RSA® NETWITNESS®
Logs
Implementation Guide

Acalvio Technologies
Acalvio ShadowPlex

Daniel R. Pintal, RSA Partner Engineering
Last Modified: July 18, 2017

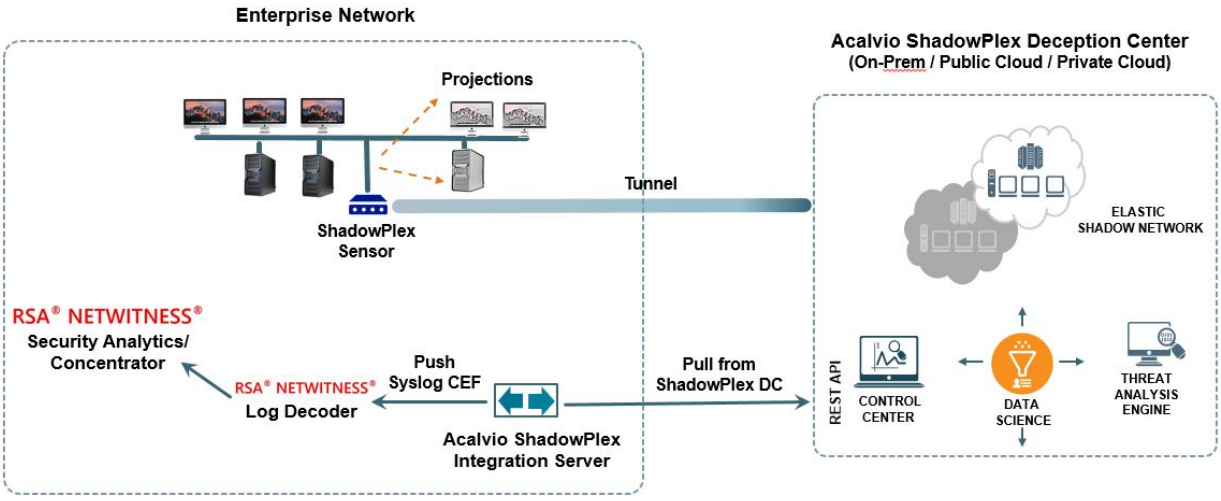
RSA READY

Solution Summary

Acalvio ShadowPlex integrates with RSA NetWitness to provide an Advanced Threat Defense solution which allows DevOps to detect, engage and respond to malicious activity inside the perimeter.

When an adversary traversing the network looking for network resources to exploit attempts to connect to a deception shadow network resource they trigger an event which is sent to RSA NetWitness. This allows the SOC to actively engage the adversary.

RSA NetWitness Features	
Acalvio ShadowPlex Version 2017.07	
Integration package name	Common Event Format
Device display name within Security Analytics	acalvio_shadowplex
Event source class	Advanced Threat Detection
Collection method	Syslog



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
07/18/2017	Initial support for Acalvio.

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 Acalvio ShadowPlex 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 Acalvio 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 Acalvio ShadowPlex is properly configured and secured before deploying to a production environment. For more information, please refer to the Acalvio ShadowPlex documentation or website.

Acalvio ShadowPlex Configuration

After the Acalvio ShadowPlex Integration Server is installed, it needs to be configured to generate CEF formatted syslog output to RSA NetWitness log decoder. No additional software needs to be installed to configure the ShadowPlex components for the integration.

Configure Acalvio ShadowPlex Integration Server

The Integration Server installation directory contains the configuration file called "config.json". A copy of the file is below.

```
{
    "event_source":{
        "adc_api_host" : "10.10.1.195",
        "adc_api_port" : 8443,
        "api_user_name": "userid",
        "api_user_password": "password"
    },
    "syslog":{
        "host" : "10.10.1.19",
        "port" : 514,
        "protocol" : "UDP",
        "local_host_name" : "ADC",
        "app_name" : "ShadowPlex"
    },
    "fetch_options":{
        "fetching_interval" : 10,
        "event_list" : "'ACAL_EVENT_NVCTL_TRIP'"
    }
}
```

We describe only the fields that need to be changed in the configuration file. The configuration has three parts. The first part specifies the location of the Acalvio ShadowPlex Deception Center.

```
"adc_api_host" : IP Address of the ShadowPlex Deception Center
"api_user_name": The user name for authentication with ShadowPlex Deception Center
"api_user_password": Password for the above user name
```

The second part describes where the CEF formatted syslog needs to be forwarded to. This is the IP address of the RSA NetWitness log decoder. The rest of the fields should remain the same.

```
"host" : IP Address of the RSA NetWitness Log Decoder VM
```

The third part configures the polling interval and the type of events. In this version, only the `'ACAL_EVENT_NVCTL_TRIP'` events are sent to the NetWitness log decoder.

```
"fetching_interval" : Polling interval in seconds
```

After the configuration is done, the integration server needs to be restarted.

