Analysts can use it to pivot from Log or Flow events in QRadar to the RSA NetWitness Suite Investigation screen, using source IP or destination IP data from a QRadar event as the starting drill-point into the RSA NetWitness Suite packet capture dataset. It is possible to configure basic right-click actions without QLink; however, QLink enables focused timeQbased searches of the RSA NetWitness Suite dataset, as opposed to broad, IPQ-only searches. QLink also addresses the following date-time challenges:

- QRadar outputs date-time variables in an epoch format, for example **1448281678752**. NetWitness uses human-readable date-time format for inbound deep links, for example **2015Q11Q23T12:27Z**. Thus, the QRadar date-time format must be converted.
- QRadar appliances can be configured to run in a local time zone, whereas RSA NetWitness Suite is always configured to use UTC with the administration interface applying a local timezone offset. QRadar date-time values must be adjusted to UTC, or else RSA NetWitness Suite is queried for the wrong time period.
- QRadar date-time epoch values have millisecond resolution while the minimum time period for queries to RSA NetWitness Suite is one minute. Thus, QRadar date-time range values must have second and millisecond values removed. The perceived loss of event time resolution is normally negated by the following point.
- QRadar provides start (event) time for Log events and a start to end time range for Flow events. However, analysts normally prefer to search RSA NetWitness Suite data with a broader time range, so as to include additional activity and context around the event. To avoid repetitive clicking in the RSA NetWitness Suite console to change time range values, QLink allows right-click actions that enlarge the range in preconfigured increments.

After date-time corrections are applied, QLink automatically redirects the analyst's browser to RSA NetWitness Suite, making the right-click pivot transparent.

### Deployment

You can deploy QLink to either your IBM QRadar appliance or a separate CentOS instance.

## Deploy to IBM QRadar Appliance

The QLink script should be installed to `/opt/qradar/www/qlink.php` on the IBM QRadar appliance. The support status with IBM for this on-box location is not fully known, but other integration and utility scripts provided by IBM in the default QRadar installation image reside here.

## Deploy to Separate CentOS Instance

Alternatively, it is a simple task to create a CentOS instance from a minimal ISO. You will need to install the **httpd** and **php** rpm packages, configure httpd to start automatically on boot, and configure the iptables firewall to permit inbound connections to the web server. Once you complete these steps, install QLink be to `/var/httpd/www`, then start the httpd service.

If you deploy QLink to a separate host, make sure to consider the sensitivity of the URL queries submitted to RSA NetWitness Suite: RSA encourages you to use SSL configuration to ensure queries are submitted from inside a secure session.

