RSA Ready Implementation Guide for RSA Security Analytics

BeyondTrust Software PowerBroker Servers for UNIX/Linux

Daniel R. Pintal, RSA Partner Engineering
Last Modified: February 16, 2016

RSA READY

Solution Summary

PowerBroker Servers for UNIX/Linux has an extensive event log recording capability that allows the customers to collect detailed information when a command is executed with escalated privileges through PowerBroker interface. These event log records can be used to generate reports for monitoring and auditing purposes. PBUL has the ability to write these event log records in the system syslog files. The integration of PBUL with RSA Security Analytics allows our joint customers to centralized log-management as part of their broader management ecosystem.

RSA Security Analytics Features	
PowerBroker Servers for UNIX/Linux 8.0	
Integration package name	Beyondtrustpe.envision
Device display name within Security Analytics	beyondtrustpe
Event source class	Access Control
Collection method	Syslog

RSA Security Analytics (SA) Community

The RSA Security Analytics (SA) Community is an online forum for customers and partners to exchange technical information and best practices with each other. The forum also contains the location to download the SA Integration Package for this guide. All Security Analytics customers and partners are invited to register and participate in the **RSA Security Analytics Community**.

Once you have downloaded the SA Integration Package, the next steps are to deploy this on all log decoders. For steps to disable or remove the Security Analytics Integration Package, please refer to the **Appendix** of this Guide.

The RSA Security Analytics package consists of the following files:

Filename	File Function
Beyondtrustpe.envision	SA package deployed to parse events from device integrations.
Beyondtrustpemsg.xml	A copy of the device xml contained within the SA package.
table-map-custom.xml	Enables Security Analytics variables disabled by default.

Release Notes

Release Date	What's New In This Release
12/02/2013	Initial support for BeyondTrust Software PowerBroker Servers for UNIX/Linux
2/12/2016	SA 10.5 support

RSA Security Analytics Configuration

Before You Begin

This section provides instructions for configuring the BeyondTrust Software PowerBroker Servers for UNIX/Linux with RSA Security Analytics. 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 BeyondTrust Software PowerBroker Servers for UNIX/Linux components must be installed and working prior to the integration. Perform the necessary tests to confirm that this is true before proceeding.

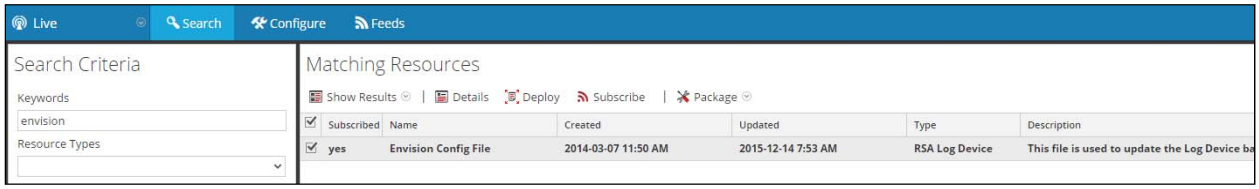
! Important: The configuration shown in this Implementation Guide is for example and testing purposes only. It is not intended to be the optimal setup for the device. It is recommended that customers make sure BeyondTrust Software PowerBroker Servers for UNIX/Linux is properly configured and secured before deploying to a production environment. For more information, please refer to the BeyondTrust Software PowerBroker Servers for UNIX/Linux documentation or website.

Deploy the enVision Config File

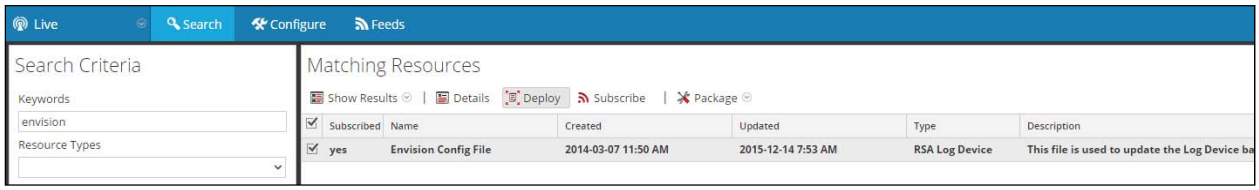
In order to use RSA Partner created content, you must first deploy the *Envision Config File* from the **Security Analytics Live** module. Log into Security Analytics and perform the following actions:

