RSA® NETWITNESS®
Logs
Implementation Guide

ObserveIT 7.1.0

Daniel R. Pintal, RSA Partner Engineering
Last Modified: November 2, 2017

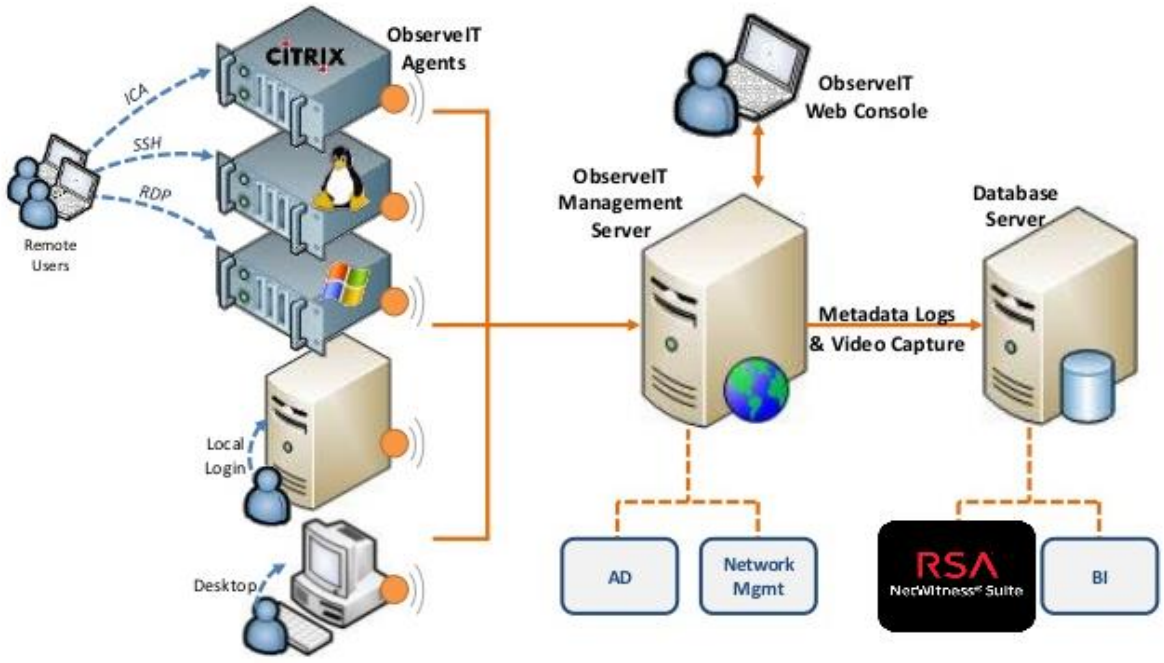
RSA READY

Solution Summary

The ObserveIT and RSA NetWitness integration provide three main business cases;

- Upload all ObserveIT video activity logging information into NetWitness.
- Enable NetWitness to run reports on ObserveIT video logs
- Enable NetWitness to run reports on ObserveIT logs containing http video links that are copied and used by any internet browser.

RSA NetWitness Features	
ObserveIT 7.1.0	
Integration package name	Common Event Format
Device display name within NetWitness	observerit_observeit
Event source class	Access
Collection method	Syslog CEF



RSA NetWitness Community

The RSA NetWitness Community is an online forum for customers and partners to exchange technical information and best practices with each other. All NetWitness customers and partners are invited to register and participate in the **RSA NetWitness Community**.

Release Notes

Release Date	What's New In This Release
11/2/2017	Initial support for ObserveIT Syslog CEF.

Important: The RSA NetWitness CEF parser is dependent on the partner adhering to the CEF Rules outlined in the *ArcSight Common Event Format (CEF) Guide*. A copy of the Common Event Format guide can be found on http://protect724.hp.com/**.**

Eg. Jan 18 11:07:53 host CEF:Version|Device Vendor|Device Product|Device Version|Signature ID|Name|Severity|[Extension]

Important: The time displayed in the CEF log header is parsed into evt.time.str. No other time formats are parsed by default.

Partner Product Configuration

Before You Begin

This section provides instructions for configuring the ObserveIT 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 ObserveIT 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 ObserveIT is properly configured and secured before deploying to a production environment. For more information, please refer to the ObserveIT documentation or website.**

ObserveIT Configuration

Configuring SIEM Log Integration

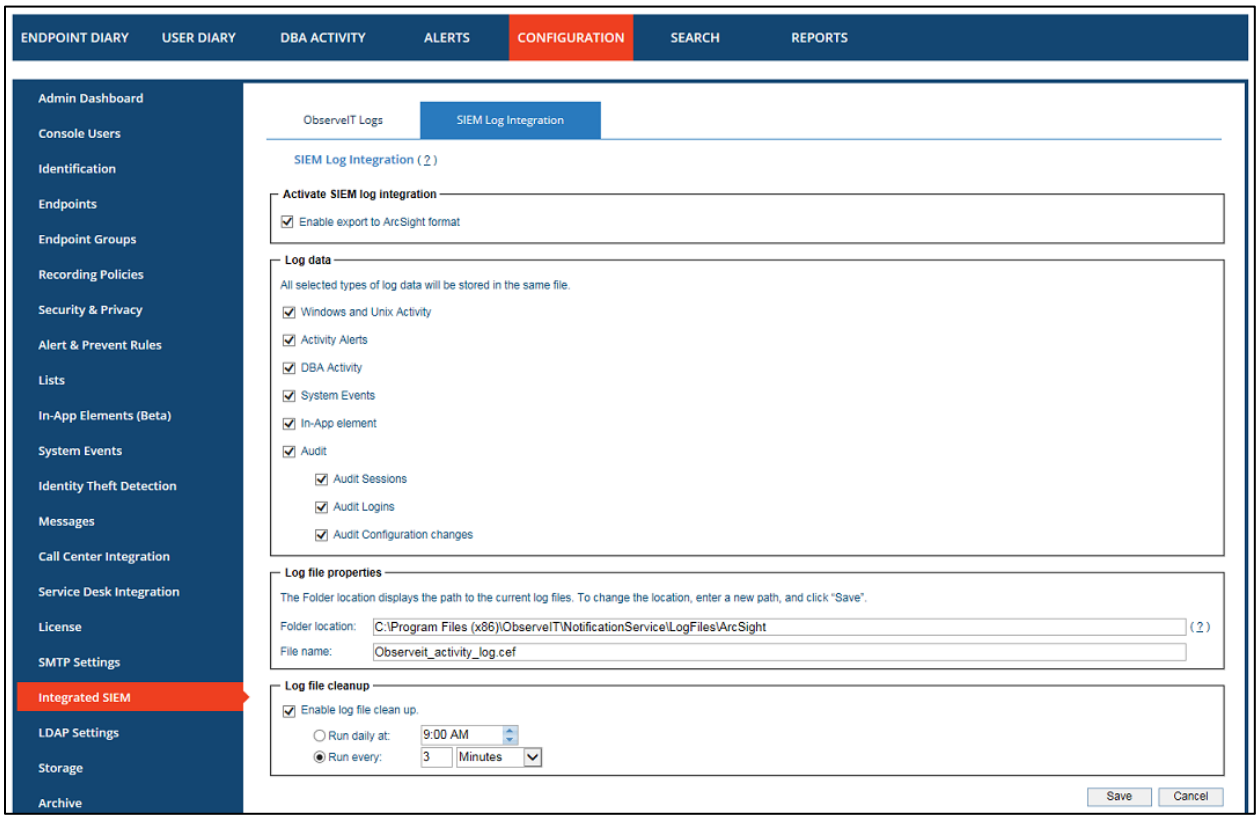
The following procedure describes how to configure SIEM log integration, including:

- Activating SIEM log integration and selecting the log data types.
- Specifying the log file location and log file name.
- Scheduling a log file cleanup.