On Linux:

```
systemctl restart sc.service
```

On Windows:

```
C:\shadow-connector> sc-win-service-restart.bat

Or via standard Windows service management tool: services.msc
```

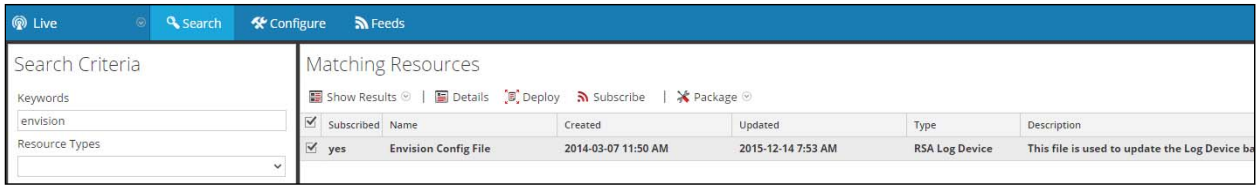
RSA NetWitness Configuration

Deploy the enVision Config File

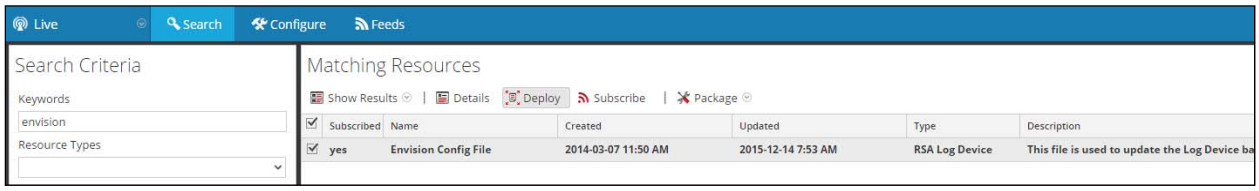
In order to use the RSA Common Event Format, you must first deploy the *enVision Config File* from the **NetWitness Live** module. Log into NetWitness and perform the following actions:

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

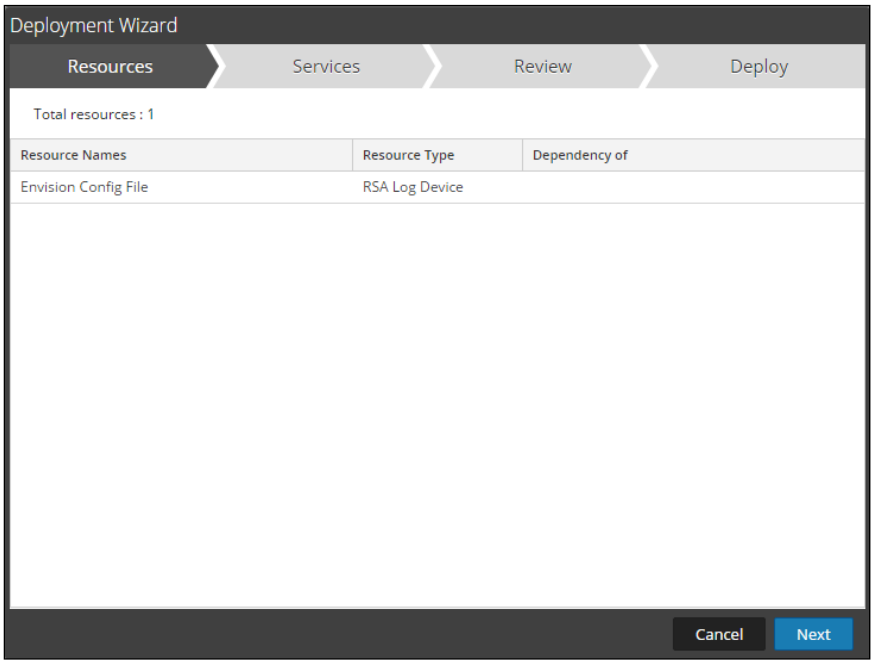
1. From the Security Analytics menu, select **Live > Search**.
2. In the keywords field, enter: **enVision**.
3. Security Analytics will display the **Envision Config File** in Matching Resources.
4. Select the checkbox next to **Envision Config File**.



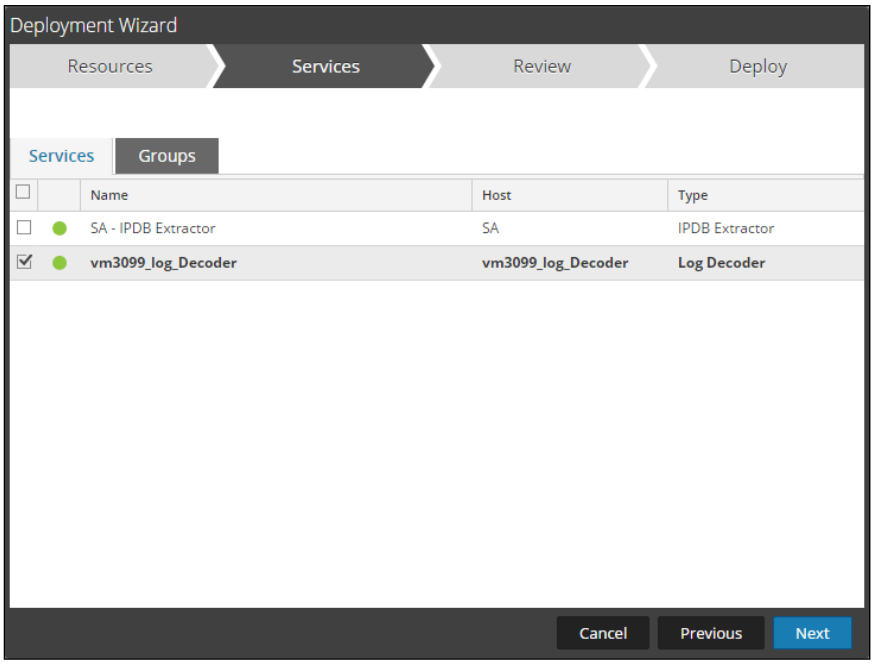
5. Click **Deploy** in the menu bar.