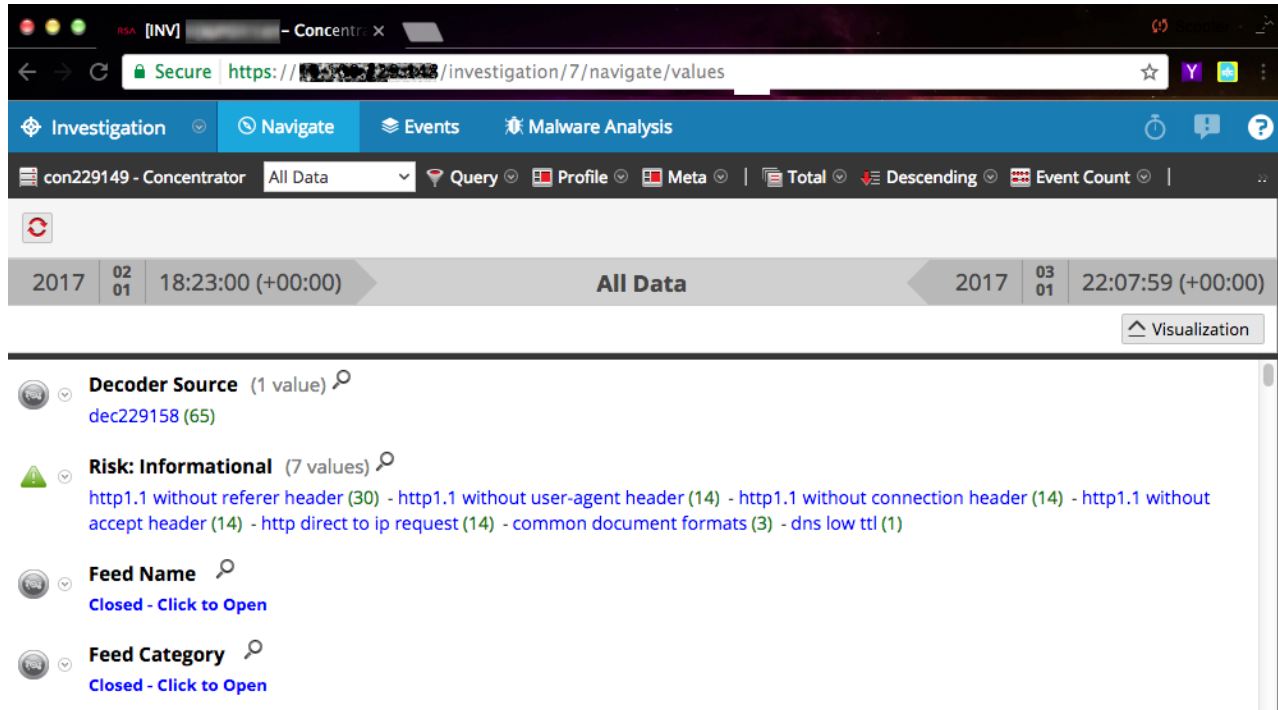
Please remember to check the `/etc/httpd/conf/httpd.conf` file. Place the `qlink.php` file in the folder mentioned under the **General Setup > DocumentRoot**.

## Configuration

Once QLink has been deployed, three configuration items need to be set in the header of the script, and one in the footer of the script. The script has a number of clearly written comments to help you find these locations:

- **Header:** The FQDN or IP address of the RSA NetWitness Suite server the script redirects to after transposing the deepQlink URL format from QRadar.
- **Header:** The node ID of the RSA NetWitness Suite appliance in the GUI. This can be deduced from the Investigation screen. In the screenshot below, the ID value “7” can be seen in the URL bar.





- **Header:** The QRadar appliance time zone in PHP time zone format, as shown on the following website: <http://php.net/manual/en/timezones.php>. QLink uses Etc/UTC for the RSA NetWitness Suite server time zone—this should not be changed.
- **Footer:** The script outputs text in a browser to show you the inbound and converted URLs during testing. To place the script into **autoQredirect** mode, you must comment out the echo statements that generate text, and uncomment the redirection function at the bottom of the script. The `/opt/qradar/conf/arielRightClick.properties` file holds configuration for QRadar right-click actions. It can be accessed via the QRadar SSH console. You can edit this file with a text editor (for example `vi`). In most installations, this file already exists. The example below shows default content.

## Right-Click Properties File

The default content of the `arielRightClick.properties` file is available in the Appendix: [Right-Click Properties Source Code](#).

**Note:** Comment lines begin with the pound sign (#).

## Required Elements

The following elements must be included in the QLink URL:

- **query**="**<ip.src=\$sourceIP\$>**". This is the query that you want to run against the RSA NetWitness Suite dataset. The query must be enclosed in double quotes. Queries with multiple statements or logic that requires the use of spaces or complex characters such as **%&()** must be URL Encoded (see [http://www.w3schools.com/tags/ref\\_urlencode.asp](http://www.w3schools.com/tags/ref_urlencode.asp)) to ensure correct passing of the query in the URL. For example:

`ip.src=192.168.10.2 && ip.dst=65.14.23.12` would be configured as:

`query="ip.src=$sourceIP$%20%26%26%20ip.dst=$destinationIP$"`

- **stime**=\$variable\$. This determines the start of the time range for the RSA NetWitness Suite search query. The **\$startTime** Ariel SDK variable is used for Log events, and **\$firstPacketTime** is used for Flow events.
- **etime**=\$variable\$. This determines the end of the time range for the RSA NetWitness Suite search query. The **\$endTime** Ariel SDK variable is used for Log events, and **\$lastPacketTime** is used for Flow events. In Log events, **\$endTime** contains the same value as **\$startTime**, whereas in Flow events, the values are normally different, specifying a true time range.
- **timeRange**="value". This is the time padding value used to expand the searched date-time range in the RSA NetWitness Suite data set. By default, the range is 60 minutes.