> **Important: By default, SIEM log integration is disabled. You cannot enable both ObserveIT logging and SIEM logging simultaneously, since this might cause serious performance issues.**

To configure SIEM log integration

1. Navigate to **Configuration** > **Integrated SIEM**.
2. Click the **SIEM Log Integration** tab.



3. Select the Enable export to ArcSight format check box.
4. In the Log data section, select at least one of the following data types for monitoring:

    - Windows and Unix Activity - selected by default.
    - Activity Alerts - selected by default
    - DBA Activity
    - System Events
    - In-App Elements
    - Audit

        - Audit Sessions
        - Audit Logins
        - Audit Configuration Changes

All selected log type data will be stored in one file; by default, `Observeit_activity_log.cef`.

5. In the Log file properties section:
    1. In the Folder location field, accept the default log file location **C:\Program Files(x86)\ObserveIT\NotificationService\LogFiles\ArcSight** or specify a new path to the monitor log files. When changing the default log folder location, new session data will be stored in the new path; existing data will remain in the old location.

> **! Important: The user account used by the ObserveIT Notification Service must have read and write permissions for the path. If the user account does not have sufficient permissions to create the directory or write to the log file, a system event is generated. In addition, the log file size is limited to a predefined size; if the file size exceeds the maximum defined size, a system event will be generated. For further details, see System Events.**

    2. In the File name field, use the default log file name Observeit_activity_log.cef or specify a new one.
6. In the Log file cleanup section, schedule the frequency for clearing the log file:
    a) Select **Run daily** at, and specify the required time of day for the daily cleanup.
    3. Or
    4. Select **Run every**, and specify the required number of days, hours, or minutes for the cleanup.
7. Click Save to save the settings.

After a few minutes, the log file will be generated. A new log file will be created according to the scheduled cleanup frequency.

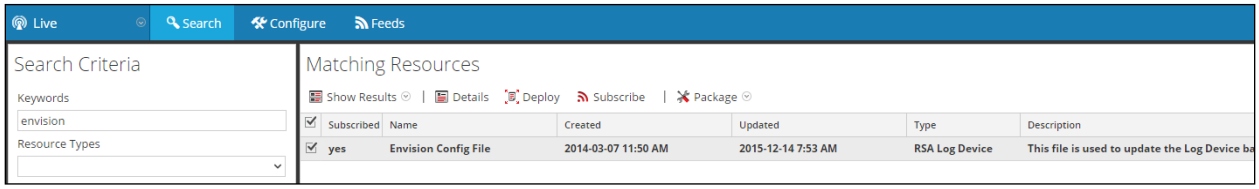
RSA NetWitness Configuration

Deploy the enVision Config File

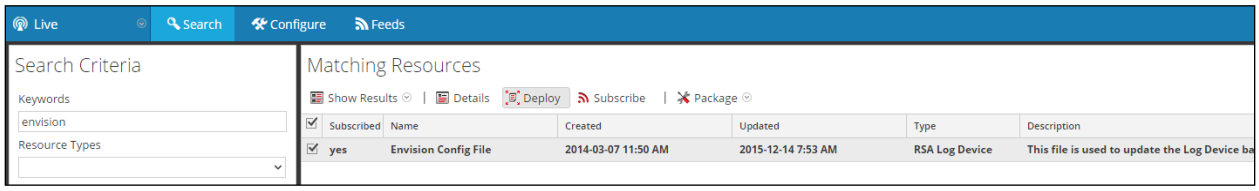
In order to use the RSA Common Event Format, you must first deploy the *enVision Config File* from the **NetWitness Live** module. Log into NetWitness and perform the following actions:

> **Important: Using this procedure will overwrite the existing table_map.xml.**

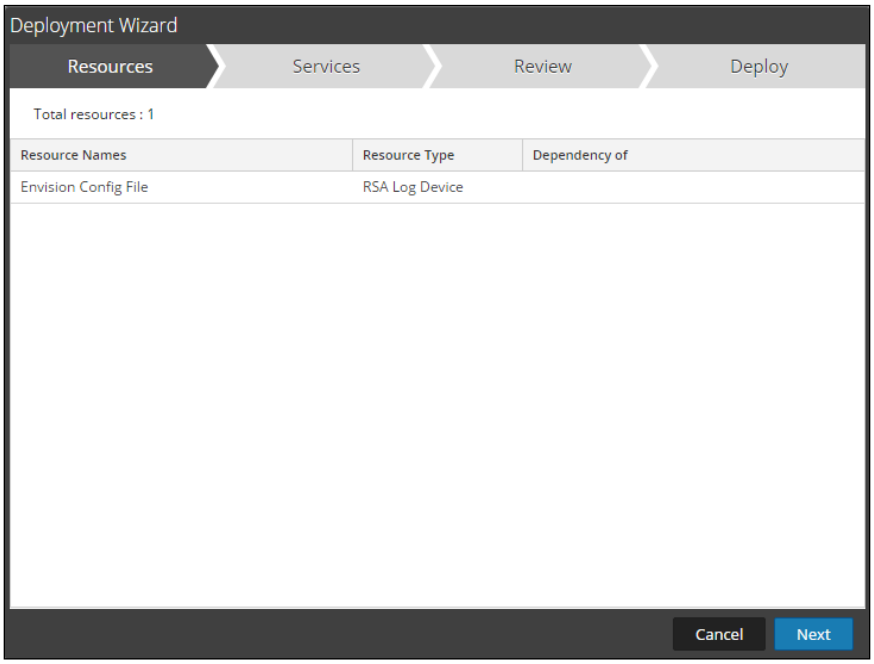
1. From the NetWitness menu, select **Live > Search**.
2. In the keywords field, enter: **enVision**.
3. NetWitness will display the **Envision Config File** in Matching Resources.
4. Select the checkbox next to **Envision Config File**.