6. Select **Next**.

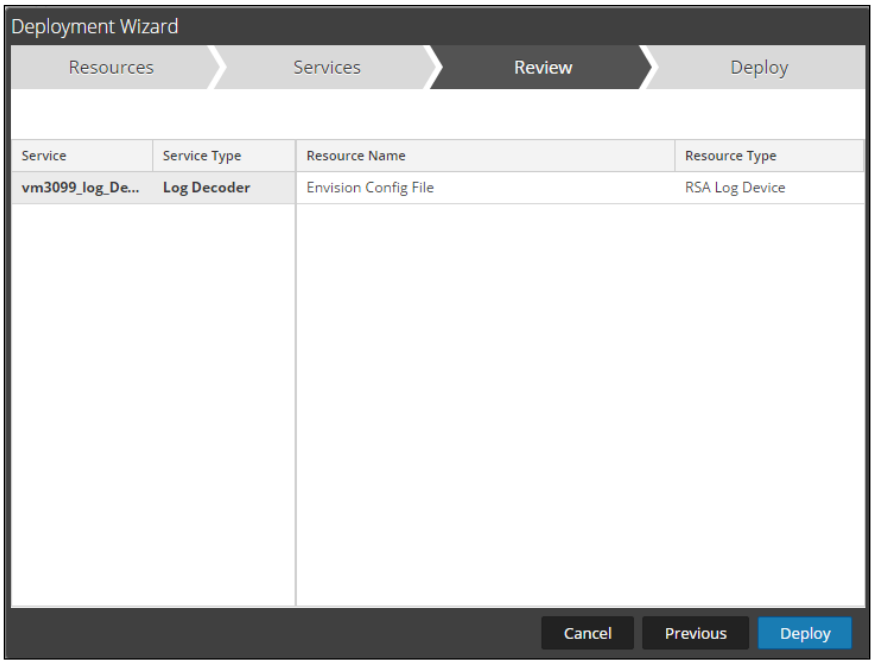


7. Select the **Log Decoder** and select **Next**.

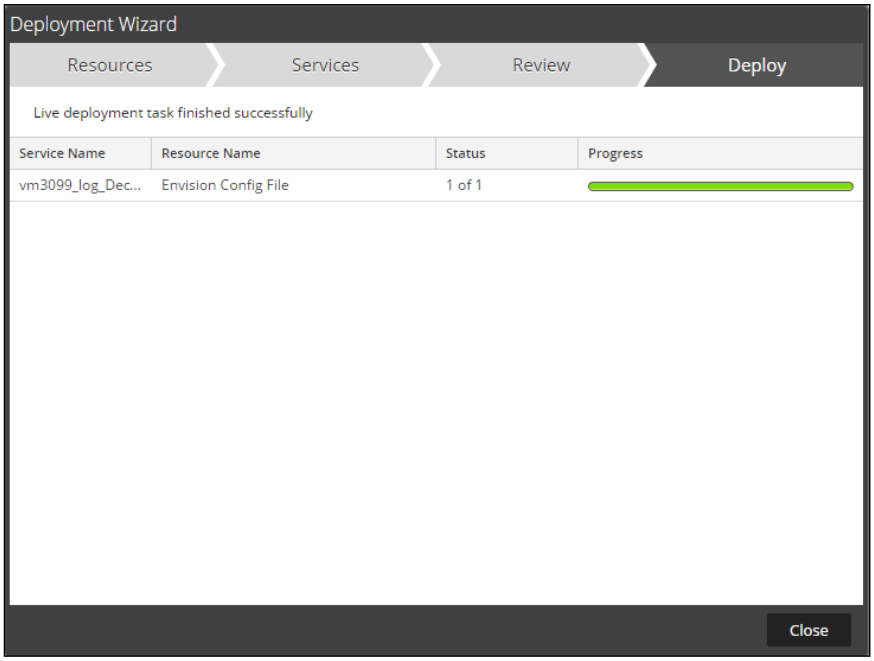


Important: In an environment with multiple Log Decoders, deploy the Envision Config File to each Log Decoder in your network.

