RSA® NETWITNESS®

Security Operations Implementation Guide

Gurucul Risk Analytics

Jeffrey Carlson, RSA Partner Engineering

Last Modified: June 27th, 2017

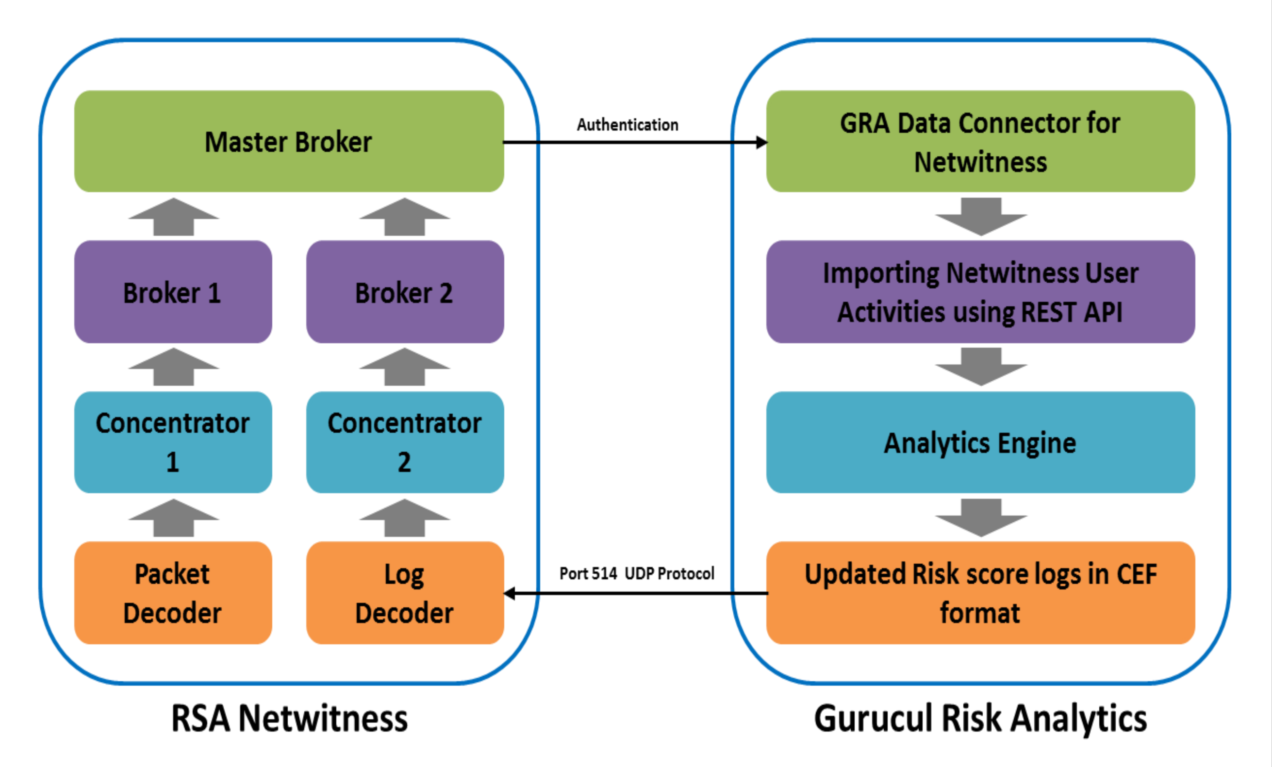
Gurucul Risk Analytics (GRA) Overview

Gurucul is changing the way enterprises protect themselves against fraud, insider threats and external intruders on premise and in the cloud. The company's user behavior analytics and identity access intelligence technology uses machine learning and predictive anomaly detection algorithms to reduce the attack surface for accounts, unnecessary access rights and privileges, and identify, predict and prevent breaches. Gurucul provides Hybrid Behavior Analytics (HBA) architecture with the breadth of Identity Access Intelligence to User Behavior Analytics, and the depth from cloud apps to on-premises behavior.

Gurucul is backed by an advisory board comprised of Fortune 500 CISOs, and world renowned-experts in government intelligence and cyber security. The company was founded by seasoned entrepreneurs with a proven track record of introducing industry changing enterprise security solutions. Our mission is to help organizations protect their intellectual property, regulated information, and brand reputation from insider threats and sophisticated external intrusions.

Gurucul technology is used globally by organizations to detect insider fraud, IP theft, external attacks and more.

RSA NetWitness – Gurucul Risk Analytics Diagram



Partner Product Configuration

Before You Begin

This section provides instructions for configuring Gurucul Risk Analytics (GRA) 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 Gurucul Risk Analytics 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 Gurucul Risk Analytics is properly configured and secured before deploying to a production environment. For more information, please refer to the Gurucul Risk Analytics documentation or website.

