

RSA NetWitness Logs

Event Source Log Configuration Guide



Akamai Kona

Last Modified: Friday, February 2, 2018

Event Source Product Information:

Vendor: [Akamai](#)

Event Source: Akamai Kona

Versions: 1.0

RSA Product Information:

Supported On: NetWitness Suite 10.0 and later

Event Source Log Parser: cef

Note: The CEF parser parses this event source as `device.type=akamaikona`

Collection Method: Syslog

Event Source Class.Subclass: Security.Application Firewall

To configure the Akamai Kona event source, you must:

- I. Configure Syslog Output on Akamai Kona CEF Connector
- II. Configure RSA NetWitness Suite for Syslog Collection

Configure Syslog Output Akamai Kona CEF Connector

The Akamai Managed Kona Site Defender Service is a managed security service designed to help build a responsive cloud security strategy. It leverages the expertise and infrastructure provided by Akamai's Security Operations Center to help customers maintain attack readiness, get security monitoring and attack support, and receive ongoing security reporting.

Hardware and Software Requirements

The following are the software and hardware requirements for running the CEF connector:

- Sun JRE 1.8+
- 2 CPU cores
- 6GB RAM
- 2GB Free Disk Space
- Run a Linux Kernel greater than 2.6

Configuration Procedure

Perform the following steps to configure the Akamai Kona CEF connector.

1. Visit <https://developer.akamai.com/tools/siem-integration> to get the latest CEFConnector distribution package.
2. Download and unzip the distribution package anywhere on the file system.
3. Install CEF Connector as a service, by creating symbolic link to the **bin/AkamaiCEFConnector.sh** shell script in `/etc/init.d`.

The shell script accepts the following commands:

- start
- stop
- status
- resetdb

Note: Resetdb deletes **cefconnector.db**, which contains the last successful offset data pull. Removing the file causes the connector to process **offset=NULL** as long as **timebased** setting is false. If **timebased** is true, a new offset is saved after the first successful pull.

4. Configure the `config/CEFCConnector.properties` file with the user specific parameters provided by Akamai. For details, see [CEF Connector Properties](#).
5. Configure the `config/log4j2.xml` file to provide the RSA NetWitness Logs and Packets Host IP, Port, and Protocol properties in the CEF Syslog Configuration Section.

Log Pattern for CEF should be provided in the following format:

<>**PARITY_NUM**>> %d{yyyy-MM-dd HH:mm:ss} %msg%n

```

<?xml version="1.0" encoding="UTF-8"?>
<Configuration status="warn">
  <Properties>
    <!--
      log-path: Path location of file logs. Use relative or specific path (example: logs)
      log-name: File name for logs (example: filename)
      SizeBasedTriggeringPolicy: Log size rollover limit (example 1MB)
      DefaultRolloverStrategy: Max number of Logs rollover limit.
    -->
    <Property name="log-path">/home/kona-user/</Property>
    <Property name="log-name">cefconnector</Property>
    <Property name="SizeBasedTriggeringPolicy">100 MB</Property>
    <Property name="DefaultRolloverStrategy">20</Property>
    <!--
      CEF Syslog Configuration
      CEFHost: Remote CEF Syslog Server Host (example: 127.0.0.1)
      CEFPort: Remote CEF Syslog Server Port (example: 514)
      CEFProtocol: Remote CEF Syslog Server Protocol (UDP/TCP)
    -->
    <Property name="CEFHost"></Property>
    <Property name="CEFPort"></Property>
    <Property name="CEFProtocol">TCP</Property>
    <!--
      Log Patterns
      logPattern-CEF: CEF syslog pattern for remote CEF syslog server (do not change)
      logPattern-Console: Console log pattern
      logPattern-FileInfo: File log pattern for all [INFO] type logs
      logPattern-FileError: File log pattern for all [ERROR] type logs
    -->
    <Property name="logPattern-CEF">&lt;&gt;&lt;&gt; %d{yyyy-MM-dd HH:mm:ss} %msg%n</Property>
    <Property name="logPattern-Console">&lt;&gt;&lt;&gt; %d{yyyy-MM-dd HH:mm:ss} %msg%n</Property>
    <Property name="logPattern-FileInfo">&lt;&gt;&lt;&gt; %d{yyyy-MM-dd HH:mm:ss} %msg%n</Property>
    <Property name="logPattern-FileWarn">&lt;&gt;&lt;&gt; %d{yyyy-MM-dd HH:mm:ss} %msg%n</Property>
    <Property name="logPattern-FileError">&lt;&gt;&lt;&gt; %d{yyyy-MM-dd HH:mm:ss} %msg%n</Property>
  </Properties>
</Configuration>

```

Note: Logs can be collected on the local machine if the **log-path** is provided with the **log-name** (Optional). Log patterns (Console log, File Information, File Warning, and File Error) are optional, but might prove useful.

6. Start the CEF Connector service, using the shell script.

Once you start the service, logs are collected in the Common Event Format (CEF).

CEF Connector Properties

The following table describes the available settings in the `config/CEFConnector.properties` file.