8. Select **Deploy**.



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

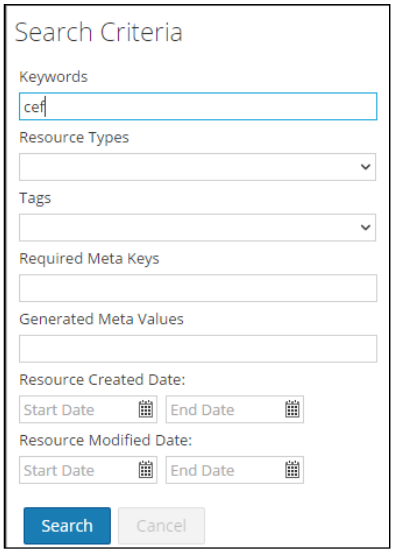


Deploy the Common Event Format

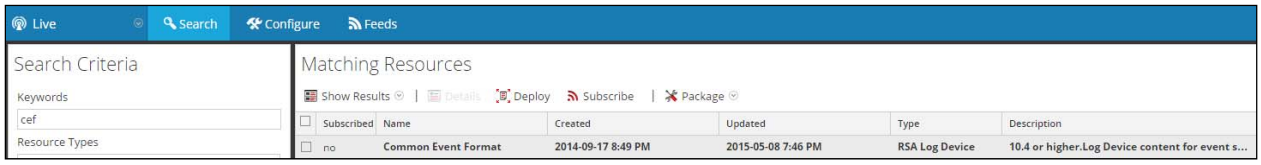
Next, you will need to deploy the *Common Event Format file* from the **NetWitness Live** module. Log into NetWitness and perform the following actions:

> **Important: Using this procedure will overwrite an existing cef.xml.**
>
> **It is recommended that NetWitness Live updates be disabled for cef.xml to prevent overwriting the customizations detailed in this guide.**

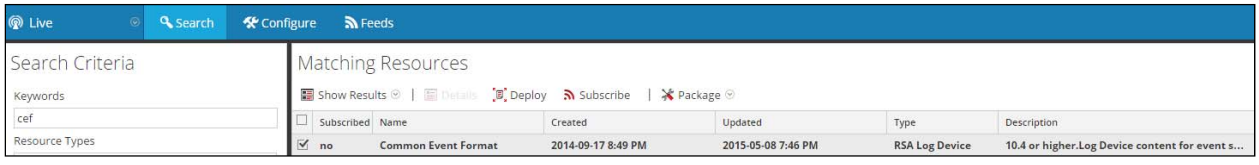
1. From the NetWitness menu, select **Live > Search**.
2. In the keywords field, enter: **CEF**



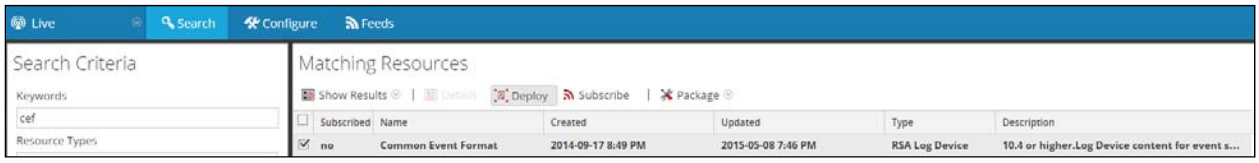
3. RSA NetWitness will display the **Common Event Format** in Matching Resources.



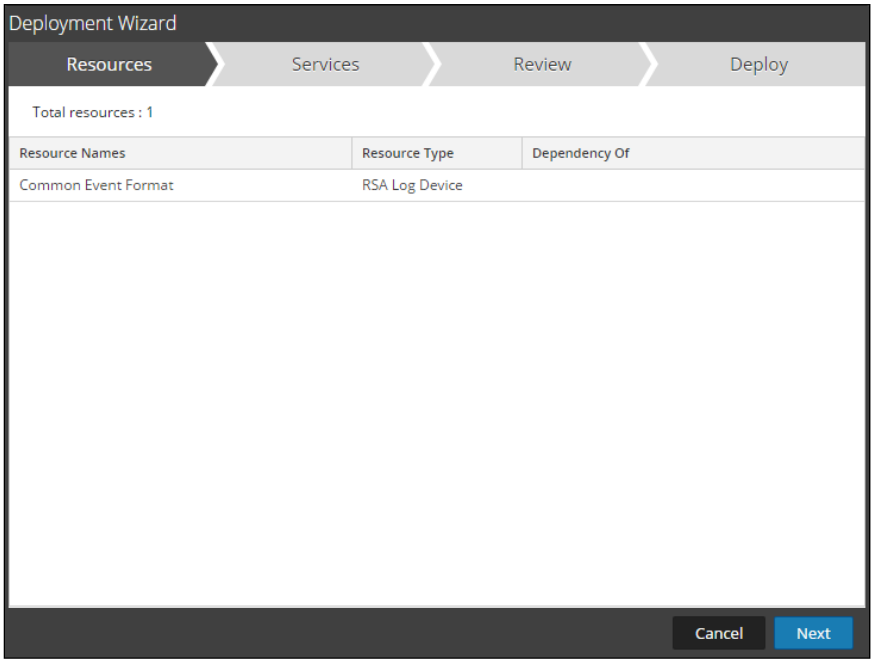
4. Select the checkbox next to **Common Event Format**.