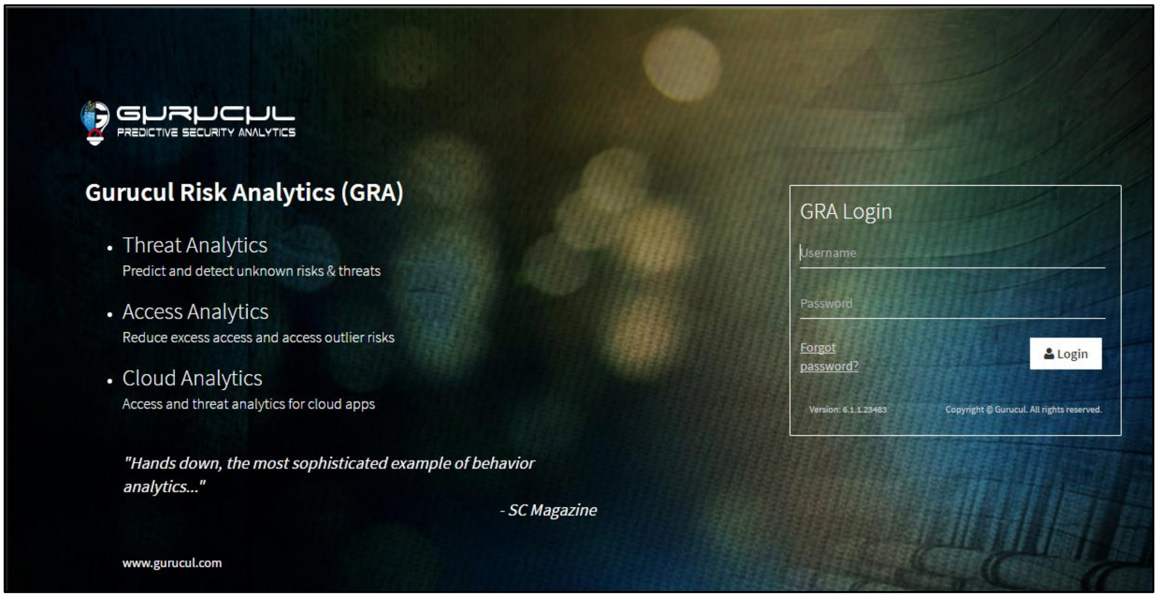
Creating a Data Source in Gurucul Risk Analytics (GRA)

1. Login to GRA

   Open a browser of your choice and type the following web address to launch the web interface:

   ```
   http://<hostname>:8080
   ```

   This opens the login page of GRA.



Predict and detect unknown risks & threats

Reduce excess access and access outlier risks

Access and threat analytics for cloud apps

"Hands down, the most sophisticated example of behavior analytics..."

- SC Magazine

Version: 6.1.1.23483

Copyright © Gurucul. All rights reserved.

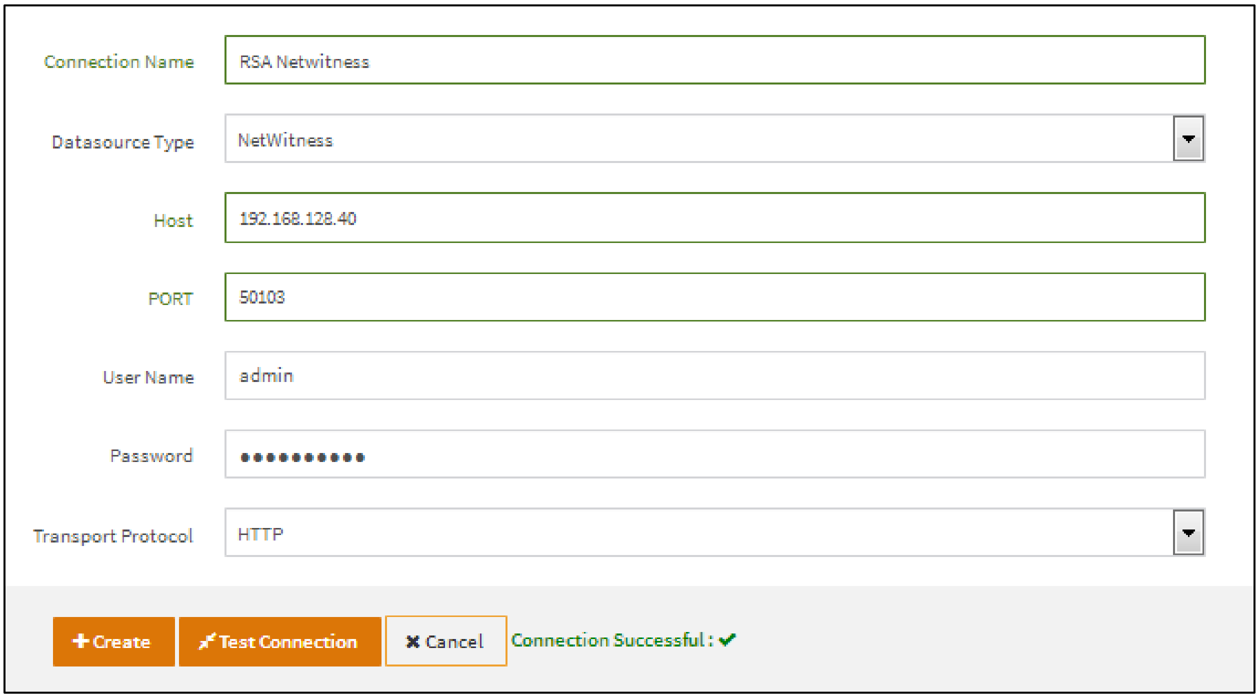
2. Navigate to **Configuration > Data > Setup > Data Source**
3. Click on the **+Add** button on the top right corner of the screen

4. Select **Datasource Type** as **NetWitness**.
5. **Host:** Enter the Hostname/IP of the Master Broker
6. **Port:** 50103
7. **Username:** <username> (Username for RSA NetWitness)

   Eg: Admin (user specific)
8. **Password:** <password> (Password for the RSA NetWitness)