## Useful Elements

The following are useful elements of the the QRadar `arielRightClick.properties` file:

- **pluginActions=XXXX**. Each action must have a unique **actionName**, for example **XFE\_URL\_Lookup** as shown in the default example above. **pluginActions** must reference all configured **rightQclick** **actionName** values on a single line. The **actionName** is free text, but when applying many similar actions to the file, we suggest short, simple names with a logical structure to avoid confusion. You can add comment lines to keep track of actions.
- **actionName.useFormattedValue=false**. Some online examples use this command.
- **actionName.arielProperty=sourceIP**. This configures **actionName** as an action when you right click on **sourceIP** events in the admin console. The QRadar console uses the same SDK variable in both Log and Flow events. Thus **sourceIP** **rightQclick** actions created for Log events are be visible when you right click **sourceIP** on Flow events; use text naming to differentiate the actions clearly. QRadar appears to support **rightQclick** actions on **sourceIP**, **destinationIP**, **sourcePort** and **destinationPort** only.
- **actionName.text=Investigate Event in RSA (ip.src)**. This sets the text description for the event in the admin console. RSA suggests simple, short statements for clarity.

- **actionName.url=https://qradar/qmlink.php?query=XXXX**. This is the specially formatted URL used for the action. The URL points to the FQDN or IP of the QRadar admin console (or external web server) and path to the **qmlink.php** script.

## QLink Script Contents

The contents of the script are listed in the Appendix: [QLink Configuration Script Source Code](#).

## QRadar Admin Console

The following are screen shots show how the integration appears in the QRadar admin console.

For Source IP and Destination IP right-click actions, the previous file provides options for investigating in NetWitness UI for the following:

- ip.src
- ip.dst
- ip.addr
- username: Since there is no right click on username by QRadar, this is a workaround. Provided the logs do have username, if not, null.

The following image shows **Right Click on Source IP**.

The screenshot displays the IBM QRadar Security Intelligence Admin Console interface. The top navigation bar includes tabs for Dashboard, Offenses, Log Activity, Network Activ..., Assets, Reports, Risks, Vulnerabilities, and Admin. The main content area shows a table of events with columns: Event Name, Log Source, Event Count, Time, Low Level Category, Source IP, Source Port, Destination IP, Destination Port, Username, and Magnitude. A right-click context menu is open over the 'Source IP' column, showing options such as 'Filter on Source IP is 1.1.1.1', 'Filter on Source IP is not 1.1.1.1', 'Filter on Source or Destination IP is 1.1.1.1', 'Quick Filter...', 'False Positive', 'View path from 1.1.1.1 to 10.31.125.59', 'View in DSM Editor', 'More Options...', and 'Plugin options...'. The 'Plugin options...' option is highlighted, and a sub-menu is visible with options: 'Investigate in NetWitness (ip.src)', 'Investigate in NetWitness (ip.dst)', 'Investigate in NetWitness (ip.addr)', and 'Investigate in NetWitness (username)'.

The following image shows **Right Click on Destination IP**.