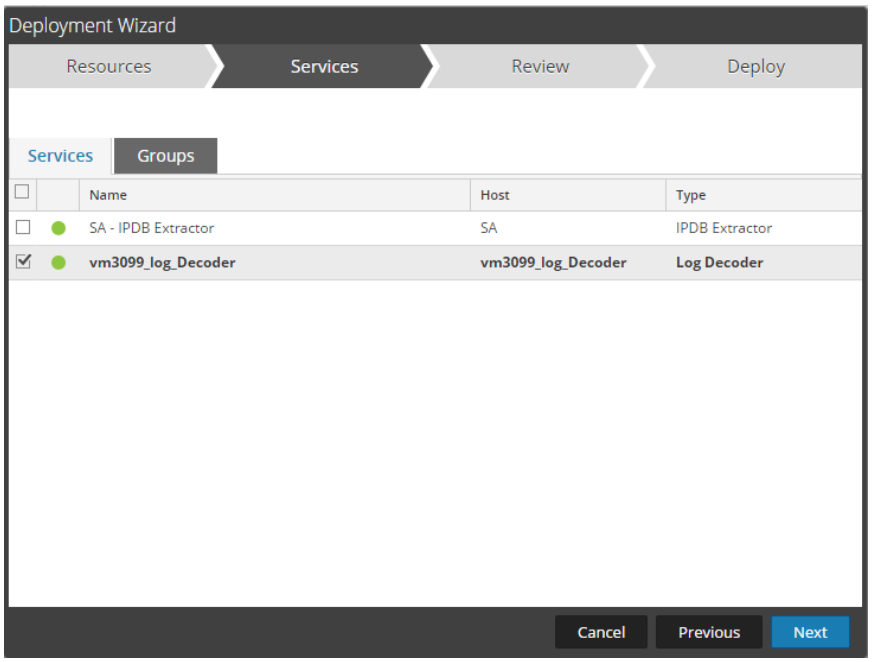
5. Click **Deploy** in the menu bar.