   Eg: netwitness (user specific)
9. **Transport Protocol:** HTTP

After providing correct credentials and RSA Master Broker details, check for connectivity between the applications using the **Test Connection** button on the bottom of the screen. See the screenshot below:



Connection Name	RSA Netwitness
Datasource Type	NetWitness
Host	192.168.128.40
PORT	50103
User Name	admin
Password	●●●●●●●●●●●
Transport Protocol	HTTP

Connection Successful : ✓

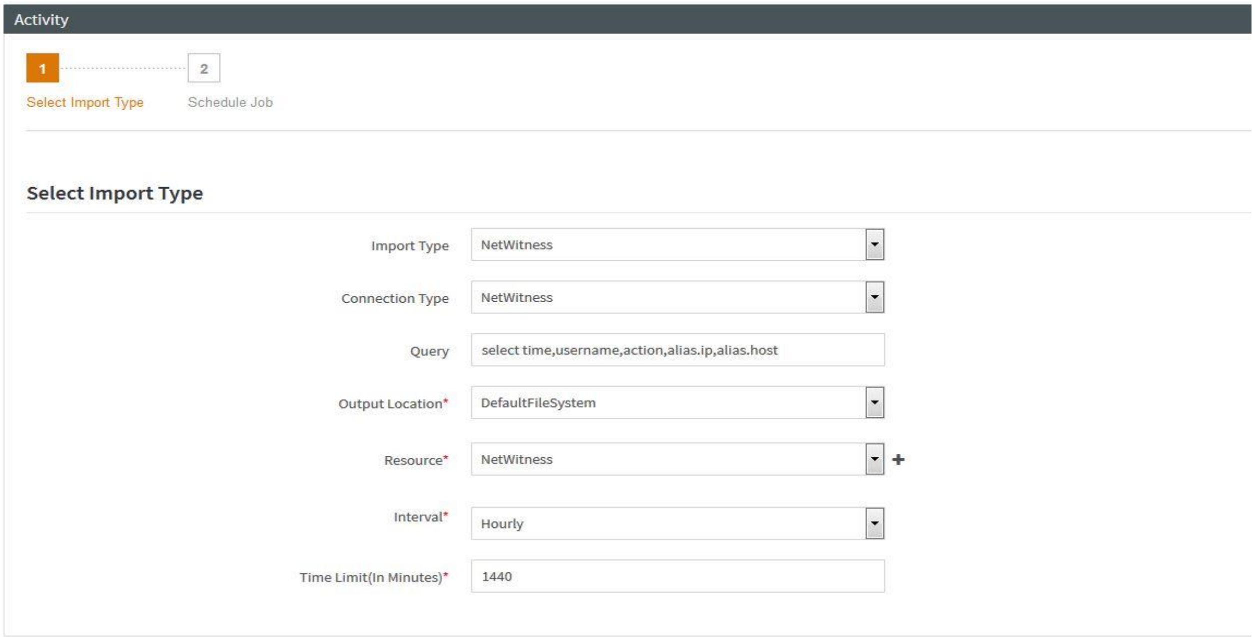
After the test connection is successful, click the **+Create** button to create the Datasource.

Create a Data Import Job in Gurucul Risk Analytics (GRA)

After the data source is created, the data present in RSA NetWitness can be imported into GRA for analysis.

1. Navigate to **Configuration > Data > Data Import > Activity**
2. Click the **+Add** button on the top right corner to create a new data import job.
3. Select the attributes for the job as follows:
    a. **Import Type:** NetWitness
    b. **Connection Type:** Data Source Name (eg: Netwitness)
    c. **Query:** select time,username,action,ip.src,tcp.dstport,filename,message (Sample)
    d. **Output Location:** DefaultFileSystem (where normalized data will be stored in the file under source folder).
    e. **Resource:** DLP, VPN, Firewall, VMWare etc.
    f. **Interval:** The time period for which we want to fetch logs Hourly(last 1 hour), Daily (last 1 day) , Last 7 Day, Last 30 Days , User defined date.
    g. **Time Limit (Minutes):** Time slicing of files (eg: 60, will create multiple files having all the events grouped by hour)
4. Click the **Next** button
5. Provide mappings for the fields:

> **Note: All the users must be present in GRA Global users before importing activities.**