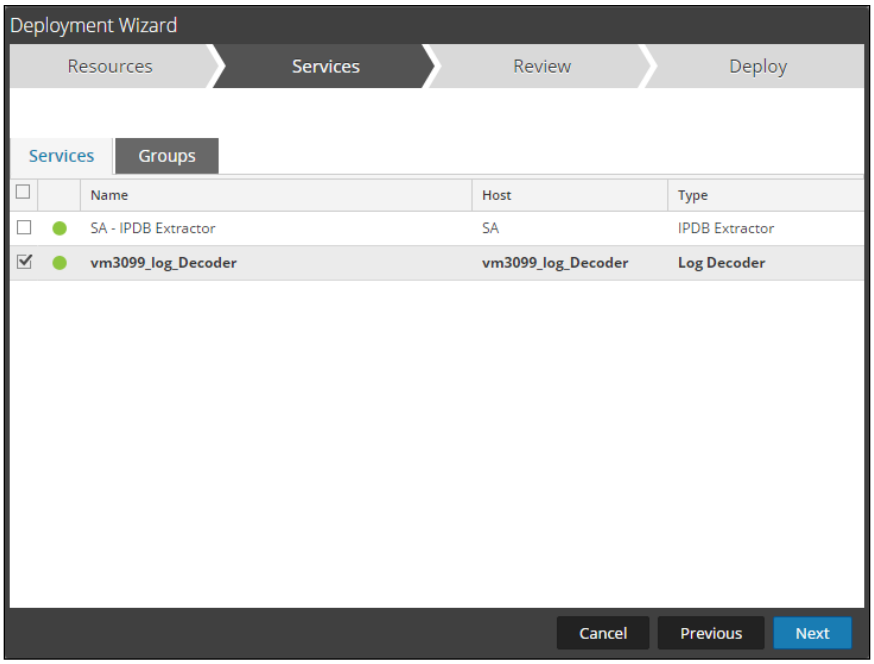
5. Click **Deploy** in the menu bar.



6. Select **Next**.

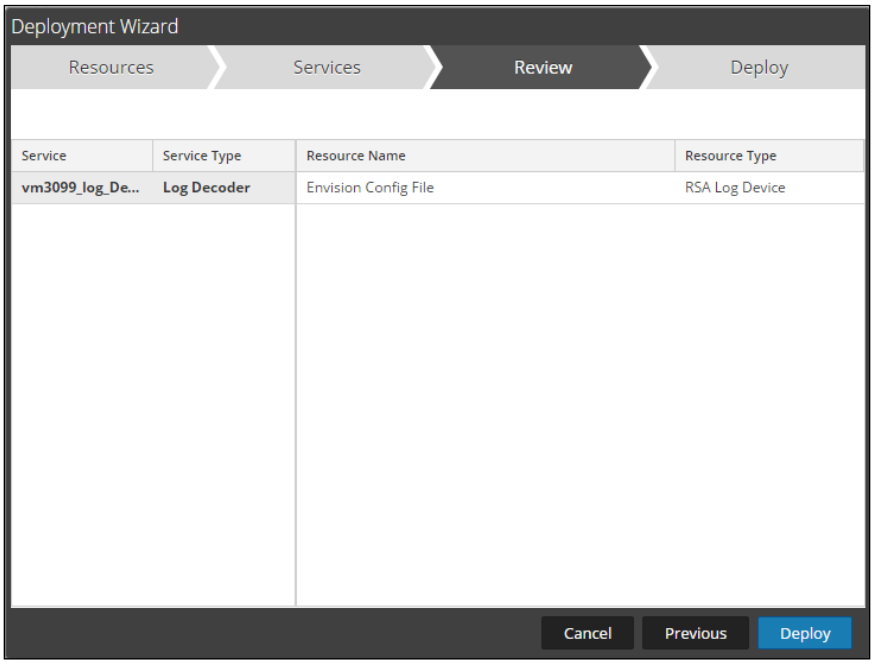


7. Select the **Log Decoder** and select **Next**.

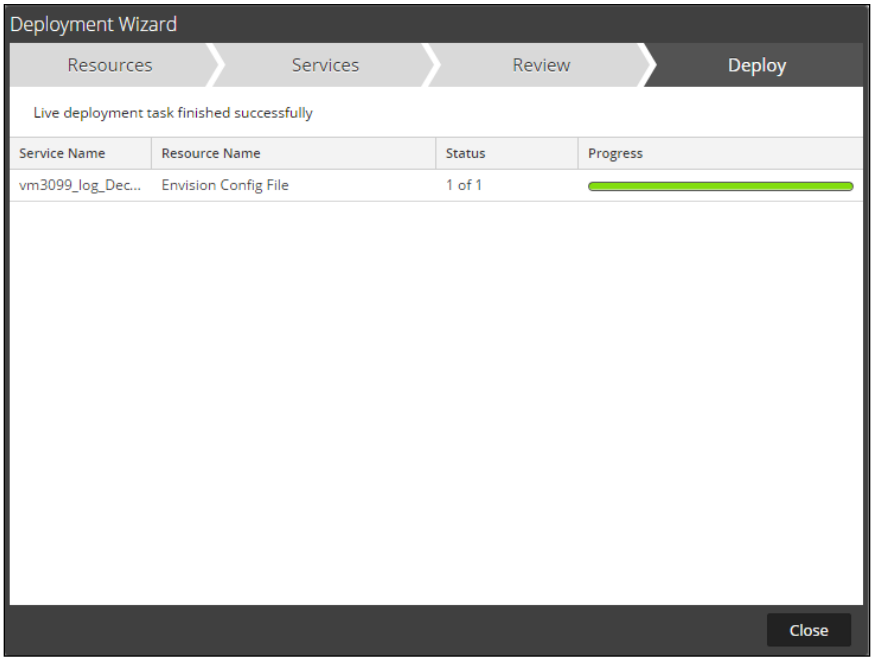


! Important: In an environment with multiple Log Decoders, deploy the Envision Config File to each Log Decoder in your network.

8. Select **Deploy**.



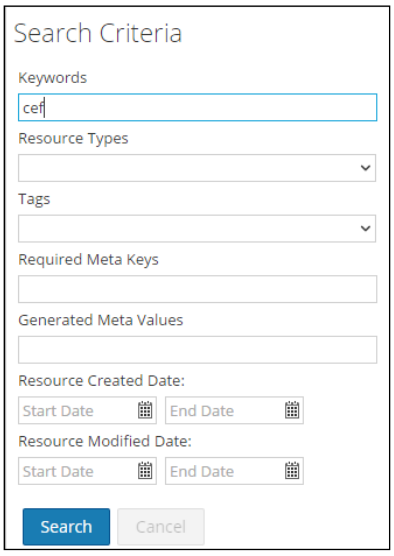
9. Select **Close**, to complete the deployment of the Envision Config file.