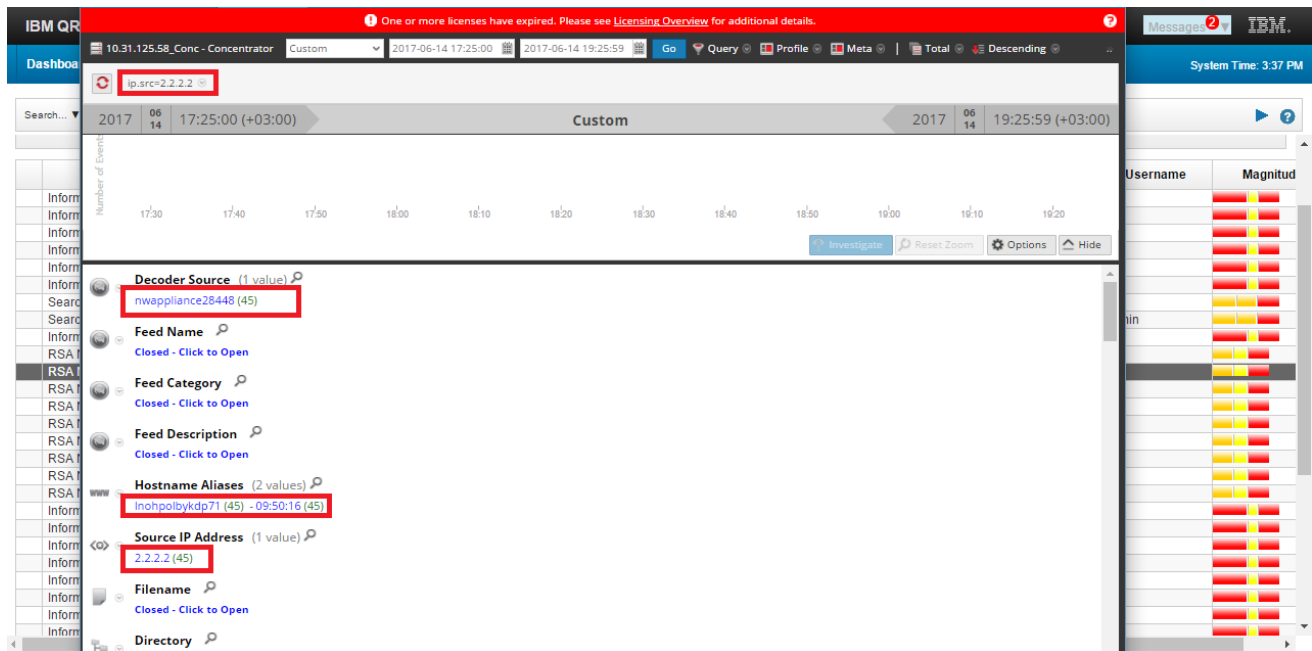
## IBM QRadar Integration

The screenshot shows the IBM QRadar Security Intelligence interface. The top navigation bar includes 'Dashboard', 'Offenses', 'Log Activity', 'Network Activ...', 'Assets', 'Reports', 'Risks', 'Vulnerabilities', and 'Admin'. The 'Log Activity' tab is selected. Below the navigation bar, there is a search bar and a table of events. The table has columns: Event Name, Log Source, Event Count, Time, Low Level Category, Source IP, Source Port, Destination IP, Destination Port, Username, and Magnitude. A context menu is open over the table, showing options like 'Filter on Destination IP is 10.31.125.59', 'Quick Filter...', 'False Positive', 'View path from 2.2.2.2 to 10.31.125.59', 'View in DSM Editor', 'More Options...', and 'Plugin options...'. The 'Plugin options...' option is highlighted, showing a list of plugins including 'Investigate in NetWitness (ip.src)', 'Investigate in NetWitness (ip.dst)', 'Investigate in NetWitness (ip.addr)', and 'Investigate in NetWitness (username)'.

The following image shows selecting “Investigate in NetWitness (ip.src)” -1.