Activity

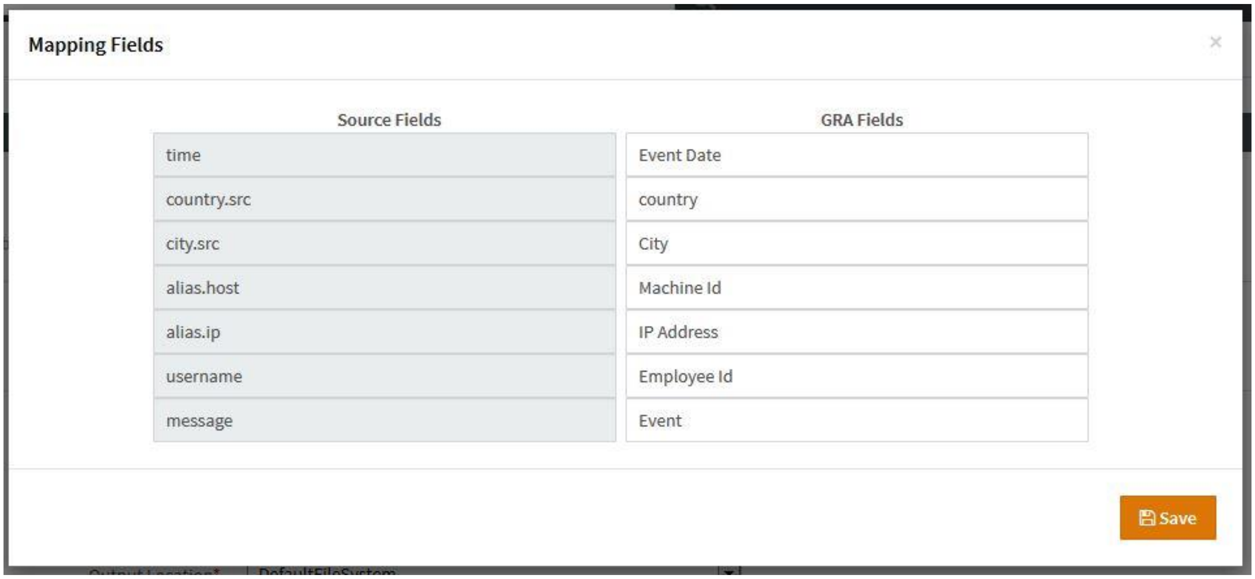
1 Select Import Type
2 Schedule Job

Select Import Type

Import Type	NetWitness
Connection Type	NetWitness
Query	select time,username,action,alias.ip,alias.host
Output Location*	DefaultFileSystem
Resource*	NetWitness
Interval*	Hourly
Time Limit(In Minutes)*	1440

The fields to be fetched are also dynamic, so they can be changed at any time by either adding or removing a field from the **gra-appconfig.properties** file with property name as

NetWitness Schema (**Configuration > System Settings > Properties > gra-appconfig.properties**)



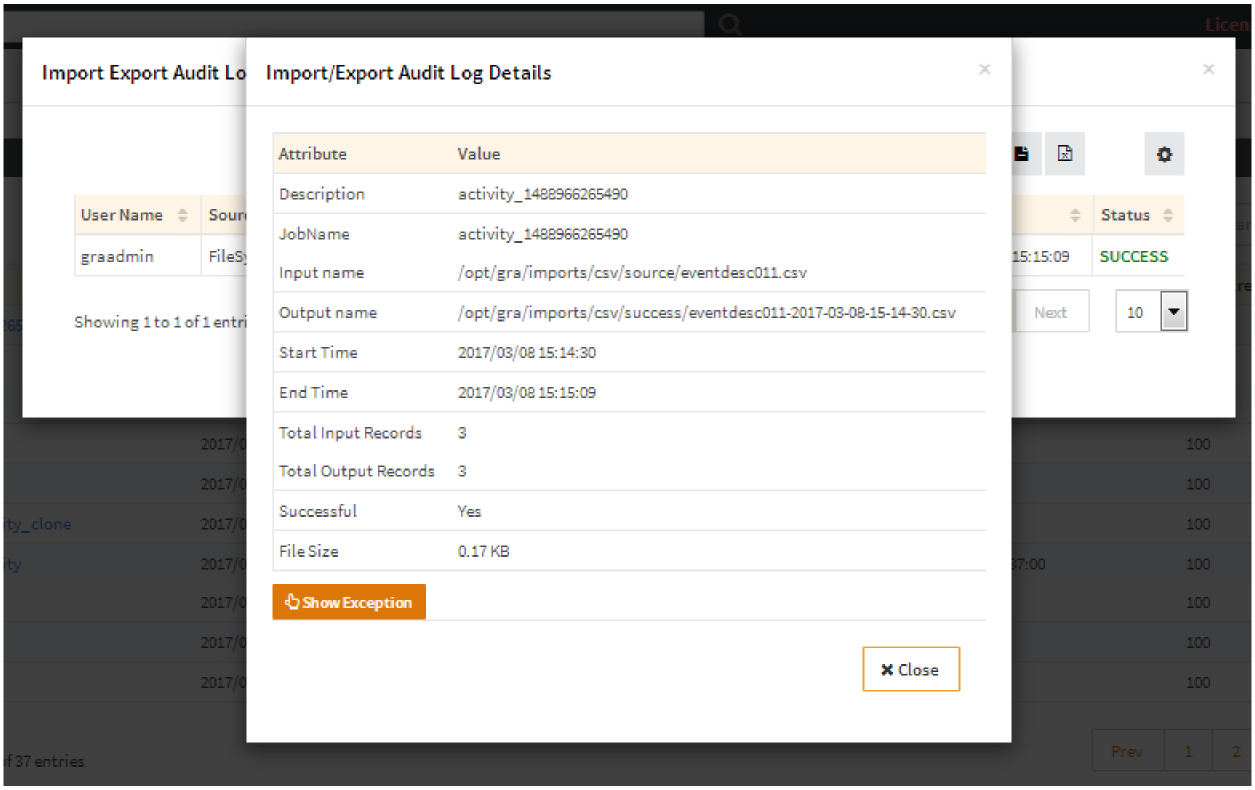
Mapping Fields

Source Fields	GRA Fields
time	Event Date
country.src	country
city.src	City
alias.host	Machine Id
alias.ip	IP Address
username	Employee Id
message	Event

Note: This is an example mapping (All the fields can be fetched from data source and mapped to GRA resource attributes)

Data Source Fields	GRA Fields
Time	Event Date* (Mandatory)
Username	Employee Id
alias.host	Machine Id
country.dst	Country (Txncustomfield14)
ip.dst	IP Address

6. **Save** the Mapping.
7. Provide the **Job name** and **Description** for the Job
8. The job can be scheduled (hour / day / Week / Month) or can be run directly by clicking **Run Now**.
9. After the Job is triggered, the status of the job can be validated on the **Activity Import Page** (**Configuration > Data > Data Import > Activity page**).