6. Select **Next**.

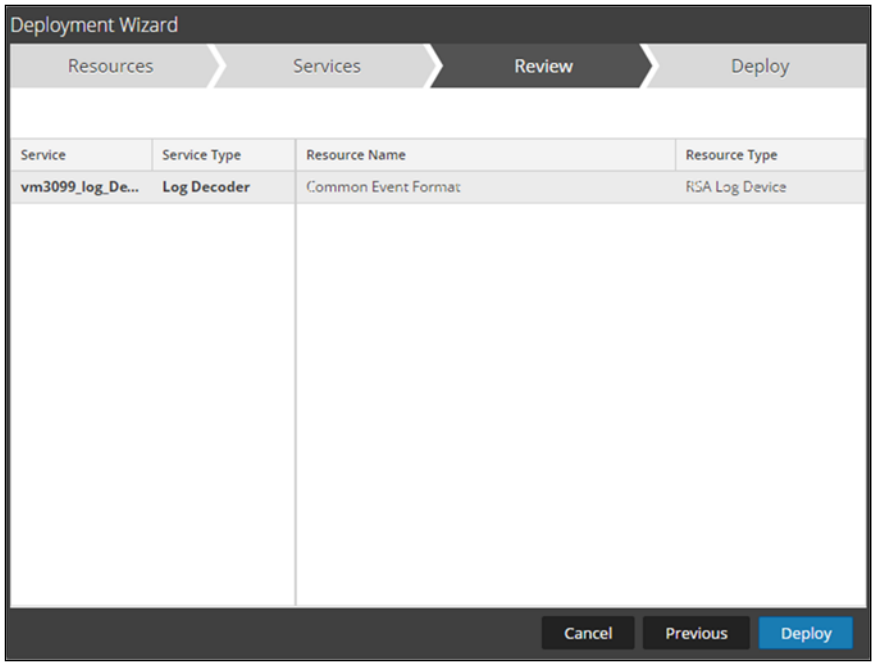


7. Select the **Log Decoder** and Select **Next**.

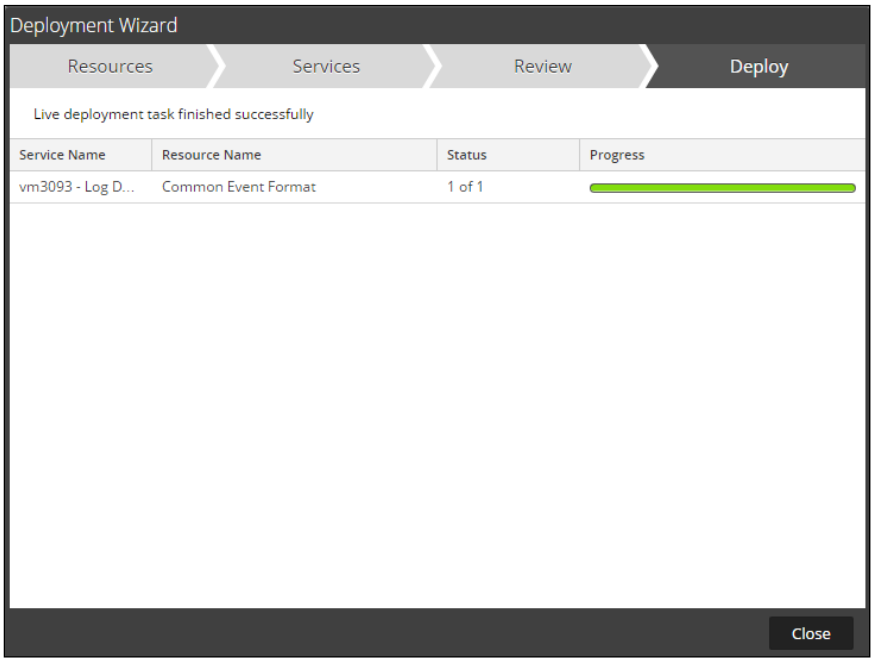


Important: In an environment with multiple Log Decoders, deploy the Common Event Format to each Log Decoder in your network.

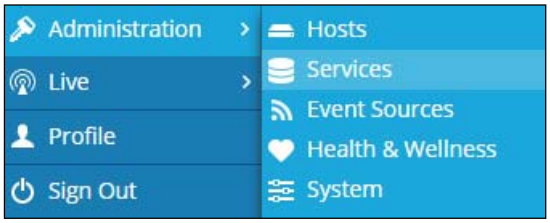
8. Select **Deploy**.



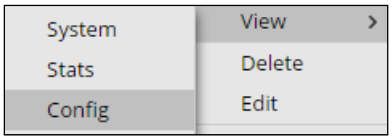
9. Select **Close**, to complete the deployment of the Common Event Format.



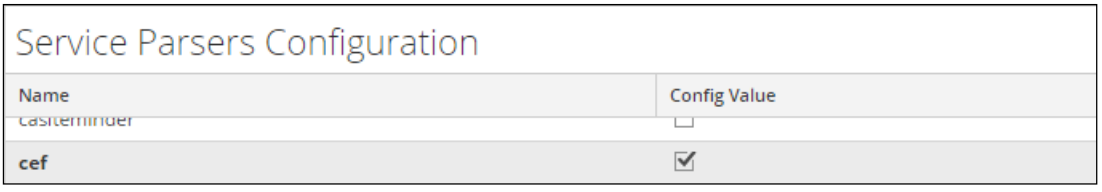
10. Ensure that the CEF Parser is enabled on the Log Decoder(s) by selecting **Administration> Services** from the NetWitness Dashboard.



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 support custom fields

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 xml text below into the file after the /> of the preceding <MESSAGE and xml text;

```
<MESSAGE
                level="4"
                parse="1"
                parsedefvalue="1"
                tableid="74"
                id1="acalvio_shadowplex"
                id2="acalvio_shadowplex"
                eventcategory="1612000000"

        content="&lt;@event_name:*HDR(event_description)&gt;&lt;@msg:*PARMVAL($MS
G)&gt;&lt;@starttime:*EVNTTIME($MSG,'%B %D %W
%H:%T:%S',param_starttime)&gt;&lt;@endtime:*EVNTTIME($MSG,'%B %D %W
%H:%T:%S',param_endtime)&gt;&lt;param_starttime&gt;&lt;param_endtime&gt;&lt;msg
hold&gt;" />
```

Example.

```
<MESSAGE
                level="4"
                parse="1"
                parsedefvalue="1"
                tableid="74"
                id1="azureaudit"
                id2="azureaudit"
                eventcategory="1612000000"

        content="&lt;@event_name:*HDR(event_description)&gt;&lt;@msg:*PARMVAL($MS
G)&gt;&lt;@event_time:*EVNTTIME($MSG,'%W-%G-
%FT%ZZ',param_event_time)&gt;&lt;param_event_time&gt;&lt;msghold&gt;" />

<MESSAGE
                level="4"
                parse="1"
                parsedefvalue="1"
                tableid="74"
                id1="acalvio_shadowplex"
                id2="acalvio_shadowplex"
                eventcategory="1612000000"

        content="&lt;@event_name:*HDR(event_description)&gt;&lt;@msg:*PARMVAL($MS
G)&gt;&lt;@starttime:*EVNTTIME($MSG,'%B %D %W
%H:%T:%S',param_starttime)&gt;&lt;@endtime:*EVNTTIME($MSG,'%B %D %W
%H:%T:%S',param_endtime)&gt;&lt;param_starttime&gt;&lt;param_endtime&gt;&lt;msg
hold&gt;" />
```