The screenshot shows the IBM QRadar Security Intelligence interface. The top navigation bar includes 'Dashboard', 'Offenses', 'Log Activity', 'Network Activ...', 'Assets', 'Reports', 'Risks', 'Vulnerabilities', and 'Admin'. The 'Log Activity' tab is selected. Below the navigation bar, there is a search bar and a table of events. The table has columns: Event Name, Log Source, Event Count, Time, Low Level Category, Source IP, Source Port, Destination IP, Destination Port, Username, and Magnitude. A context menu is open over the table, showing options like 'Filter on Source IP is 2.2.2.2', 'Filter on Source IP is not 2.2.2.2', 'Filter on Source or Destination IP is 2.2.2.2', 'Quick Filter...', 'False Positive', 'View path from 2.2.2.2 to 10.31.125.59', 'View in DSM Editor', 'More Options...', and 'Plugin options...'. The 'Plugin options...' option is highlighted, showing a list of plugins including 'Investigate in NetWitness (ip.src)', 'Investigate in NetWitness (ip.dst)', 'Investigate in NetWitness (ip.addr)', and 'Investigate in NetWitness (username)'.

The following image shows selecting “Investigate in NetWitness (ip.src)” -2.



For changes to take effect, you must run the service tomcat restart command. You can run this command through the Command-Line Interface (CLI) or by using the QRadar UI: **Admin > Advance > Restart Web Server**.

## License and Support for QLink Configuration

The QLink script is not shipped with any form of license; you are simply free to use or modify it in any way to suit your needs. It is provided with no warranty or implied fitness for purpose, and is not supported by RSA Technical Support or RSA Professional Services. If you have questions on QLink, please post them to the RSA NetWitness Suite community forum where other QLink users may be able to offer assistance. The forum is located here: <https://community.rsa.com/community/products/netwitness>

## RSA NetWitness Right Click to QRadar

The integration where an analyst pivots from a meta field (for example IP address) in the RSA NetWitness Suite Investigations console to IBM QRadar is not currently possible. RSA fully supports right-click actions being added to any meta element, but IBM does not expose a deep-linking URL structure to facilitate deep-link integrations by the means of right clicking and drilling into QRadar.

## Right Click from QRadar to NetWitness

QRadar can also be configured for right-click integration with RSA NetWitness Suite outside of the QLink script. You can create a right-click action by following the detailed documentation for supported versions on the IBM Website at this link:

[http://www.ibm.com/support/knowledgecenter/SS42VS\\_7.2.6/com.ibm.qradar.doc/t\\_CUSTOMIZING\\_THE\\_RIGHT\\_CLICK\\_MENU.html](http://www.ibm.com/support/knowledgecenter/SS42VS_7.2.6/com.ibm.qradar.doc/t_CUSTOMIZING_THE_RIGHT_CLICK_MENU.html)

This can only be done on IP-based searches and do not have the time bound feature.

The following images show how the context menu behaves.

The top screenshot shows the IBM QRadar Security Intelligence interface. The search results table displays various events, including 'Information Message', 'Anomaly Detection Engine', 'RSA NetWitness', and 'Health Metric'. A right-click context menu is open over a row, showing options such as 'Filter on Source IP is 15.15.15.15', 'Filter on Source IP is not 15.15.15.15', 'Filter on Source or Destination IP is 15.15.15.15', 'Quick Filter...', 'False Positive', 'View path from 15.15.15.15 to 10.31.125.59', 'View in DSM Editor', 'More Options...', and 'Plugin options...'. The bottom screenshot shows the 'All Data' view of a search for 'ip.src=15.15.15.15'. The left pane displays a detailed view of the selected event, including fields like 'Decoder Source', 'Feed Name', 'Feed Category', 'Feed Description', 'Hostname Aliases', and 'Source IP Address'. The right pane shows the search results table.

To get this behavior:

1. Open `/opt/qradar/conf/ip_context_menu.xml` in a text editor (such as vi).
2. Add the following line within the `<contextMenu></contextMenu>` element:

```
<menuEntry name="NetWitness IP lookup (ip.src)"
url="https://<Ip_of_SA>/investigation/<DeviceID>/navigate/query/ip.src+%3D+%IP%"/>
```

3. Save the file and exit from your text editor.
4. Restart the tomcat service.

Additionally, you can also use the Threat Analytics Plugin for Google Chrome. The link to configure the Threat Analytics Plugin is located here: <https://inside.dell.com/docs/DOC-182438>.