Import/Export Audit Log Details

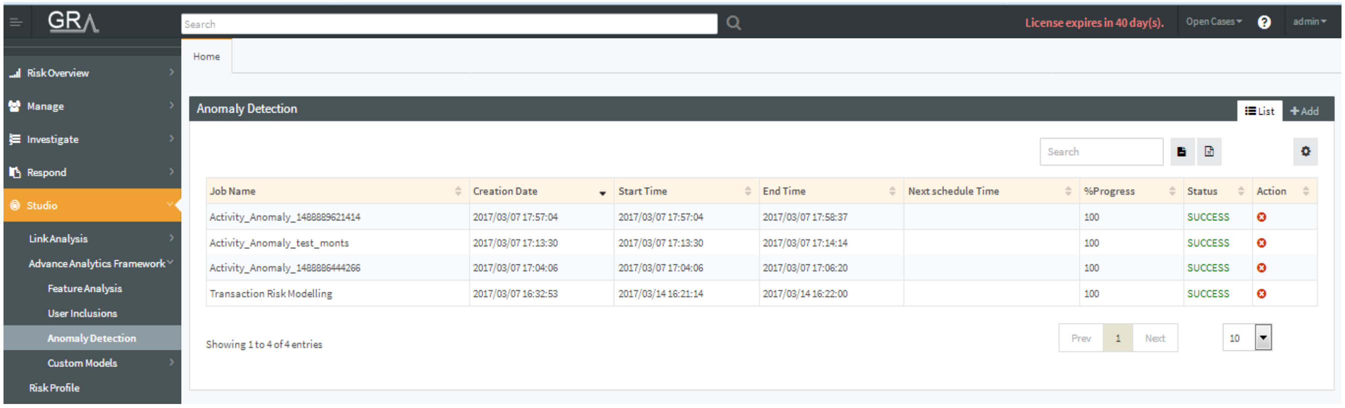
Attribute	Value
Description	activity_1488966265490
JobName	activity_1488966265490
Input name	/opt/gra/imports/csv/source/eventdesc011.csv
Output name	/opt/gra/imports/csv/success/eventdesc011-2017-03-08-15-14-30.csv
Start Time	2017/03/08 15:14:30
End Time	2017/03/08 15:15:09
Total Input Records	3
Total Output Records	3
Successful	Yes
File Size	0.17 KB

Creating an Anomaly Job in Gurucul Risk Analytics (GRA)

As all of the necessary data is stored in the GRA database, you can trigger a job to **Run User Analysis** based on the activities performed by the user.

To Run GRA Analysis (Activity Anomaly), perform the following steps:

1. Navigate to **Studio > Advance Analytics Framework > Anomaly Detection**.
2. Click the **+Add** button on the Top right corner.
3. Select the **All Users** tab, this will select all the users present in the GRA database.
4. Click **Next**.
5. Select **All Resources**.
6. **Data Period**, for which we want to detect if any user has done any Anomalous Activity during the provided time period.
7. Click the **+Add** button after providing the Data Period.
8. Schedule and Save the job.
    a. Provide the **Job name** and **Description**
    b. **Run** or **Schedule** the job.



Anomaly Detection

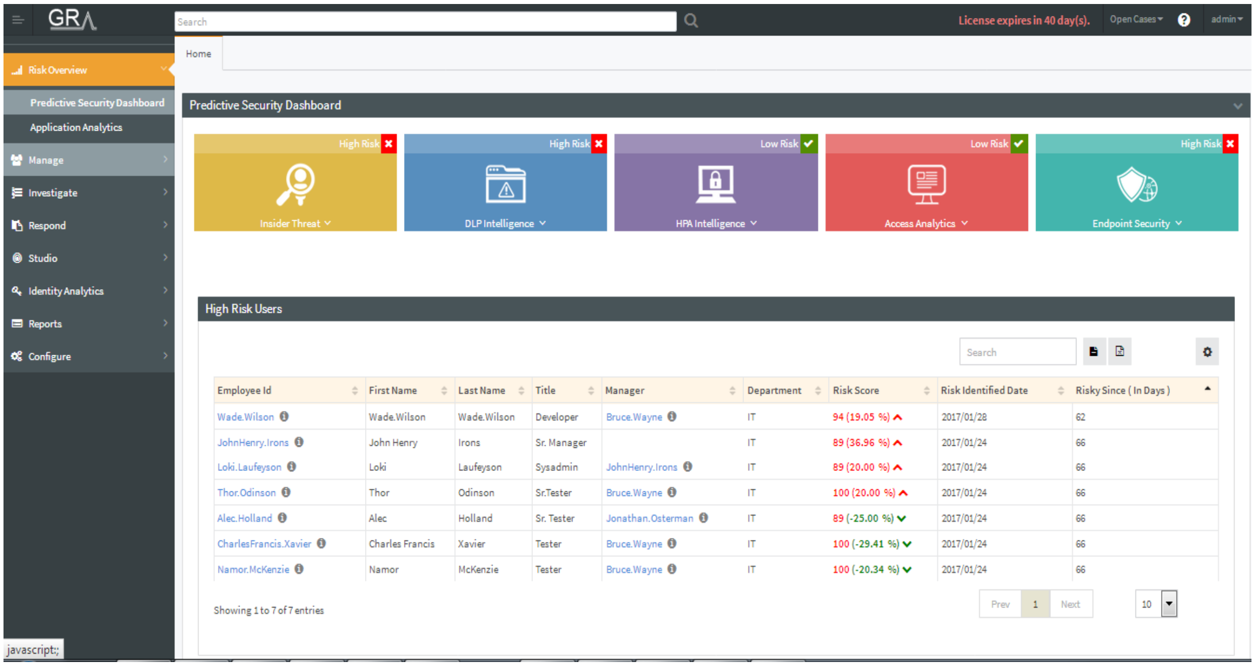
Job Name	Creation Date	Start Time	End Time	Next schedule Time	%Progress	Status	Action
Activity_Anomaly_1488889621414	2017/03/07 17:57:04	2017/03/07 17:57:04	2017/03/07 17:58:37		100	SUCCESS	✖
Activity_Anomaly_test_monts	2017/03/07 17:13:30	2017/03/07 17:13:30	2017/03/07 17:14:14		100	SUCCESS	✖
Activity_Anomaly_1488886444266	2017/03/07 17:04:06	2017/03/07 17:04:06	2017/03/07 17:06:20		100	SUCCESS	✖
Transaction Risk Modelling	2017/03/07 16:32:53	2017/03/14 16:21:14	2017/03/14 16:22:00		100	SUCCESS	✖