3. Next, locate the following line;

```
<ExtensionKey cefName="end" metaName="param_endtime"/>
```

4. Copy the following line and insert it below the preceding line;

```
<ExtensionKey cefName="rt" metaName="param_endtime"/>
```

Example.

```
<ExtensionKey cefName="end" metaName="param_endtime"/>
<ExtensionKey cefName="rt" metaName="param_endtime"/>
```

5. Next, locate the following line;

```
<ExtensionKey cefName="cs1" metaName="cs_fld" >
```

6. Copy the following line and insert it after the last device2meta but before </ExtensionKey> in the cs1 section;

```
<device2meta device="acalvio_shadowplex" metaName="flipType" label="flipType"/>
```

Example.

```
<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="acalvio_shadowplex" metaName="flipType"
      label="flipType"/>
</ExtensionKey>
```

7. Next, locate the following line;

```
<ExtensionKey cefName="cs2" metaName="cs_fld">
```

8. Copy the following line and insert it after the last device2meta but before </ExtensionKey> in the cs2 section;

```
<device2meta device="acalvio_shadowplex" metaName="subnet" label="subnet"/>
```

Example.

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

9. Next, locate the following line;

```
<ExtensionKey cefName="cs3" metaName="cs_fld">
```

10. Copy the following line and insert it after the last device2meta but before </ExtensionKey> in the cs3 section;

```
<device2meta device="acalvio_shadowplex" metaName="subnetName"
label="subnetName"/>
```

Example.

```
<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="acalvio_shadowplex" metaName="subnetName"
label="subnetName"/>
</ExtensionKey>
```

11. Next, locate the following line;

```
<ExtensionKey cefName="cs4" metaName="cs_fld">
```

12. Copy the following line and insert it after the last device2meta but before </ExtensionKey> in the cs4 section;

```
<device2meta device="acalvio_shadowplex" metaName="protocolType"
label="protocolType"/>
```

Example.

```
<ExtensionKey cefName="cs4" metaName="cs_fld">
        <device2meta device="mcafeewg" metaName="info" label="URL Categories"/>
        <device2meta device="acalvio_shadowplex" metaName="protocolType"
label="protocolType"/>
</ExtensionKey>
```

13. Next, locate the following line;

```
<ExtensionKey cefName="cn1" metaName="cs_fld">
```

14. Copy the following line and insert it after the last device2meta but before </ExtensionKey> in the cn1 section;

```
<device2meta device="acalvio_shadowplex" metaName="eventType"
label="eventType"/>
```

Example.

```
<ExtensionKey cefName="cn1" metaName="cn_fld">
        <device2meta device="trendmicrods" metaName="hostid" label="Host ID"/>
        <device2meta device="trendmicrodsa" metaName="hostid" label="Host ID"/>
        <device2meta device="mcafeewg" metaName="result" label="Block Reason"/>
        <device2meta device="acalvio_shadowplex" metaName="eventType"
        label="eventType"/>
</ExtensionKey>
```

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"?>
<!--
# 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="hardware_id" nwName="eventid" flags="None"/>
    <mapping envisionName="application" nwName="app" flags="None"/>
    <mapping envisionName="flipType" nwName="flipType" flags="None"
format="Text" />
    <mapping envisionName="subnet" nwName="subnet" flags="None" format="Text" />
    <mapping envisionName="protocolType" nwName="protocolType" flags="None"
format="Text" />
    <mapping envisionName="subnetName" nwName="subnetName" flags="None"
format="Text" />
    <mapping envisionName="eventType" nwName="eventType" flags="None"
format="Text" />
    <mapping envisionName="starttime" nwName="starttime" flags="None"
format="TimeT" envisionDisplayName="StartTime" />
    <mapping envisionName="ddomain" nwName="ddomain" flags="None" />
    <mapping envisionName="dmacaddr" nwName="eth.dst" flags="None" format="MAC"
envisionDisplayName="DestMacAddress|DestinationMacAddress" />
    <mapping envisionName="endtime" nwName="endtime" flags="None" format="TimeT"
envisionDisplayName="EndTime,rt,end" />