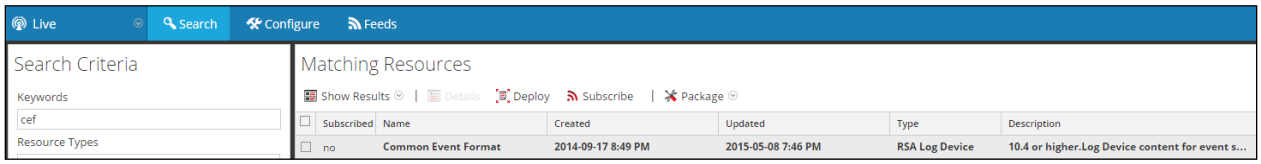
Deploy the Common Event Format

Next, you will need to deploy the *Common Event Format file* from the **NetWitness Live** module. Log into NetWitness and perform the following actions:

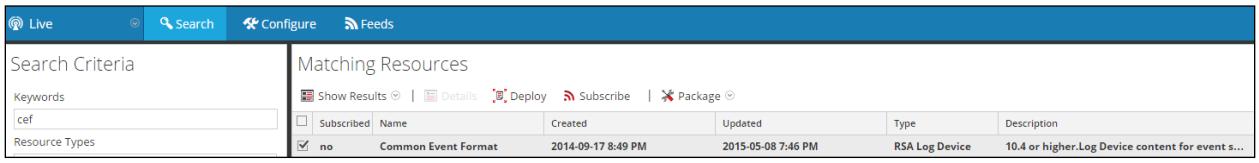
1. From the NetWitness menu, select **Live > Search**.
2. In the keywords field, enter: **CEF**



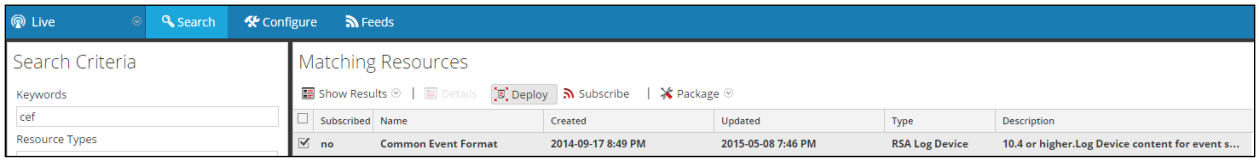
3. RSA NetWitness will display the **Common Event Format** in Matching Resources.



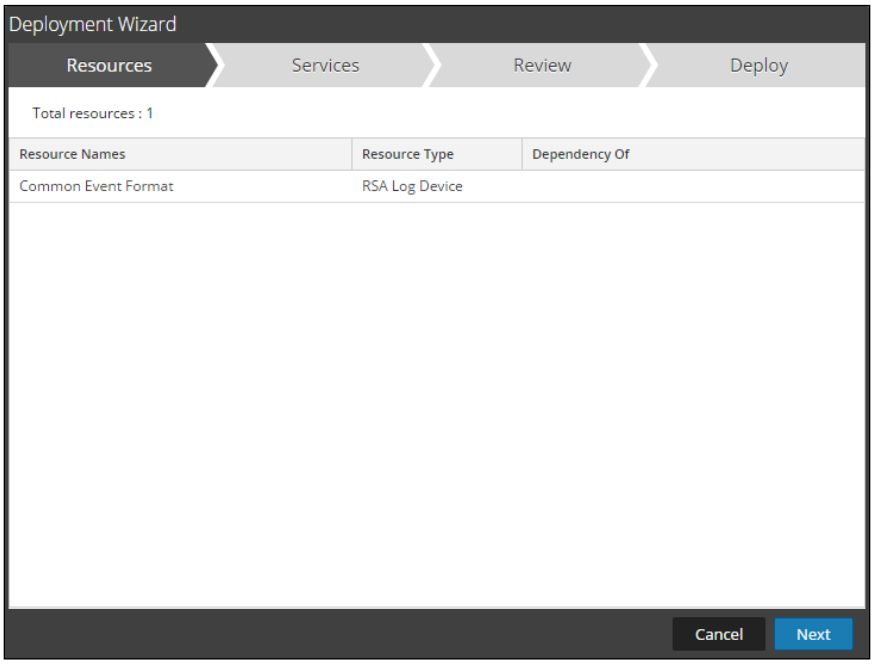
4. Select the checkbox next to **Common Event Format**.



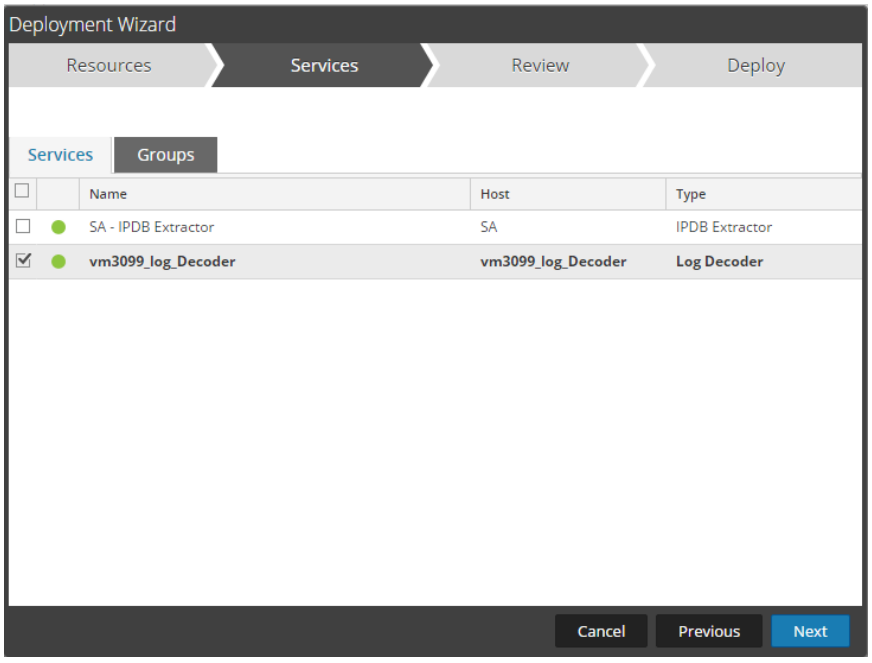
5. Click **Deploy** in the menu bar.