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

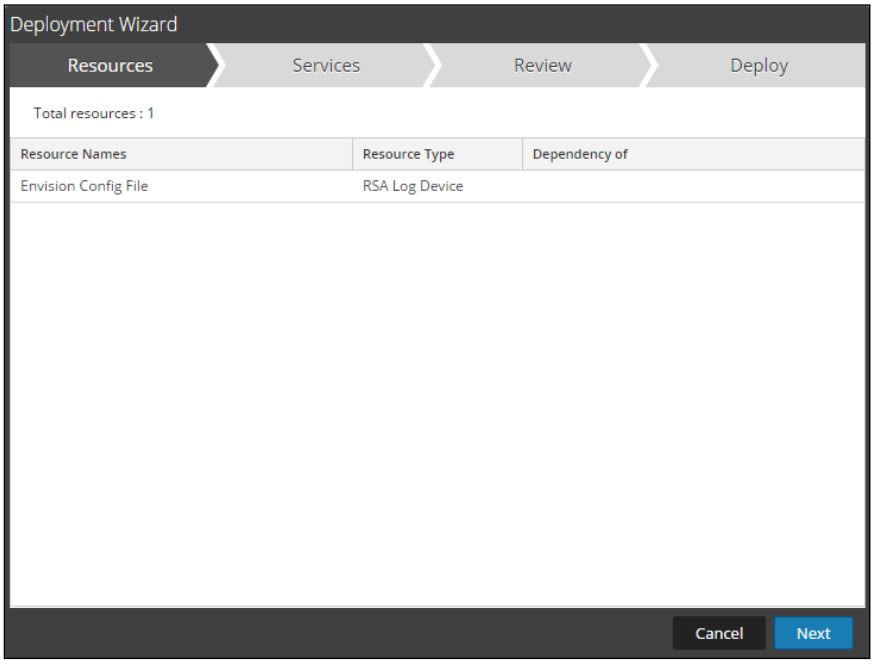
1. From the Security Analytics menu, select **Live > Search**.
2. In the keywords field, enter: **Envision**.
3. Security Analytics will display the **Envision Config File** in Matching Resources.
4. Select the checkbox next to **Envision Config File**.



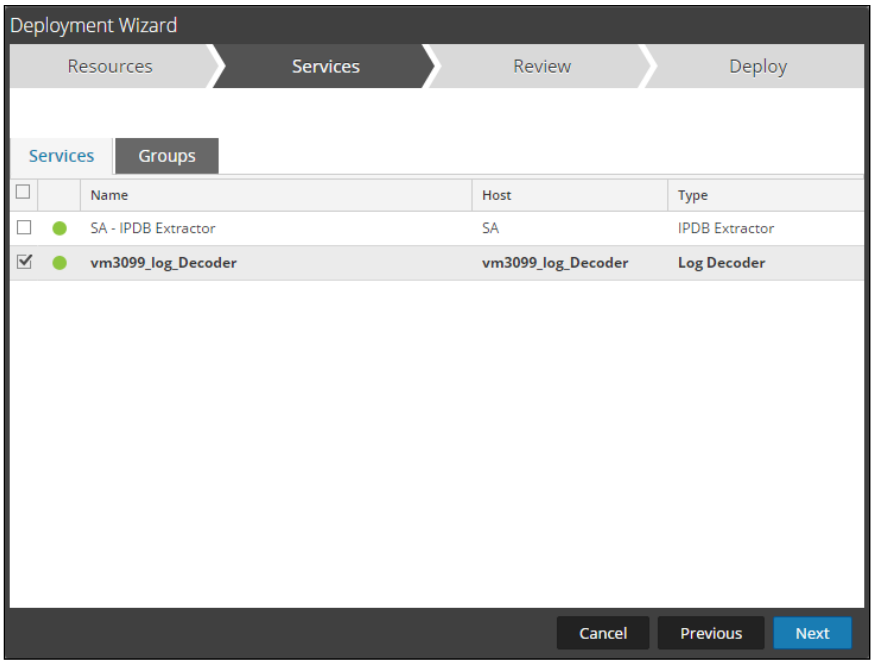
5. Click **Deploy** in the menu bar.