</mappings>
```

NetWitness Collection Example:

Certification Checklist for RSA NetWitness

Date Tested: July 14, 2017

Certification Environment		
Product Name	**Version Information**	**Operating System**
RSA NetWitness	10.6.3	Virtual Appliance
Acalvio ShadowPlex	2017.07	Physical/Virtual Appliance

Security Analytics Test Case	Result
Investigation	
Device name displays properly from Device Type	✓
Displays Meta Data properly within Investigator	✓

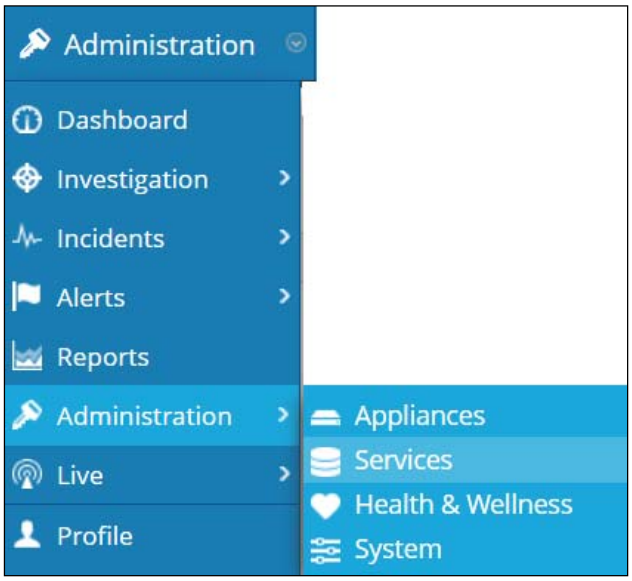
✓ = Pass ✗ = Fail N/A = Non-Available Function

Appendix

NetWitness Disable the Common Event Format Parser

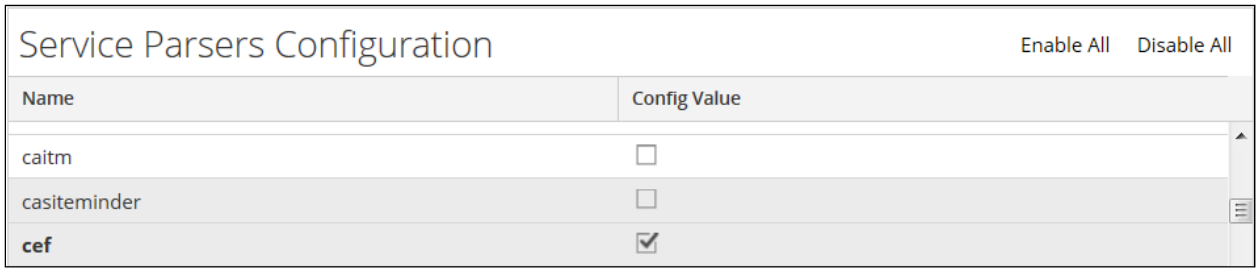
To disable the Security Analytics Common Event Format Parser and not delete it perform the following:

1. Select the Security Analytics **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.

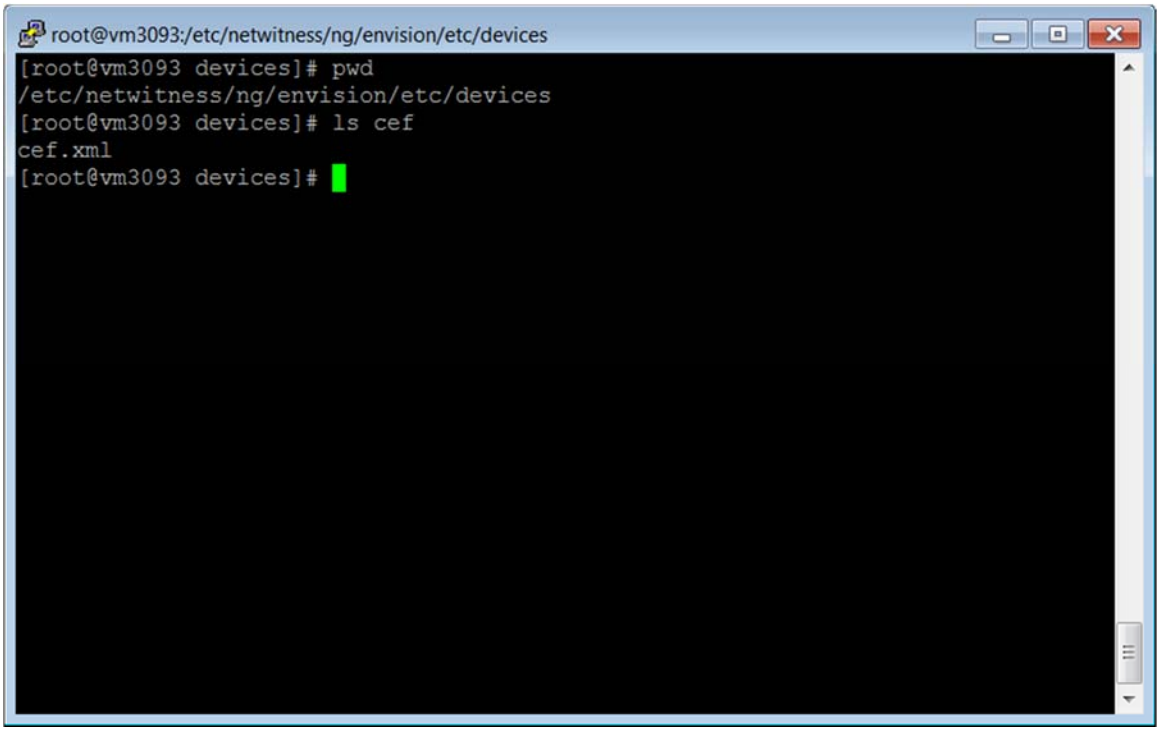


4. Click **Apply** to save settings.

NetWitness Remove Device Parser

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

1. Connect to the Security Analytics 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.