6. Select **Next**.

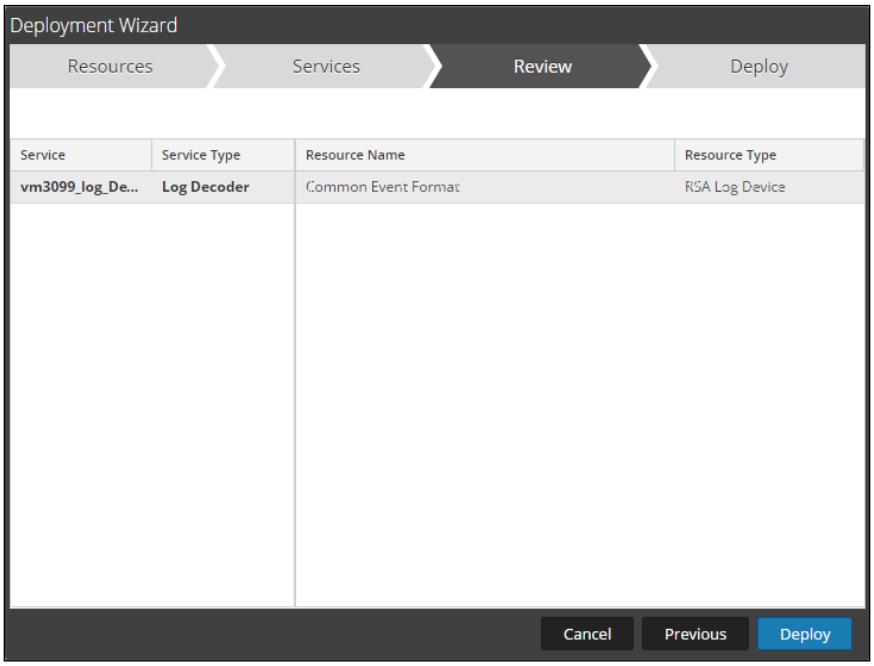


7. Select the **Log Decoder** and Select **Next**.

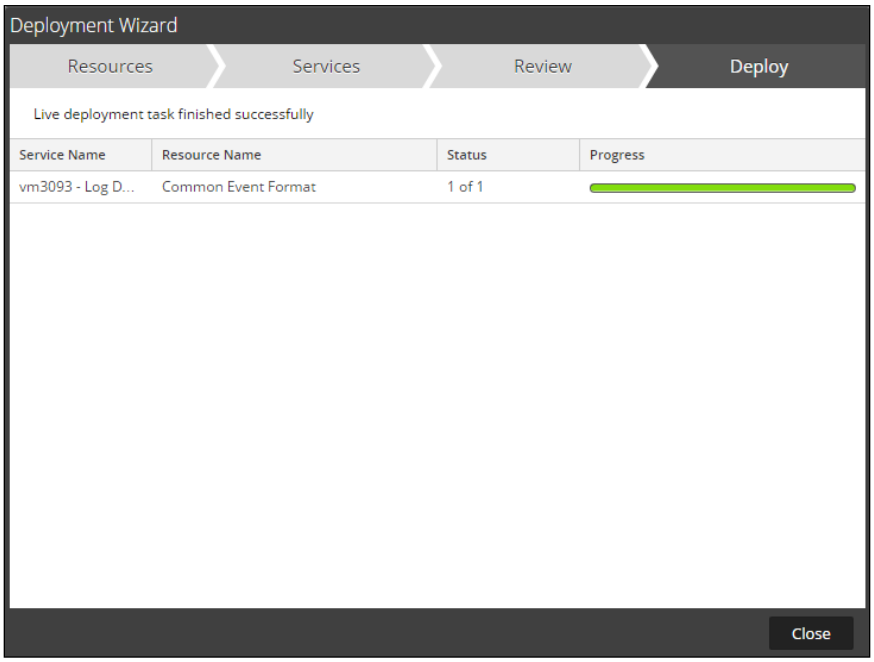


! Important: In an environment with multiple Log Decoders, deploy the Common Event Format to each Log Decoder in your network.

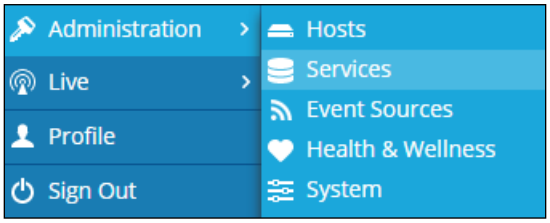
8. Select **Deploy**.



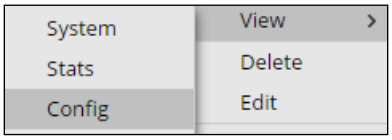
9. Select **Close**, to complete the deployment of the Common Event Format.



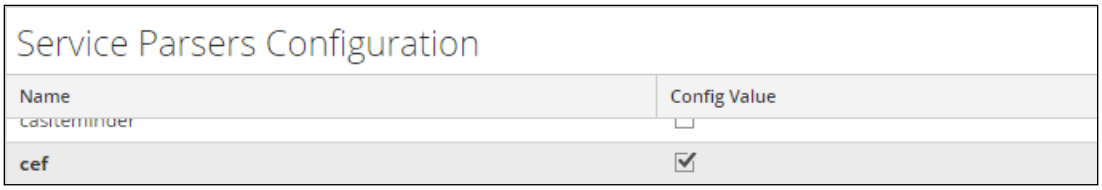
10. Ensure that the CEF Parser is enabled on the Log Decoder(s) by selecting **Administration, Services** from the NetWitness Dashboard.



11. Locate the Log_Decoder and click the gear to the right and select **View, Config**.



12. **Check** the box next to the cef Parser within the Service Parsers Configuration and select **Apply**.



13. Restart the **Log Decoder services**.

Edit the Common Event Format to collect ObserveIT event times

Important: The cef.xml file is overwritten by NetWitness Live during updates, it is important to maintain backups of the file in the event of a typing error or unforeseen event.

1. Using WinSCP or other application to access the RSA NetWitness Log Decoder open a connection and locate the **/etc/netwitness/ng/envision/etc/devices/cef** folder. Backup cef.xml and edit the existing CEF.XML file.
2. Locate the end of the <MESSAGE section and copy/paste the following lines below into the file after the /> of the preceding <MESSAGE and contents;

Example.

```
<MESSAGE
                level="4"
                parse="1"
                parsedefvalue="1"
                tableid="74"
                id1="observeit_observeit"
                id2="observeit_observeit"
                eventcategory="1901000000"

content="&lt;@event_name:*HDR(event_description)&gt;&lt;@msg:*PARMVAL($MSG)&gt;
&lt;@event_time:*EVNTTIME($MSG,'%B %F
%H:%T:%S',param_event_time)&gt;&lt;msghold&gt;&lt;@endtime:*EVNTTIME($MSG,'%B
%F %H:%T:%S',param_endtime)&gt;&lt;@starttime:*EVNTTIME($MSG,'%B %F
%H:%T:%S',param_starttime)&gt;&lt;param_event_time&gt;&lt;param_endtime&gt;&lt;
param_starttime&gt;" />
```

Edit the Common Event Format Custom to support custom fields

Important: The cef-custom.xml file is not overwritten by NetWitness Live during updates, however it is important to maintain backups of the file in the event of a typing error or unforeseen event.

1. Using WinSCP or other application to access the RSA NetWitness Log Decoder open a connection and locate the **/etc/netwitness/ng/envision/etc/devices/cef** folder. If the cef-custom.xml file does not exist create one. If the file exists create a backup cef-custom.xml and edit the file.
2. If this is a new cef-custom.xml file, copy the following into the file, otherwise copy only the required sections.