6. Select **Next**.

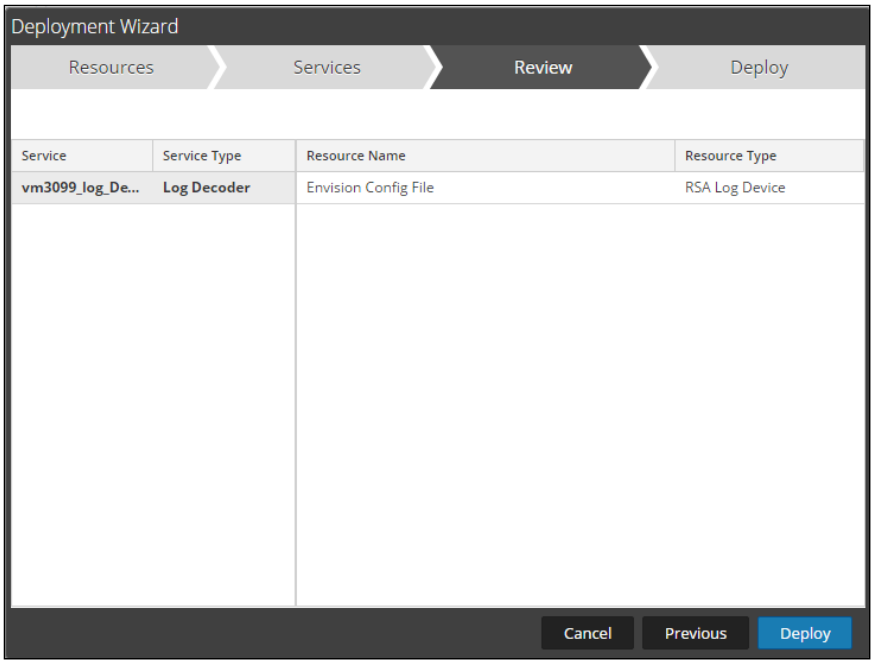


7. Select the **Log Decoder** and select **Next**.

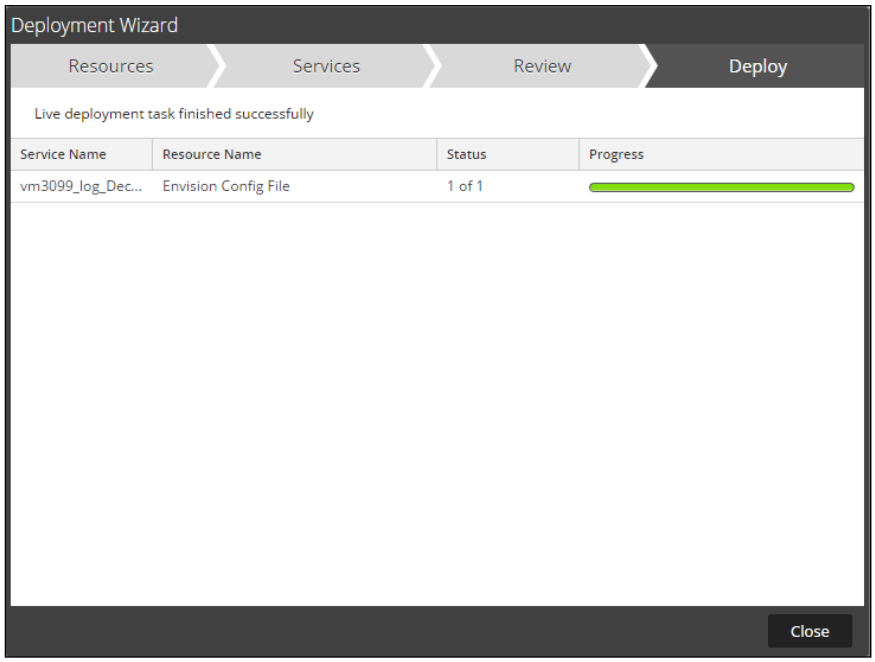


Important: In an environment with multiple Log Decoders, deploy the Envision Config File to each Log Decoder in your network.

