# RSA NetWitness Platform

Event Source Log Configuration Guide

# Azure Monitor

Last Modified: Monday, September 16, 2019

**Event Source Product Information:**

**Vendor**: Microsoft
**Event Source**: Azure Monitor Activity and Diagnostic Logs
**Versions**: API v1.0

**RSA Product Information:**

**Supported On**: NetWitness Platform 11.2.1 and later
**Event Source Log Parser**: cef

**Note:** The CEF parser parses this event source as **device.type=azuremonitor**.

**Collection Method**: Plugin Framework
**Event Source Class.Subclass**: Host.Cloud

RSA Netwitness can pull in the Azure Activity and Diagnostic logs from Azure EventHub. To configure Azure EventHub and RSA NetWitness Platform to capture Activity and Diagnostic Logs, you must complete these tasks:

I. Configure Azure Monitor to export Activity and Diagnostic Logs to an EventHub

II. Set Up Azure Monitor Event Source in RSA NetWitness

# About Azure Monitor

Azure Monitor maximizes the availability and performance of your applications by delivering a comprehensive solution for collecting, analyzing, and acting on telemetry from your cloud and on-premises environments. It helps you understand how your applications are performing and proactively identifies issues affecting them and the resources upon which they depend. Azure Monitor has the functionality to export Azure Activity, Diagnostic, Azure Active Directory Sign-in and Audit Logs.

The **Azure Activity Log** is a subscription log that provides insight into subscription-level events that have occurred in Azure. This includes a range of data, from Azure Resource Manager operational data to updates on Service Health events. Using the Activity Log, you can determine the 'what, who, and when' for any write operations (PUT, POST, DELETE) taken on the resources in your subscription. You can also understand the status of the operation and other relevant properties.

**Azure Monitor Diagnostic Logs** are logs emitted by an Azure service that provide rich, frequent data about the operation of that service.

**Azure Active Directory Events** consist of the following two type of Events:

- Sign-ins: Sign-in logs provides information about the usage of managed applications and user sign-in activities.
- Audit logs: Audit logs provide traceability through logs for all changes done by various features within Azure AD. Examples of audit log entries include changes made to any resources within Azure AD, such as adding or removing users, apps, groups, roles and policies.

For more information, see the following topics, available from Microsoft Docs website (docs.microsoft.com):

- Azure Activity Logs: Monitor Subscription Activity with the Azure Activity Log
- Azure Monitor Diagnostic Logs: Collect and consume log data from your Azure Resources
- Azure AD events: Azure AD activity logs in Azure Monitor

# Configure Azure Monitor to export Activity and Diagnostic Logs to an EventHub

Please refer to the following topics on Microsoft Docs website (docs.microsoft.com):

- To create an EventHub and forward Activity logs to it, perform the steps described in Stream the Azure Activity Log to Event Hubs
- To export Diagnostic logs to an EventHub, perform the steps described in the following link: Stream Azure Diagnostic Logs to an event hub
- To export Azure AD Sign-In and Audit logs to an EventHub, perform the steps described in the following link: Azure AD activity logs in Azure Monitor

# Set Up the Azure Monitor Event Source in NetWitness Platform

In RSA NetWitness Platform, perform the following tasks:

I. Deploy the CEF parser from Live

II. Configure the event source.

## Deploy Azure Monitor Files from Live

Azure Monitor uses the cef parser.

### To deploy the cef parser from Live:

1. In the RSA NetWitness Platform menu, select **CONFIGURE**.

   The **Live Content** tab is displayed.

2. Browse Live Content for the **Common Event Format (cef)** parser, using **Log Device** as the **Resource Type**.

3. Select the **cef** parser from the list and click **Deploy** to deploy it to the appropriate Log Decoders, using the Deployment Wizard.

4. You also need to deploy the Azure Monitor package. Browse Live for Azure Monitor EventLogs content, typing "azuremonitor" into the Keywords text box, then click **Search**.

5. Select the package and click **Deploy** to deploy it to the appropriate Log Collectors.

   **Note:** On a hybrid installation, you need to deploy the package on both the VLC and the LC.

For more details, see the Add or Update Supported Event Source Log Parsers topic, or the Live Services Management Guide.

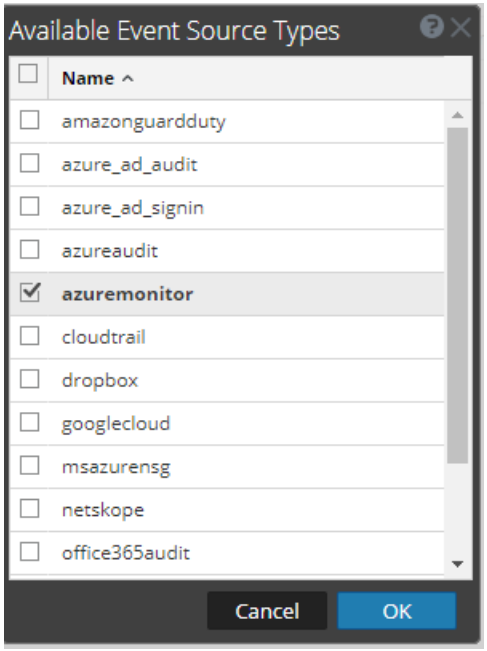
## Configure the Event Source

### To configure the Azure Monitor Platform Event Source:

1. In the RSA NetWitness Platform menu, select **ADMIN > Services**.
2. In the Services grid, select a Log Collector service, and from the Actions menu, choose **View** > **Config**.
3. In the **Event Sources** tab, select **Plugins/Config** from the drop-down menu.