## Appendix: Source Code

---

The IBM QRadar package is delivered as a ZIP archive that contains the context actions code. You download the file from RSA Link. You can find the package on the [RSA Link Downloads](#) space [here](#).

The code is to be used in QRadar system: add the code files to the desired folder and follow the steps mentioned below.

### QLink Configuration Script Source Code

The file path is `/opt/qradar/www/qlink.php`

```
<?php

////////////////////////////////////
//
// QLINK CONFIGURATION:
//
////////////////////////////////////
//
// set the NetWitness FQDN or IP address:
$saHost = "<IP of SA>";
//
// set the NetWitness device ID (crib from SA 'Investigation' URLs)
$saDevice = "<Go to the investigation page, use the number. For example
https://10.31.125.60/investigation/13/navigate use 13>";
//
// configure QRadar timezone, e.g. http://php.net/manual/en/timezones.php
$dateTimeZoneIBM = new DateTimeZone('<check the timezone info from above and then
recheck using SSH into the QRadar box>');
//
////////////////////////////////////
//
// DO NOT TOUCH BELOW HERE!
//
////////////////////////////////////
//
// set NetWitness timezone to UTC by default, ONLY change if it is NOT in UTC
$dateTimeZoneRSA = new DateTimeZone('Etc/UTC');

// calculate time in local timezones
```



```
$dateTimeRSA = new DateTime("now", $dateTimeZoneRSA);
$dateTimeIBM = new DateTime("now", $dateTimeZoneIBM);

// calculate the timezone offset
$offset = $dateTimeZoneIBM->getOffset($dateTimeRSA) - $dateTimeZoneRSA->getOffset($dateTimeRSA);

// parse value pairs from the URL
$ibm="https://".$_SERVER['HTTP_HOST'].$_SERVER['REQUEST_URI'];
$ibm_parts = parse_url($ibm);
$ibm_query_parts = parse_str($ibm_parts['query']);

// remove quotes from query, stime, etime, fuzz
$query = str_replace("\"", "", $query);
$stime = str_replace("\"", "", $stime);
$etime = str_replace("\"", "", $etime);
$timeRange = str_replace("\"", "", $timeRange);

// if no end time, use start time
if ($etime == 0) {
    $etime = $stime;
}

// trim seconds from ibm epoch times
$stime = substr($stime, 0, -3);
$etime = substr($etime, 0, -3);

// logic to determine what timezone offset maths to use
if ($offset > 0) {
    // IBM timezone is ahead of RSA
    $stime = $stime - $offset - $timeRange;
    $etime = $etime - $offset + $timeRange;
} elseif (0 > $time_offset) {
    // RSA timezone is ahead of IBM
    $stime = $stime + $offset - $timeRange;
    $etime = $etime + $offset + $timeRange;
} else {
```

```
// no timezone difference";
$stime = $stime - $timeRange;
$etime = $etime + $timeRange;
}

// output stime/etime in rsa format
$stime = date('Y-m-d H:i', $stime);
$stime = str_replace(" ", "T", $stime);
$stime = $stime .="Z";

$etime = date('Y-m-d H:i', $etime);
$etime = str_replace(" ", "T", $etime);
$etime = $etime .="Z";

// build NetWitness URL
$rsa = "https://" . $sahost . "/investigation/" . $sadevice . "/navigate/query/" .
$query . "/date/" . $stime . "/" . $etime;

////////////////////////////////////
//
// CONFIGURE TEXT-MODE/REDIRECTION-MODE BELOW HERE:
//
////////////////////////////////////

// uncomment for debugging
//echo "query = " . $query . "<p>";
//echo "ibm = " . $ibm . "<p>";
//echo "rsa = " . $rsa;

// uncomment to redirect browser after transformation
header("Location: $rsa");
die();
?>
```