8. Select **Deploy**.



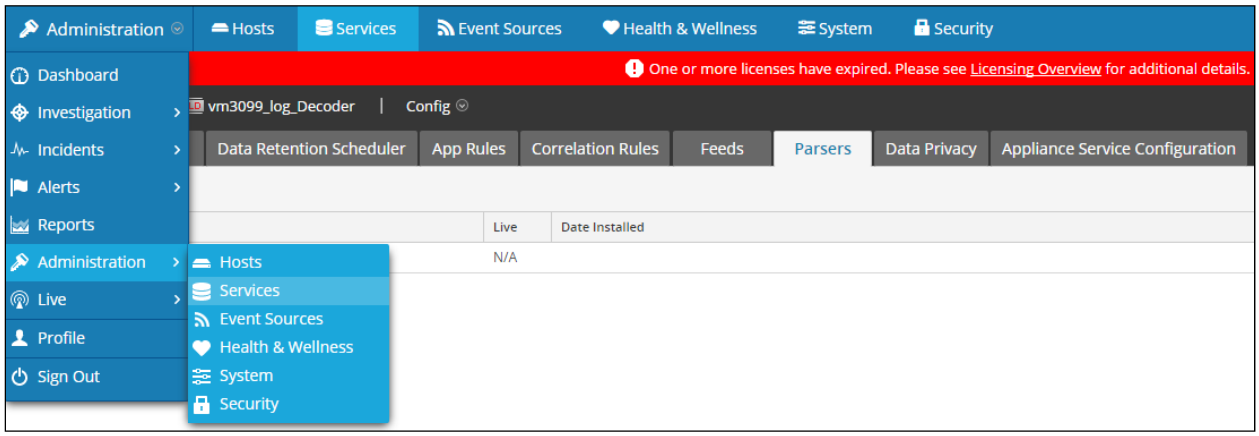
9. Select **Close**, to complete the deployment of the Envision Config file.



Deploy the Security Analytics Integration Package

After completing the previous section, ***Deploy the enVision Config File***, you can now deploy the Security Analytics Integration Package. Download the appropriate RSA Partner Integration Package, then log into Security Analytics to perform the following actions:

1. From the Security Analytics menu, select **Administration > Services.**



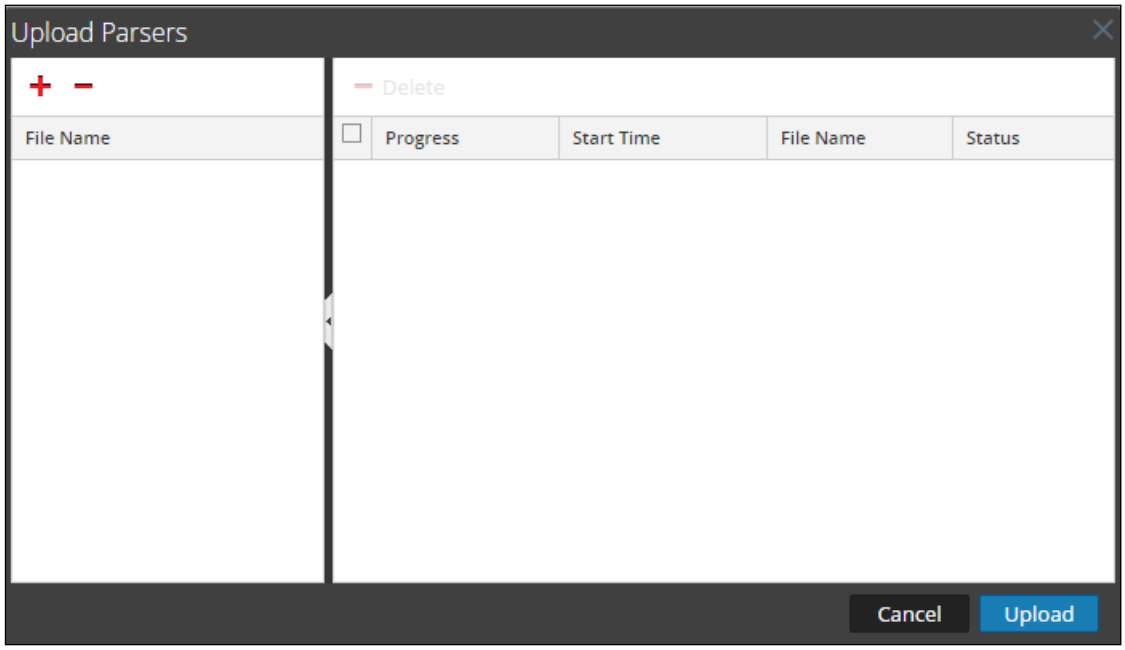
2. Select your Log Decoder from the list, select **View > Config**.