   The Event Categories panel displays the File event sources that are configured, if any.
4. In the Event Categories panel toolbar, click +.

   The Available Event Source Types dialog is displayed.

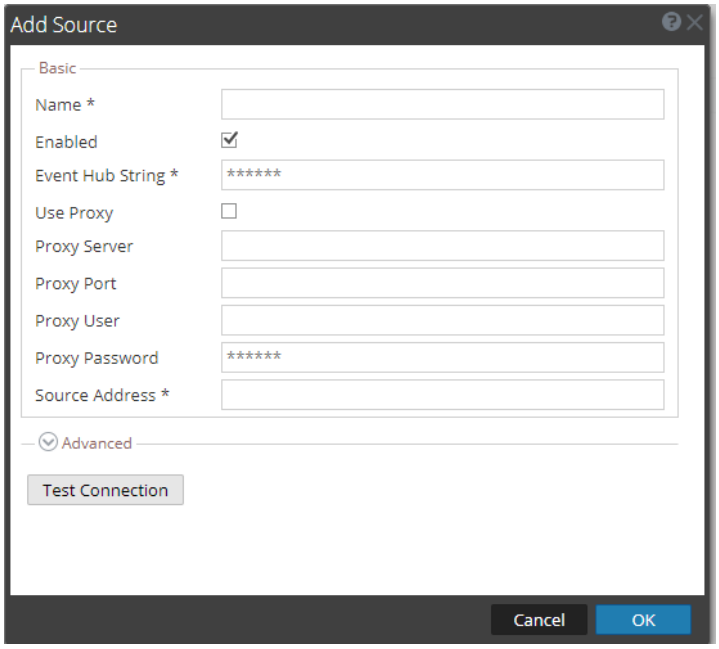



5. Select **azuremonitor** from the list, and click **OK**.

   The newly added event source type is displayed in the Event Categories panel.

6. Select the new type in the Event Categories panel and click + in the Sources panel toolbar.

   The Add Source dialog is displayed.




7. Define parameter values, as described in Azure Monitor Collection Configuration Parameters.

8. Click **Test Connection**.

   The result of the test is displayed in the dialog box. If the test is unsuccessful, edit the device or service information and retry.

   **Note:** The Log Collector takes approximately 60 seconds to return the test results. If it exceeds the time limit, the test times out and RSA NetWitness Platform displays an error message.

9. If the test is successful, click **OK**.

   The new event source is displayed in the Sources panel.

10. Repeat steps 4–9 to add another Azure Monitor plugin type.

# Azure Monitor Collection Configuration Parameters

The following table describes the configuration parameters for the Azure Monitor Platform integration with RSA NetWitness Platform.

Note the following:

- Fields marked with an asterisk (*) are required.
- If a proxy is being used, the proxy needs to allow traffic through port 5671 (used for AMQPS).

## Basic Parameters

| Name | Description |
|---|---|
| Name* | Enter an alpha-numeric, descriptive name for the source. This value is only used for displaying the name on this screen. |
| Enabled | Select the box to enable the event source configuration to start collection. The box is selected by default. |
| Event Hub String | You obtain the Event Hub connection string from the **Access Policy** tab of the Event Hub. |
| Use Proxy | Check to enable proxy. |
| Proxy Server | If you are using a proxy, enter the proxy server address. |
| Proxy Port | If proxy is being used, then the proxy needs to allow traffic through port 5671, used for AMQPS. |
| Proxy User | Username for the proxy (leave empty if using anonymous proxy). |
| Proxy Password | Password for the proxy (leave empty if using anonymous proxy). |

| Name | Description |
|---|---|
| Source Address | IP address that is to be provided to the Azure Monitor plugin instance: logs from this event source will be collected with this device IP.<br><br>**Note:** This is an arbitrary ip address chosen by the user. This value has no bearing on the collection of logs: its value is captured by the **device.ip** meta key, and can help you to query or group events collected by a particular instance of the plugin. |
| Test Connection | Checks the configuration parameters specified in this dialog to make sure they are correct. |

## Advanced Parameters

| Parameter | Description |
|---|---|
| Polling Interval | Interval (amount of time in seconds) between each poll. The default value is **180**.<br><br>For example, if you specify 180, the collector schedules a polling of the event source every 180 seconds. If the previous polling cycle is still underway, it will wait for it to finish that cycle. If you have a large number of event sources that you are polling, it may take longer than 180 seconds for the polling to start because the threads are busy. |
| Max Duration Poll | The maximum duration of polling cycle (how long the cycle lasts) in seconds. |
| Max Events Poll | The maximum number of events per polling cycle (how many events collected per polling cycle). |
| Max Idle Time Poll | Maximum idle time, in seconds, of a polling cycle. 0 indicates no limit.> **300** is the default value. |
| Command Args | Optional arguments to be added to the script invocation. |

| Parameter | Description |
| --- | --- |
| Debug | **Caution:** Only enable debugging (set this parameter to "On" or "Verbose") if you have a problem with an event source and you need to investigate this problem. Enabling debugging will adversely affect the performance of the Log Collector.<br><br>Enables and disables debug logging for the event source.<br><br>Valid values are:<br><br>• **Off** = (default) disabled<br>• **On** = enabled<br>• **Verbose** = enabled in verbose mode - adds thread information and source context information to the messages.<br><br>This parameter is designed to debug and monitor isolated event source collection issues. The debug logging is verbose, so limit the number of event sources to minimize performance impact.<br><br>If you change this value, the change takes effect immediately (no restart required). |
| SSL Enable | Uncheck this box to disable SSL certificate verification. |



## Trademarks

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