Showing 1 to 4 of 4 entries

The Job status can be validated on the same page (Activity Anomaly)

After the job has been completed, the Users having Anomalous data have been detected and provided with a Risk Score.

9. Navigate to **Risk Overview > Predictive Security Dashboard**



Predictive Security Dashboard

High Risk Users

Employee Id	First Name	Last Name	Title	Manager	Department	Risk Score	Risk Identified Date	Risky Since (In Days)
Wade.Wilson	Wade.Wilson	Wade.Wilson	Developer	Bruce.Wayne	IT	94 (19.05 %) ▲	2017/01/28	62
JohnHenry.Irons	John Henry	Irons	Sr. Manager		IT	89 (36.96 %) ▲	2017/01/24	66
Loki.Laufeyson	Loki	Laufeyson	Sysadmin	JohnHenry.Irons	IT	89 (20.00 %) ▲	2017/01/24	66
Thor.Odinson	Thor	Odinson	Sr.Tester	Bruce.Wayne	IT	100 (20.00 %) ▲	2017/01/24	66
Alec.Holland	Alec	Holland	Sr. Tester	Jonathan.Osterman	IT	89 (-25.00 %) ✓	2017/01/24	66
CharlesFrancis.Xavier	Charles Francis	Xavier	Tester	Bruce.Wayne	IT	100 (-29.41 %) ✓	2017/01/24	66
Namor.McKenzie	Namor	McKenzie	Tester	Bruce.Wayne	IT	100 (-20.34 %) ✓	2017/01/24	66

Showing 1 to 7 of 7 entries

All the users considered as **High Risk** will appear on this page.

Now that all of the high risk users with a number of anomalies have been detected, GRA can forward updated risk data to RSA NetWitness via syslog in CEF format.

Creating a Data Forwarder in Gurucul Risk Analytics (GRA)

Updated risk logs are forwarded to RSA NetWitness in CEF Format, using the default CEF parser for the RSA NetWitness suite. The updated risk logs are sent from GRA via syslog in CEF format to the NetWitness Log collector. Perform the following steps to configure the syslog forwarder:

Creating the NetWitness Data Source

1. Navigate to **Configuration > Data > setup > Data source**.
2. Click the **+Add** button on top right corner.
3. Select **Datasource** type as **NetWitness**.
4. **Host:** IP/Hostname of Log Collector
5. **Port:** 514 (Universal syslog port)
6. **Username:** admin (user specific)
7. **Password:** netwitness (user specific)
8. **Transport protocol:** UDP

Use the **Test Connection** to ensure that the configuration was successful.

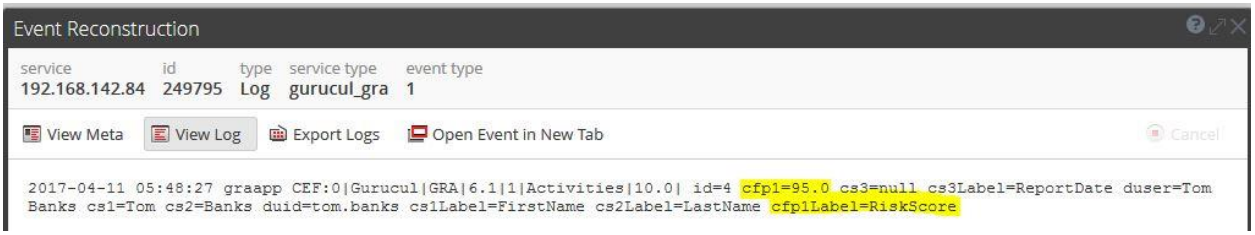
Generating a Data Forwarder Job

1. Navigate to **Configuration > Data > Data Exports > Data Forwarder**.
2. Click **Forwarder** in the top right corner
3. Select the job attributes as follows:
   a. **Destination Type:** NetWitness
   b. **Destination:** Datasource (created for reverse Integration)
   c. **Log Format:** CEF
4. Click **Next**.
5. Provide the **Job Name** and **Description**
6. **Schedule** or **Run Now**.