Important: In an environment with multiple Log Decoders, repeat on the deployment of the RSA Partner Integration Package on each Log Decoder.

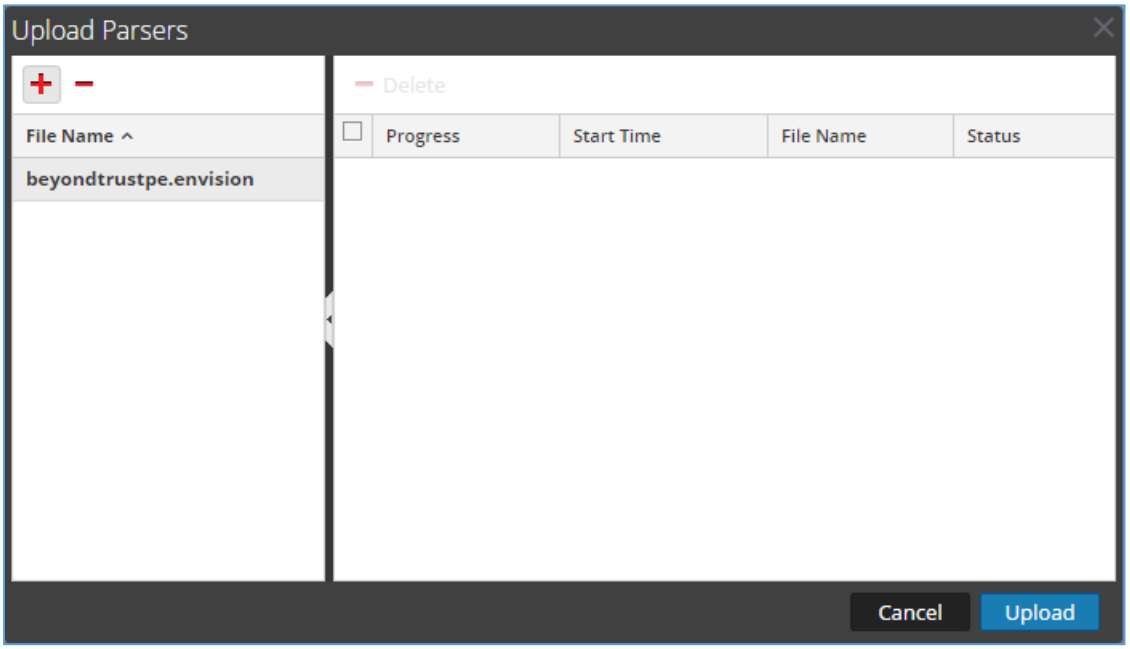
3. Next, select the **Parsers** tab and click the **Upload** button.