Name	Description
<code>Connector.refresh.period *</code>	The rate that the connector will pull from the SIEM API in seconds. Default Value is 60 . Set this parameter to a positive integer value. Any other value will be ignored (default value will be used).
<code>Akamai.data.requesturlhost *</code>	Request URL for API. This value cannot be blank or commented out.
<code>akamai.data.configs *</code>	Security configuration IDs, separated with commas (,). This parameter cannot be blank or commented out.
<code>akamai.data.timebased *</code>	Boolean value for using an offset token. <ul style="list-style-type: none"> Set to true to pull data from a specific time Set to false to use an offset token
<code>akamai.data.timebased.from</code>	If timebased is true, the from field in epoch format will be used as the beginning timestamp to pull security events. This field will be ignored if the timebased is false.
<code>akamai.data.timebased.to</code>	If timebased is true, the to field in epoch format will be used as the end timestamp to pull security events. If no value or invalid format is provided, default value will be used. This field will be ignored if the timebased is false.
<code>akamai.data.limit</code>	Limits the number of events to pull. If no value is provided or an invalid value is provided, the default limit on the API side will be used. Default value is 200000. Set this parameter to a positive integer value. Any other value will be ignored (default value will be used).
<code>akamai.data.accesstoken *</code>	OPEN API credentials need to be provisioned by the customer in Akamai LUNA portal and to be configured by the SIEM administrator.
<code>akamai.data.clienttoken *</code>	OPEN API credentials need to be provisioned by the customer in Akamai LUNA portal and to be configured by the SIEM administrator.
<code>akamai.data.clientsecret *</code>	OPEN API credentials need to be provisioned by the customer in Akamai LUNA portal and to be configured by the SIEM administrator.
<code>akamai.data.baseurl *</code>	OPEN API credentials need to be provisioned by the customer in Akamai LUNA portal and to be configured by the SIEM administrator.
<code>akamai.cefformatheader *</code>	CEF Header Values are separated by " ". If " " is part of a static string, then it must be escaped with "\\". Values can be static or generated from available functions: requestURL() , eventClassId() , name() , severity() , appliedAction() , ipv6src() . There need to be 7 values, separated by " " and starting with CEF :

Name	Description
akamai.cefformatextension *	<p>CEF Extension Values are separated by a space. Values can be any of the following:</p> <ul style="list-style-type: none"> • static • generated from available functions (eventClassId(), name(), severity(), appliedAction(), ipv6src()), or • pulled from JSON API. <p>JSON API is defined by {}\$ and each JSON object is separate by a period (.). Static Values are defined by quotation marks. Function generated values are defined by () and must be one of the available functions defined in documentation. Each space-separated value needs to be a pair.</p>
akamai.base64fields	If an API JSON object is base64 encoded, it must be defined here.
akamai.urlencoded	If an API JSON object is URL-encoded, it must be defined here.
akamai.multivaluedelim	Delimiter used to separate multi-valued CEF fields. Default value is a comma (.). Specifying " " (a space) is treated the same as "" (empty string) and the default value is used.
connector.consumer.count	Limits the number of consumer threads. Default value is 3.

Access the SIEM API

To access the SIEM API from behind a proxy server, ensure that your proxy:

- Whitelists the domains ***.cloudsecurity.akamaiapis.net**
- Does not interfere with HTTP request headers for those domains. If, due to a strict enterprise security policy, your proxy does change these headers, make sure that at a minimum you allow and do not change the Host and Authorization headers.

Configure RSA NetWitness Suite

Perform the following steps in RSA NetWitness Suite:

- Ensure the required parser is enabled
- Configure Syslog Collection

Ensure the Required Parser is Enabled

If you do not see your parser in the list while performing this procedure, you need to download it in RSA NetWitness Suite Live.

Ensure that the parser for your event source is enabled:

1. In the **NetWitness** menu, select **Administration > Services**.
2. In the Services grid, select a Log Decoder, and from the Actions menu, choose **View > Config**.
3. In the Service Parsers Configuration panel, search for your event source, and ensure that the **Config Value** field for your event source is selected.

Note: The required parser is **cef**.



Configure Syslog Collection

Note: You only need to configure Syslog collection the first time that you set up an event source that uses Syslog to send its output to NetWitness.

You should configure either the Log Decoder or the Remote Log Collector for Syslog. You do not need to configure both.

To configure the Log Decoder for Syslog collection:

1. In the **NetWitness** menu, select **Administration > Services**.
2. In the Services grid, select a Log Decoder, and from the Actions menu, choose **View > System**.
3. Depending on the icon you see, do one of the following:

- If you see  **Start Capture**, click the icon to start capturing Syslog.
- If you see  **Stop Capture**, you do not need to do anything; this Log Decoder is already capturing Syslog.

To configure the Remote Log Collector for Syslog collection:

1. In the **NetWitness** menu, select **Administration > Services**.
2. In the Services grid, select a Remote Log Collector, and from the Actions menu, choose **View > Config > Event Sources**.
3. Select **Syslog/Config** from the drop-down menu.

The Event Categories panel displays the Syslog event sources that are configured, if any.

4. In the Event Categories panel toolbar, click **+**.

The Available Event Source Types dialog is displayed.

5. Select either **syslog-tcp** or **syslog-udp**. You can set up either or both, depending on the needs of your organization.
6. Select the new type in the Event Categories panel and click **+** in the Sources panel toolbar.

The Add Source dialog is displayed.

7. Enter **514** for the port, and select **Enabled**. Optionally, configure any of the Advanced parameters as necessary.

Click **OK** to accept your changes and close the dialog box.

Once you configure one or both syslog types, the Log Decoder or Remote Log Collector collects those types of messages from all available event sources. So, you can continue to add Syslog event sources to your system without needing to do any further configuration in NetWitness.

Copyright © 2018 Dell Inc. or its subsidiaries. All Rights Reserved.

Trademarks

For a list of RSA trademarks, go to www.emc.com/legal/emc-corporation-trademarks.htm#rsa.