Example.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<DEVICEMESSAGES>
<!--

# ObserveIT DRP 9-25-2017
#
# cef-custom.xml Reference: https://community.rsa.com/docs/DOC-79189
#

<<DRP 9-25-2017 Place in CEF.XML>>

<MESSAGE
              level="4"
              parse="1"
              parsedefvalue="1"
              tableid="74"
              id1="observeit_observeit"
              id2="observeit_observeit"
              eventcategory="1901000000"

       content="&lt;@event_name:*HDR(event_description)&gt;&lt;@msg:*PARMVAL($MS
G)&gt;&lt;@event_time:*EVNTTIME($MSG,'%B %F
%H:%T:%S',param_event_time)&gt;&lt;msghold&gt;&lt;@endtime:*EVNTTIME($MSG,'%B
%F %H:%T:%S',param_endtime)&gt;&lt;@starttime:*EVNTTIME($MSG,'%B %F
%H:%T:%S',param_starttime)&gt;&lt;param_event_time&gt;&lt;param_endtime&gt;&lt;
param_starttime&gt;" />
-->

<VendorProducts>

       <Vendor2Device vendor="ObserveIT" product="ObserveIT"
device="observeit_observeit" group="DLP"/>

</VendorProducts>

           <ExtensionKeys>
              <ExtensionKey cefName="cat" metaName="cat"/>
              <ExtensionKey cefName="externalId" metaName="hardware_id"/>
              <ExtensionKey cefName="reason" metaName="result"/>
              <ExtensionKey cefName="origin" metaName="origin"/>

              <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="observeit_observeit"
metaName="AlertDetails"/>
              </ExtensionKey>
              <ExtensionKey cefName="cs1Label" metaName="cs_fld" />

              <ExtensionKey cefName="cs2" metaName="cs_fld">
                    <device2meta device="bit9" metaName="v_instafname"
label="installerFilename"/>
                    <device2meta device="observeit_observeit" metaName="OS"/>
              </ExtensionKey>
              <ExtensionKey cefName="cs2Label" metaName="cs_fld"/>

              <ExtensionKey cefName="dhost" metaName="dhost"/>
              <ExtensionKey cefName="dntdom" metaName="ddomain"/>

              <ExtensionKey cefName="cs5" metaName="cs_fld">
                    <device2meta device="mcafeewg" metaName="policyname"
label="Policy"/>
                    <device2meta device="bit9" metaName="rulename"
label="ruleName"/>
                    <device2meta device="observeit_observeit"
metaName="AlertDetailsURL"/>
              </ExtensionKey>
              <ExtensionKey cefName="cs5Label" metaName="cs_fld"/>

              <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="observeit_observeit"
metaName="ViewURL"/>
              </ExtensionKey>
              <ExtensionKey cefName="cs3Label" metaName="cs_fld"/>

              <ExtensionKey cefName="dproc" metaName="process"/>
              <ExtensionKey cefName="duid" metaName="uid"/>
              <ExtensionKey cefName="duser" metaName="username"/>
              <ExtensionKey cefName="dvchost" metaName="hostname"/>
              <ExtensionKey cefName="dvc" metaName="hostip"/>
              <ExtensionKey cefName="msg" metaName="msg"/>
              <ExtensionKey cefName="rt" metaName="param_event_time"/>
              <ExtensionKey cefName="shost" metaName="shost"/>
              <ExtensionKey cefName="sproc" metaName="process_src"/>
              <ExtensionKey cefName="src" metaName="saddr"/>
              <ExtensionKey cefName="sntdom" metaName="sdomain"/>
              <ExtensionKey cefName="suser" metaName="c_username"/>
              <ExtensionKey cefName="suid" metaName="uid"/>
              <ExtensionKey cefName="destinationServiceName"
metaName="destinationServiceName"/>
              <ExtensionKey cefName="deviceProcessName"
metaName="deviceProcessName"/>
              <ExtensionKey cefName="sourceServiceName"
metaName="sourceServiceName"/>
              <ExtensionKey cefName="requestMethod" metaName="requestMethod"/>
              <ExtensionKey cefName="start" metaName="param_starttime"/>
              <ExtensionKey cefName="end" metaName="param_endtime"/>

              </ExtensionKeys>

</DEVICEMESSAGES>
```

Edit the NetWitness Table-Map-Custom.xml file

Important: The Table-Map-Custom.xml file is not overwritten by NetWitness Live during updates, however it is important to maintain backups of the file in the event of a typing error or unforeseen event.

1. Using WinSCP or other application to access the RSA NetWitness Log Decoder open a connection and locate the /etc/netwitness/ng/envision/etc/ folder.

2. If one exists, backup the table-map-custom.xml and then edit the existing table-map-custom.xml file.

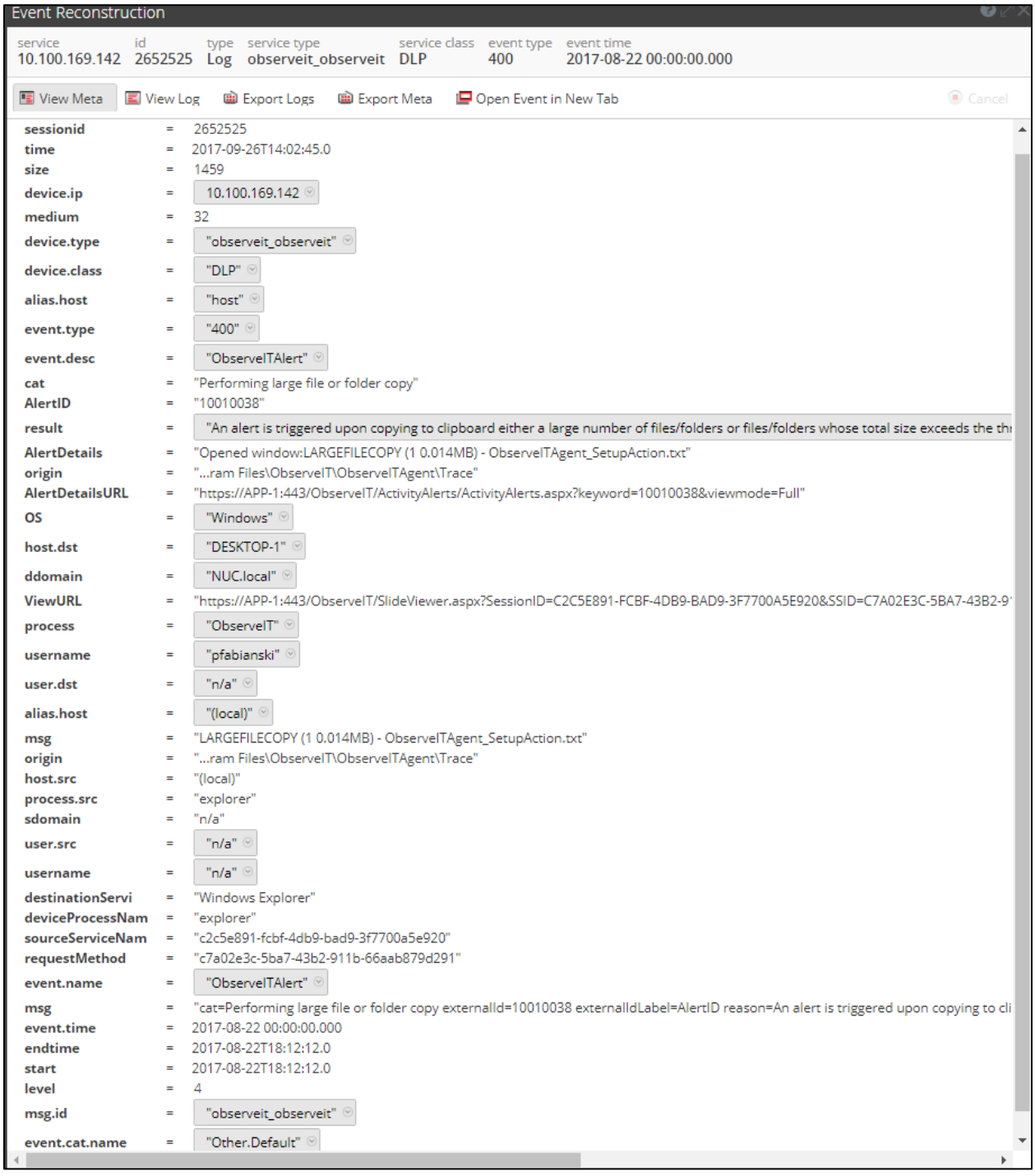
3. Copy and paste the entire section below into a new file or only the lines between the <mappings>…</mappings> if the Table-Map-Custom.xml file exists;

Example.