4. From the *Upload Parsers* window, click the + **Add** button and select the *.envision* file.

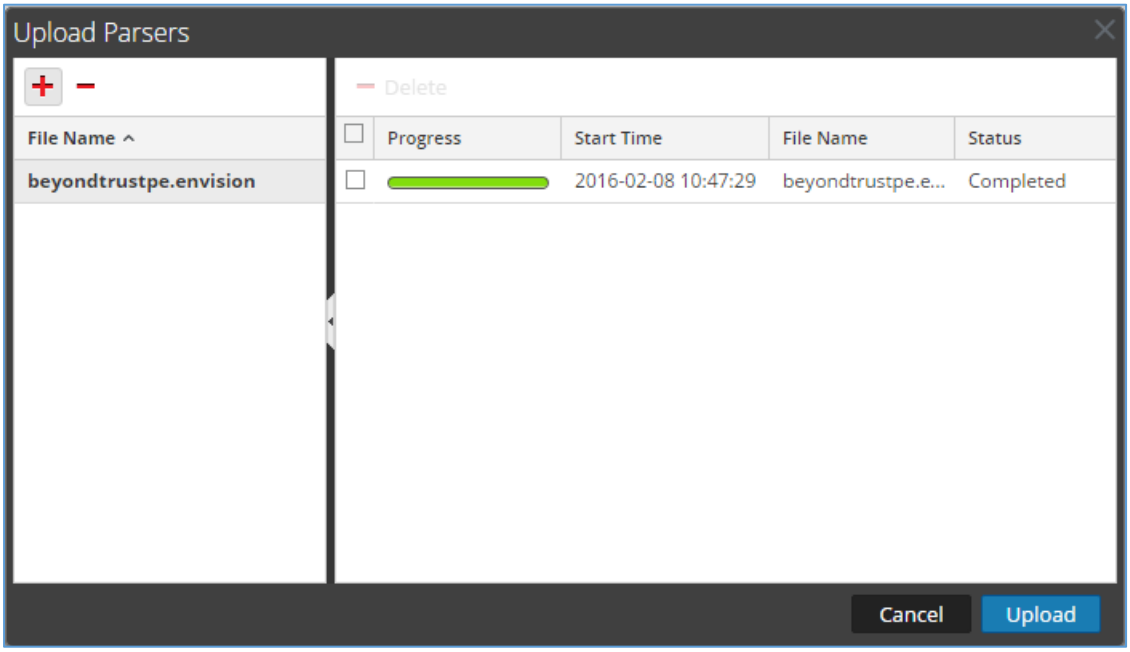
Important: The .envision file is contained within the .zip file downloaded from the RSA Community.



5. Under the file name column, select the integration package name and click **Upload**.



6. Upon completion of the upload click **Cancel**.



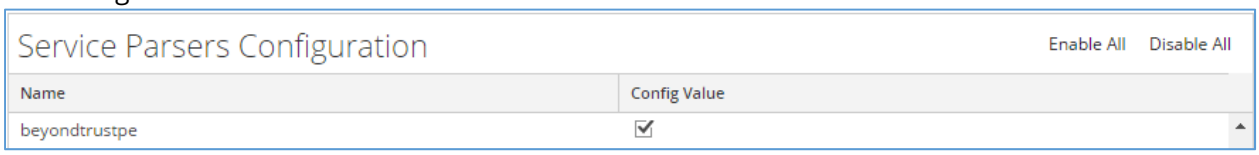
7. Connect to the Security Analytics Log Decoder Server using WinSCP. Copy the table-map-custom.xml file from the contents of the .zip file to the /etc/netwitness/ng/envision/etc folder. If the table-map-custom.xml file already exists on the log decoder(s), enter only the contents between the <mappings>…</mappings>.

```
<mappings>

        <mapping envisionName="timezone" nwName="timezone" flags="None"/>
        <mapping envisionName="sessionid" nwName="log.session.id" flags="None"/>
        <mapping envisionName="directory" nwName="directory" flags="None" envisionDisplayName="Directory|WorkingDirectory"/>
        <mapping envisionName="event_state" nwName="event.state" flags="None"/>
        <mapping envisionName="info" nwName="index" flags="None"/>
        <mapping envisionName="policy_value" nwName="policy.value" flags="None"/>

</mappings>
```