## Right-Click Properties Source Code

The file path is `/opt/qradar/conf/arielRightClick.properties`

```
pluginActions=nwIPsrc0,nwIPsrc1,nwIPdst0,nwIPdst1,nwIPaddr0,nwIPaddr1,nwUsr,XFE_URL_Lookup
```

```
#For IP search for ip.src from SourceIP
nwIPsrc0.useFormattedValue=false
nwIPsrc0.arielProperty=sourceIP
nwIPsrc0.text=Investigate in NetWitness (ip.src)
nwIPsrc0.url=https://10.31.244.188/qlink.php?query="ip.src=$sourceIP$"&stime="$startTime$"&etime="$endTime$"&timeRange="3600"

# For IP search for ip.src from destinationIP
nwIPsrc1.useFormattedValue=false
nwIPsrc1.arielProperty=destinationIP
nwIPsrc1.text=Investigate in NetWitness (ip.src)
nwIPsrc1.url=https://10.31.244.188/qlink.php?query="ip.src=$destinationIP$"&stime="$startTime$"&etime="$endTime$"&timeRange="3600"

#For IP search for ip.dst from SourceIP
nwIPdst0.useFormattedValue=false
nwIPdst0.arielProperty=sourceIP
nwIPdst0.text=Investigate in NetWitness (ip.dst)
nwIPdst0.url=https://10.31.244.188/qlink.php?query="ip.dst=$sourceIP$"&stime="$startTime$"&etime="$endTime$"&timeRange="3600"

# For IP search for ip.dst from destinationIP
nwIPdst1.useFormattedValue=false
nwIPdst1.arielProperty=destinationIP
nwIPdst1.text=Investigate in NetWitness (ip.dst)
nwIPdst1.url=https://10.31.244.188/qlink.php?query="ip.dst=$destinationIP$"&stime="$startTime$"&etime="$endTime$"&timeRange="3600"

#For IP search for ip.addr from SourceIP
nwIPaddr0.useFormattedValue=false
nwIPaddr0.arielProperty=sourceIP
nwIPaddr0.text=Investigate in NetWitness (ip.addr)
nwIPaddr0.url=https://10.31.244.188/qlink.php?query="ip.addr=$sourceIP$"&stime="$startTime$"&etime="$endTime$"&timeRange="3600"

# For IP search for ip.addr from destinationIP
nwIPaddr1.useFormattedValue=false
nwIPaddr1.arielProperty=destinationIP
```

```

nwIPAddr1.text=Investigate in NetWitness (ip.addr)
nwIPAddr1.url=https://10.31.244.188/qlink.php?query="ip.addr=$destinationIP$"&stime="$startTime$&etime="$endTime$"&timeRange="3600"

#For search on username
nwUsr.useFormattedValue=false
nwUsr.arielProperty=sourceIP,destinationIP
nwUsr.text=Investigate in NetWitness (username)
nwUsr.url=https://10.31.244.188/qlink.php?query="username=$userName$"&stime="$startTime$"&etime="$endTime$"&timeRange="3600"

# begin XFE integration
XFE_URL_Lookup.arielProperty=URL
XFE_URL_Lookup.text=X-Force Exchange Lookup
XFE_URL_Lookup.url=https://exchange.xforce.ibmcloud.com/#/url/$URL$
# end XFE integration

# Flow / destination + source / fuzz=1800
dsF1800.useFormattedValue=false
dsF1800.arielProperty=destinationIP
dsF1800.text=Investigate Flow in RSA (ip.dst & ip.src) +/Q 30 mins
dsF1800.url=https://192.168.1.181/qlink.php?query="ip.dst=$destinationIP$%20%26%26%20ip.src=$sourceIP$"&stime="$firstPacketTime$&etime="$lastPacketTime$"&fuzz="1800"

# begin XFE integration
XFE_URL_Lookup.arielProperty=URL
XFE_URL_Lookup.text=XQForce Exchange Lookup
XFE_URL_Lookup.url=https://exchange.xforce.ibmcloud.com/#/url/$URL$
# end XFE integration

```