```
<?xml version="1.0" encoding="utf-8"?>
<!--
#
# ObserveIT DRP 9-25-2017
#
# attributes:
#      envisionName: The name of the column in the universal table
#      nwName:                    The name of the NetWitness meta field
#      format:                    Optional. The language key data type. See
LanguageManager. Defaults to "Text".
#      flags:              Optional. One of None|File|Duration|Transient.
Defaults to "None".
#      failureKey:         Optional. The name of the NW key to write data if
conversion fails. Defaults to system generated "parse.error" meta.
#      nullTokens:         Optional. The list of "null" tokens. Pipe separated.
Default is no null tokens.
-->
<mappings>

       <mapping envisionName="starttime" nwName="start" flags="None"
format="TimeT" envisionDisplayName="StartTime"/>
       <mapping envisionName="endtime" nwName="endtime" flags="None"
format="TimeT" envisionDisplayName="EndTime,rt,end"/>

       <mapping envisionName="cat" nwName="cat" flags="None"
envisionDisplayName="cat"/>
       <!--DRP 9-25-2017 nwName="hardware.id" to nwName="AlertID" -->
       <mapping envisionName="hardware_id" nwName="AlertID" flags="None"/>
       <mapping envisionName="result" nwName="result" flags="None"
envisionDisplayName="Result|Volume|Information|Reason|Succeed/Failed"/>
       <mapping envisionName="origin" nwName="origin" flags="None"/>
       <mapping envisionName="AlertDetails" nwName="AlertDetails" flags="None"/>
       <mapping envisionName="AlertDetails" nwName="AlertDetails" flags="None"/>
       <mapping envisionName="OS" nwName="OS" flags="None"/>
       <mapping envisionName="dhost" nwName="host.dst" flags="None"
envisionDisplayName="DestinationHostName"/>
       <mapping envisionName="ddomain" nwName="ddomain" flags="None"/>
       <mapping envisionName="AlertDetailsURL" nwName="AlertDetailsURL"
flags="None"/>
       <mapping envisionName="ViewURL" nwName="ViewURL" flags="None"/>
       <mapping envisionName="process" nwName="process" flags="None"
envisionDisplayName="Process"/>
       <mapping envisionName="uid" nwName="username" flags="None"
envisionDisplayName="UserID|UID|Uid" nullTokens="none|-"/>
       <mapping envisionName="logon_id" nwName="username" flags="None"
envisionDisplayName="LogonID" nullTokens="none|-"/>
       <mapping envisionName="hostname" nwName="alias.host" flags="None"
envisionDisplayName="SystemName|LogHostName|Host|ComputerName|HostName"/>
```

```
        <mapping envisionName="hostip" nwName="ip.addr" flags="None"
format="IPv4" envisionDisplayName="ComputerIP|IPAddress" failureKey="ipv6.addr"
nullTokens="(null)|-"/>
        <mapping envisionName="msg" nwName="msg" flags="None" format="Text"
envisionDisplayName="Message"/>
        <mapping envisionName="shost" nwName="host.src" flags="None"
envisionDisplayName="ForeignHostName|SourceHostName"/>
        <mapping envisionName="process_src" nwName="process.src" flags="None"
envisionDisplayName="SourceProcess"/>
        <mapping envisionName="saddr" nwName="ip.src" flags="None" format="IPv4"
envisionDisplayName="ServerAddress|SourceIPAddress|SourceAddress|Address|LocalA
ddress|ClientAddress" failureKey="ipv6.src" nullTokens="(null)|-"/>
        <mapping envisionName="sdomain" nwName="sdomain" flags="None"/>
        <mapping envisionName="c_username" nwName="user.src" flags="None"
envisionDisplayName="ClientUserName" nullTokens="none|-"/>
        <mapping envisionName="suid" nwName="suid" flags="None"/>
        <mapping envisionName="destinationServiceName"
nwName="destinationServiceName" flags="None"/>
        <mapping envisionName="deviceProcessName" nwName="deviceProcessName"
flags="None"/>
        <mapping envisionName="sourceServiceName" nwName="sourceServiceName"
flags="None"/>
        <mapping envisionName="requestMethod" nwName="requestMethod"
flags="None"/>