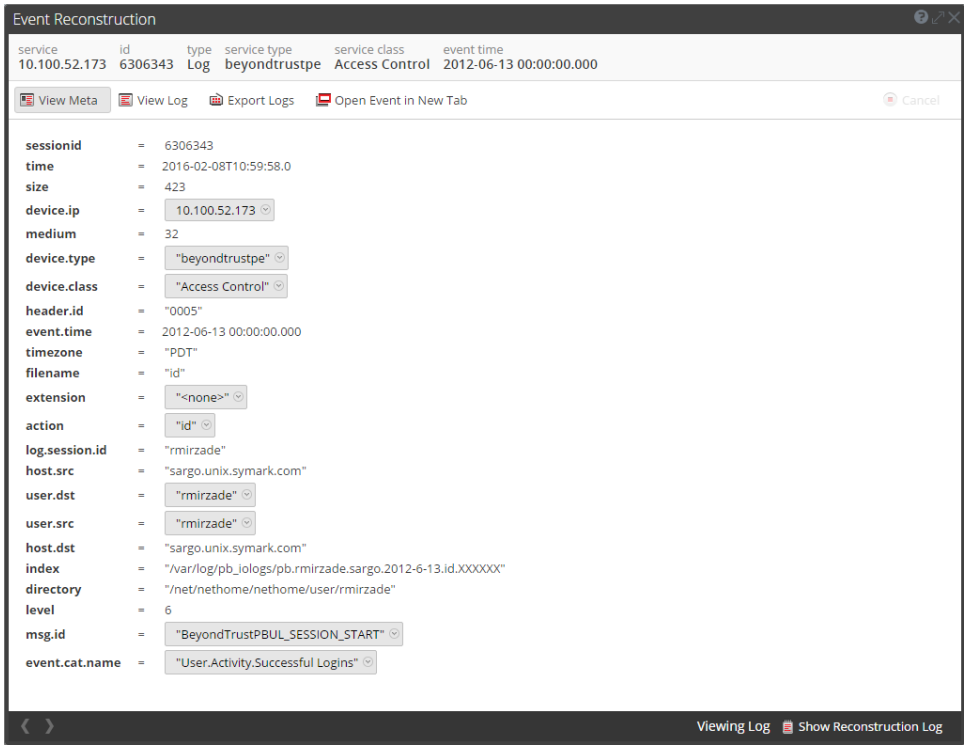
8. Navigate to **Administration > Services** and check the **Log Decoder(s)** then click **Restart.**

9. Navigate to **Administration > Services** and check the **Log Decoder(s)** then click **View> Config.**

10. The new device is listed under the Log Decoder(s) General Tab within the Service Parsers Configuration.



11. The Log Decoder is now ready to parse events for this device. Below is an example of the RSA SA metadata collected from an Absolute DDS logfile.



PowerBroker Servers for UNIX/Linux Configuration

PowerBroker Servers generates event log records every time a privileged command is accepted or rejected by the PowerBroker Master. These event logs are always recorded in the PowerBroker internal eventlog file. In order to interoperate with RSA Security Analytics, PowerBroker needs to be configured to also write the event log records in syslog. Once the product is configured to write the event log records to syslog, the “Accept” and “Reject” events will always be generated. The “Session” events (Session started, Session finished, Session failed) are optional and also need to be configured either during the installation or later in pb.settings. (See PBUL Admin Guide for details)

1. To configure PBUL components to write to syslog, during the installation
   - Set “Use syslog?” to **YES**.
   - Set the “syslog facility to use” to **LOG_AUTH**.
   - Enable the “Syslog pblocald sessions to be logged”, by entering **YES.**

2. Alternatively, after installing PBUL components, you can configure the Master as well as the Clients (submit/run hosts) to use syslog by setting the following keywords in /etc/pb.settings:

```
syslog              yes
facility            LOG_AUTH
syslogsessions      yes
```