The data to be forwarded is dynamic, and the number of fields to be forwarded can be changed under (**Configuration > Data > Data Export > Data Forwarder > Configuration**)

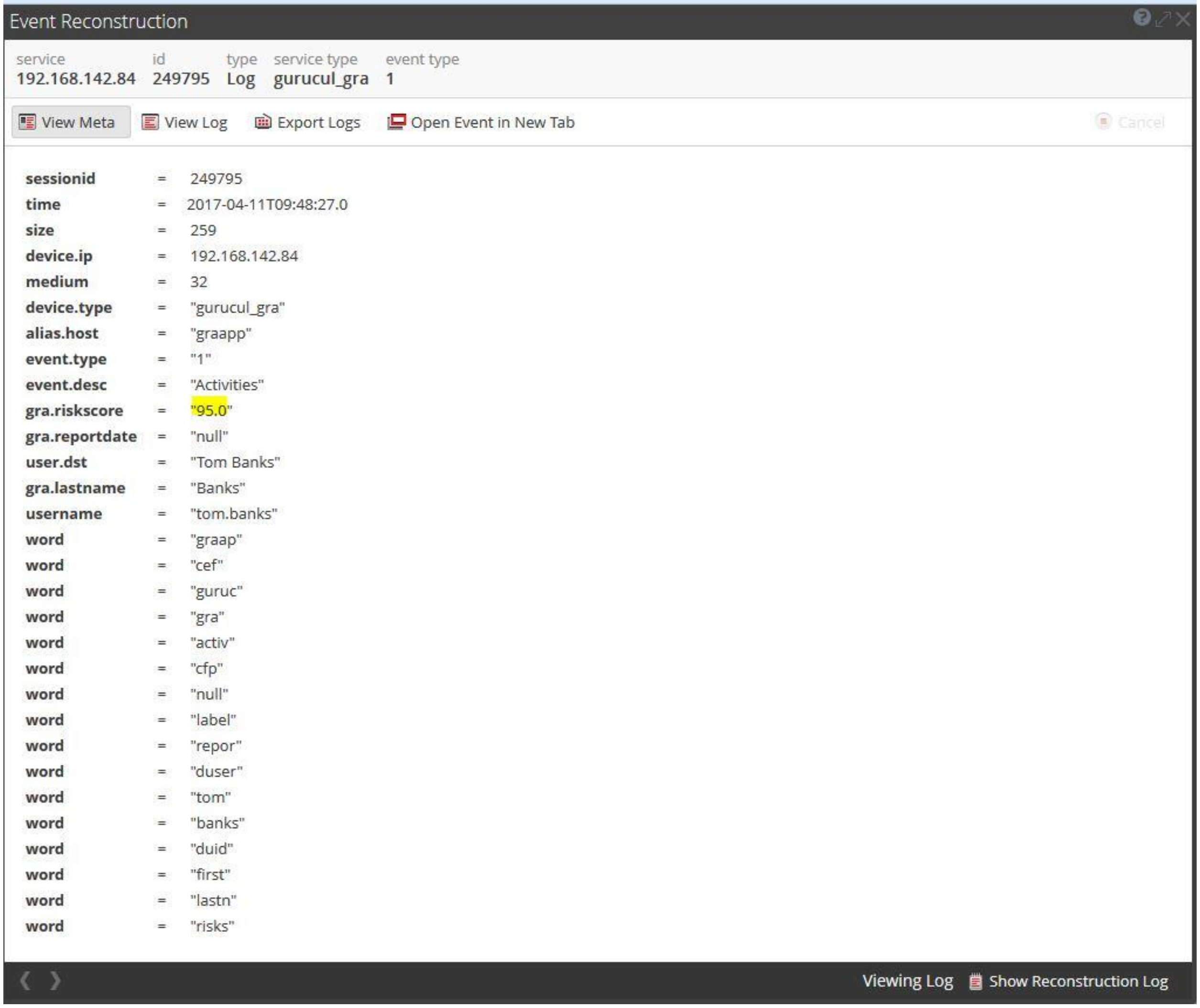
After the data is forwarded to RSA NetWitness, the status can be checked on the Forwarder job page.

Once logs are being forwarded correctly, the appropriate events are generated in RSA NetWitness. On the RSA NetWitness server, go to: **Navigate > Log Decoder > Events** and search for **GRA** on the search panel present on the right corner of the page.



service	id	type	service type	event type
192.168.142.84	249795	Log	gurucul_gra	1

```
2017-04-11 05:48:27 graapp CEF:0|Gurucul|GRA|6.1|1|Activities|10.0| id=4 cfp1=95.0 cs3=null cs3Label=ReportDate duser=Tom
Banks cs1=Tom cs2=Banks duid=tom.banks cs1Label=FirstName cs2Label=LastName cfp1Label=RiskScore
```



service	id	type	service type	event type
192.168.142.84	249795	Log	gurucul_gra	1

```
sessionid      =   249795
time           =   2017-04-11T09:48:27.0
size           =   259
device.ip      =   192.168.142.84
medium         =   32
device.type    =   "gurucul_gra"
alias.host     =   "graapp"
event.type     =   "1"
event.desc     =   "Activities"
gra.riskscore  =   "95.0"
gra.reportdate =   "null"
user.dst       =   "Tom Banks"
gra.lastname   =   "Banks"
username       =   "tom.banks"
word           =   "graap"
word           =   "cef"
word           =   "guruc"
word           =   "gra"
word           =   "activ"
word           =   "cfp"
word           =   "null"
word           =   "label"
word           =   "repor"
word           =   "duser"
word           =   "tom"
word           =   "banks"
word           =   "duid"
word           =   "first"
word           =   "lastn"
word           =   "risks"
```

RSA NetWitness Configuration

Once you deploy the enVision Config File and Deploy Common Event Format, you can now collect events from most sources supporting the Common Event Format (CEF).