</mappings>
```

ObserveIT collection example within RSA NetWitness Investigator:



Certification Checklist for RSA NetWitness

Date Tested: November 2, 2017

Certification Environment		
Product Name	**Version Information**	**Operating System**
RSA NetWitness	10.6.4	Virtual Appliance
ObserveIT	7.1.0	Proprietary

NetWitness Test Case	Result
Device Administration	
Partner's device name appears in Device Parsers Configuration	✓
Device can be enabled from Device Parsers Configuration	✓
Device can be disabled from Device Parsers Configuration	✓
Device can be removed from Device Parsers Configuration	✓
Investigation	
Device name displays properly from Device Type	✓
Displays Meta Data properly within Investigator	✓

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

Known Issues

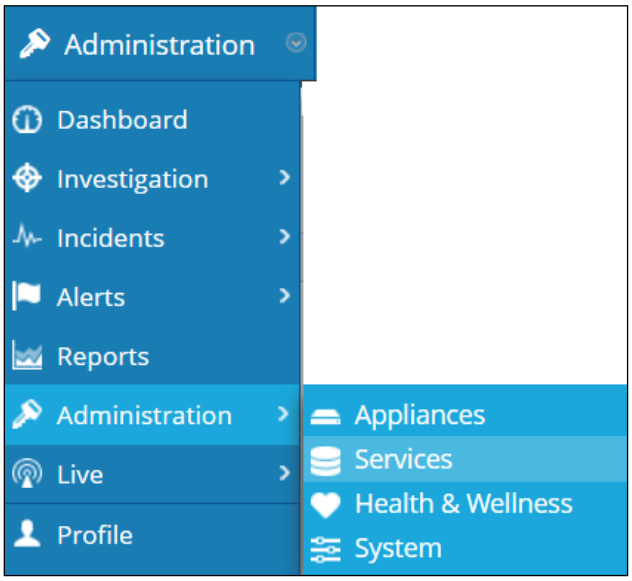
- In some cases the **cs4Label** for the **cs4** key does not appear within an ObserveIT event string, as a result the event is not collected or displayed within RSA NetWitness Investigator.
- The event contains the key **externalIdLabel** which is not required since it is not a custom key field. This key has no impact to the collection of the ObserveIT event source.
- The **origin** key appears twice in some events containing the same key value. The second appearance of the key has no impact on the collection of the ObserveIT event source.
- The **msg** key will appear twice within the NetWitness Event recreation, once containing the ObserveIT msg key contents and a second time containing the entire event. RSA NetWitness utilizes the msg key to parse the entire event string. RSA does not have a workaround for this issue.
- Key names greater than 16 characters will be truncated due to the key name size limitation within RSA NetWitness, ex. **deviceProcessName** is **deviceProcessNam**. The truncation of the key name does not impact RSA NetWitness event collection. A bug has been opened with RSA Product Management to address this issue.

Appendix

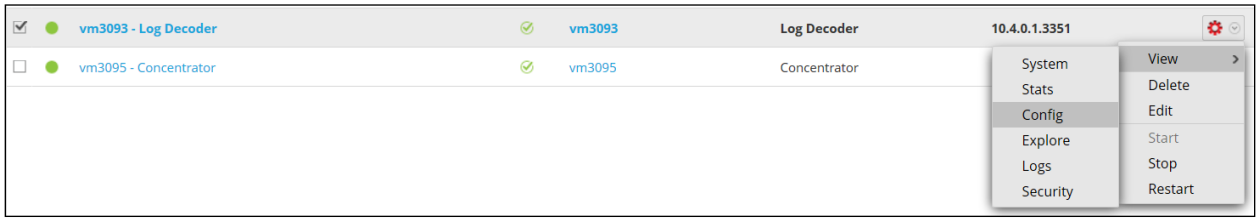
NetWitness Disable the Common Event Format Parser

To disable the NetWitness Common Event Format Parser and not delete it perform the following:

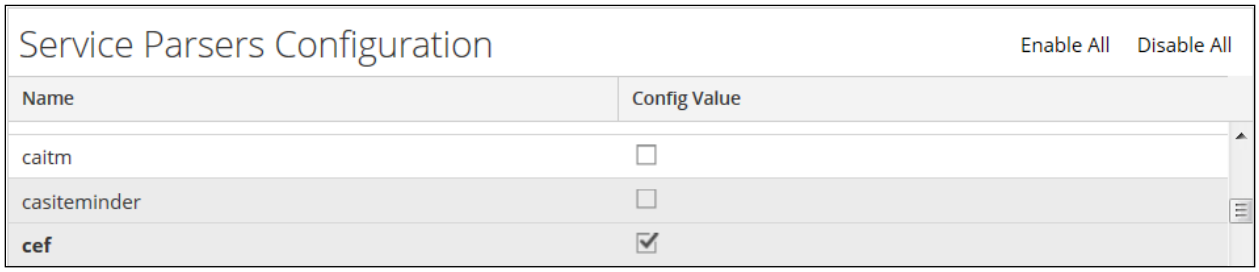
1. Select the NetWitness **Administration > Services menu**.



2. Select the Log Decoder, then select **View > Config.**



3. From the **Service Parses Configuration** window, scroll down to the CEF parser and uncheck the Config Value checkbox.

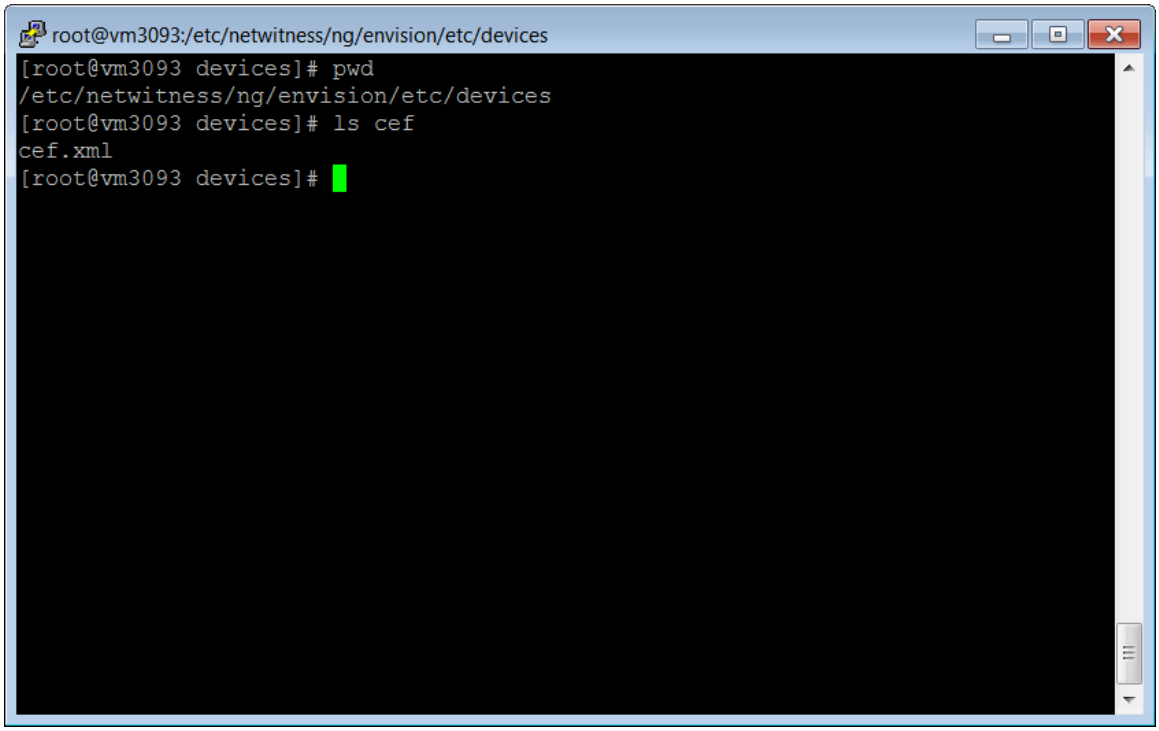


4. Click **Apply** to save settings.

NetWitness Remove Device Parser

To remove the NetWitness Integration Package files from the environment, perform the following:

1. Connect to the NetWitness Log Decoder/Collector Server using SSH and open the **/etc/netwitness/ng/envision/etc/devices** folder.



2. Search for and delete the CEF folder and its contents.