3. Verify that the superdaemon configuration files for pbmasterd, pblogd and pblocald have the "server_args" variable set to:

```
For pbmasterd and pblogd:
server_args = -ar
For pblocald:
```

server_args = -a

4. Starting with version 7.0 of PBUL, you can use the new "syslog formatting" keywords to customize the formatting of the syslog records. In order to interoperate with RSA Security Analytics beyondtrustpbulpe event source package, the syslog records generated by PBUL components need to have a specific format, and therefore the "syslog formatting" keywords need to be set to the following values in /etc/pb.settings, on the Master, Logserver and the Client components of PBUL:

```
syslog_accept_format                "BeyondTrustPBUL_ACCEPT: On %date%-%time%
%timezone% accepted command '%command%' to run as command '%runcommand%'
submitted by '%user%' on '%submithost%' as request user '%requestuser%' and run
by '%runuser%' on '%runhost%'. Policy File='%lineinfile%' - line %linenum%"

syslog_reject_format                "BeyondTrustPBUL_REJECT: On %date%-%time%
%timezone% rejected command '%command%' to run as command '%runcommand%'
submitted by '%user%' on '%submithost%' as request user '%requestuser%' and run
by '%runuser%' on '%runhost%'. Policy File='%lineinfile%' - line %linenum%"

syslogsession_start_format          "BeyondTrustPBUL_SESSION_START: On %date%-
%time% %timezone% started command '%command%' to run as command '%runcommand%'
submitted by '%user%' on '%submithost%' as request user '%requestuser%' and run
by '%runuser%' on '%runhost%'. IOLog File Name='%iolog%' - Current Working
Directory=%runcwd%"

syslogsession_start_fail_format     "BeyondTrustPBUL_SESSION_START_FAILED: On
%date%-%time% %timezone% failed to start command '%command%' to run as command
'%runcommand%' submitted by '%user%' on '%submithost%' as request user
'%requestuser%' and run by '%runuser%' on '%runhost%'. IOlog File
Name='%iolog%' - Current Working Directory=%runcwd%"

syslogsession_finished_format       "BeyondTrustPBUL_SESSION_FINISH: On %date%-
%time% %timezone% finished to run command '%command%' to run as command
'%runcommand%' submitted by '%user%' on '%submithost%' as request user
'%requestuser%' and run by '%runuser%' on '%runhost%' with exit status
'%exitstatus%'. IOlog File Name='%iolog%' - Current Working Directory=%runcwd%"
```

5. With PBUL configured to generate the syslog event records in this specific format and with the RSA Security Analytics event source package deployed, RSA Security Analytics will now be able to interpret the PBUL syslog event log records and monitor the event source.

Certification Checklist for RSA Security Analytics

Date Tested: 2/16/2016

Certification Environment		
Product Name	**Version Information**	**Operating System**
RSA Security Analytics	10.5	Virtual Appliance
BeyondTrust PowerBroker	8.0	UNIX/Linux

Security Analytics 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

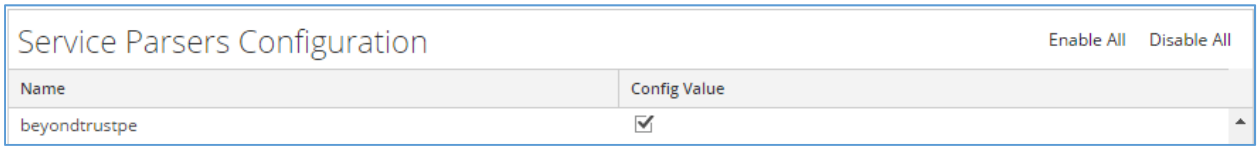
Appendix

Security Analytics Disable Device Parser

To disable the Security Analytics Integration Package but not delete the XML from the system, perform the following:

1. Navigate to **Administration > Services** and check the **Log Decoder(s)** then click **View> Config.**

2. From the **Service Parses Configuration** window, scroll down to the device you wish to disable and uncheck the Config Value checkbox.



3. Click **Apply** to save settings.

Security Analytics Remove Device Parser

To remove the Security Analytics 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 the device you are targeting for removal and delete the folder containing the device xml.
3. Returning the system to its original state will require either modifying or removing the **table-map-custom.xml** based on your systems configuration. The table-map-custom.xml file is located in the **/etc/netwitness/ng/envision/etc** folder of the SA Log Decoder(s).