To capture GRA custom messages not displayed by default in RSA NetWitness, modification to the RSA NetWitness standard **CEF.xml** and **table-map-custom.xml** is required.

For more information of working with custom keys in the CEF parser, consult the following document on RSA Link:

https://community.rsa.com/docs/DOC-45504

CEF.xml file Modifications

! › Important: In an environment with multiple Log Decoders, modify the CEF File on each Log Decoder in your network. Ensure you have backed up the original CEF.xml to a safe location. The CEF.xml file is updated when you perform an RSA Live system update and if the RSA Live Subscription is enabled.

Prior to upgrading your NetWitness servers, backup any files containing customizations to ensure your work is preserved.

1. Modify the **CEF.xml** on the Log Decoder (CEF keys to modify are cs1, cs2, cs3, cs4, cs5, cs6). The **CEF.xml** file can be found in /etc/netwitness/ng/envision/etc/devices/cef/.
2. Add the entire portion of code below including the <MESSAGE.../>, place this entry below the last <MESSAGE.../> entry.

```
<MESSAGE

       level="4"
       parse="1"
       parsedefvalue="1"
       tableid="74"
       id1="Gurucul_gra"
       id2="Gurucul_gra"
       eventcategory="1303000000"
       content="&lt;@msg:*PARMVAL($MSG)&gt;@starttime:*EVENTTIME($MSG,'%W-%M-%D
       %H:%T:%S',param_starttime)&gt;&lt;param_starttime&gt;.&lt;fld5&gt;
       &lt;msghold&gt;"/>
```

3. Modify the cef ExtensionKey cefName cs1, adding only the highlighted text below.

```
<ExtensionKey cefName="cs1" metaName="cs_fld" >
   <device2meta device="trendmicrodsa" metaName="context"/>
   <device2meta device="bluecat" metaName="action" label="query"/>
   <device2meta device="websense" metaName="policyname" label="Policy"/>
   <device2meta device="mcafeewg" metaName="virusname" label="Virus Name"/>
   <device2meta device="bit9" metaName="checksum" label="File Hash"/>
   <device2meta device="mcafeereconnex" metaName="policyname"/>
   <device2meta device="Gurucul_gra" metaName="gra.firstname" label="FirstName"/>
</ExtensionKey>
```

4. Modify the cef ExtnesionKey cefName cs2, adding only the highlighted text below.

```
<ExtensionKey cefName="cs2" metaName="cs_fld">
      <device2meta device="bit9" metaName="v_instafname" label="installerFilename"/>
```

```
       <device2meta device="Gurucul_gra" metaName="gra.lastname" label="LastName"/>
</ExtensionKey>
```

5. Modify the cef ExtensionKey cefName cs3, adding only the highlighted text below.

```
<ExtensionKey cefName="cs3" metaName="cs_fld">
       <device2meta device="websense" metaName="content_type" label="ContentType"/>
       <device2meta device="bit9" metaName="policyname"/>
       <device2meta device="mcafeereconnex" metaName="content_type"/>
       <device2meta device="Gurucul_gra" metaName="gra.reportdate"
label="ReportDate"/>
</ExtensionKey>
```

6. Modify the cef ExtensionKey cefName cfp1, adding only the highlighted text below.

```
<ExtensionKey cefName="cfp1" metaName="cn_fld">
       <device2meta device="Gurucul_gra" metaName="gra.riskscore"
label="RiskScore"/></ExtensionKey>
```

Table-map-custom.xml file Modifications

! Important: If the table-map-custom.xml does not exist, create one and set the file permissions appropriately. If appending to an existing table-map-custom.xml file only add the individual <mapping envisionName=...> and do not repeat the <mappings> or </mappings> entries.

1. Modify the table-map-custom.xml on the Log Decoder.
2. If the table-map-custom file was previously created it can be found in /etc/netwitness/ng/envision/etc/ otherwise you will need to create the xml file using the following text.

```
<mappings>
<mapping envisionName="gra.firstname" nwName="gra.firstname" flags="None"
format="Text"/>
<mapping envisionName="gra.lastname" nwName="gra.lastname" flags="None"
format="Text"/>
<mapping envisionName="gra.riskscore" nwName="gra.riskscore" flags="None"
format="Text"/>
<mapping envisionName="gra.reportdate" nwName="gra.reportdate" flags="None"
format="Text"/>
</mappings>
```

Reboot the Decoder after the above configuration.

Certification Checklist for RSA NetWitness

Date Tested: April 14, 2017

Certification Environment		
Product Name	**Version Information**	**Operating System**
RSA NetWitness	10.6.0.0.22075-5	Virtual Appliance
Gurucul Risk Analytics	6.2.0.25960	CentosOS 7

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

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