
Micro Focus Security ArcSight SmartConnectors

SmartConnector for Microsoft Azure Monitor Event Hub

Configuration Guide

Document Release Date: July 31, 2020

Software Release Date: July 31, 2020



Legal Notices

Micro Focus
The Lawn
22-30 Old Bath Road
Newbury, Berkshire RG14 1QN
UK

<https://www.microfocus.com>

Copyright Notice

© Copyright 2020 Micro Focus or one of its affiliates

Confidential computer software. Valid license from Micro Focus required for possession, use or copying. The information contained herein is subject to change without notice.

The only warranties for Micro Focus products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Micro Focus shall not be liable for technical or editorial errors or omissions contained herein.

No portion of this product's documentation may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information storage and retrieval systems, for any purpose other than the purchaser's internal use, without the express written permission of Micro Focus.

Notwithstanding anything to the contrary in your license agreement for Micro Focus ArcSight software, you may reverse engineer and modify certain open source components of the software in accordance with the license terms for those particular components. See below for the applicable terms.

U.S. Governmental Rights. For purposes of your license to Micro Focus ArcSight software, "commercial computer software" is defined at FAR 2.101. If acquired by or on behalf of a civilian agency, the U.S. Government acquires this commercial computer software and/or commercial computer software documentation and other technical data subject to the terms of the Agreement as specified in 48 C.F.R. 12.212 (Computer Software) and 12.211 (Technical Data) of the Federal Acquisition Regulation ("FAR") and its successors. If acquired by or on behalf of any agency within the Department of Defense ("DOD"), the U.S. Government acquires this commercial computer software and/or commercial computer software documentation subject to the terms of the Agreement as specified in 48 C.F.R. 227.7202-3 of the DOD FAR Supplement ("DFARS") and its successors. This U.S. Government Rights Section 18.11 is in lieu of, and supersedes, any other FAR, DFARS, or other clause or provision that addresses government rights in computer software or technical data.

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number
- Document Release Date, which changes each time the document is updated
- Software Release Date, which indicates the release date of this version of the software

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

[ArcSight Product Documentation on the Micro Focus Security Community](#)

Support

Contact Information

Phone	A list of phone numbers is available on the Technical Support Page: https://softwaresupport.softwaregrp.com/support-contact-information
Support Web Site	https://softwaresupport.softwaregrp.com/
ArcSight Product Documentation	https://community.softwaregrp.com/t5/ArcSight-Product-Documentation/ct-p/productdocs

Contents

Introduction	6
Related Azure Services	6
Understanding Data Collection	7
Deploying the Connector	10
Prerequisites	10
Certified Platforms for Azure Event Hubs Deployment	10
Setting User Permissions in Azure	10
Opening Ports	13
Enabling Windows PowerShell to Run Scripts	14
Deploying the Connector	15
Verifying the Deployment in Azure	17
Post-Deployment Configurations	18
Additional Configurations for App Service Plan	18
Streaming Diagnostic Logs	18
Updating Keystore Certificate	19
Customizing the Connector	21
Performance Scaling	22
Configuring Load Balancer	22
Upgrading the Connector	23
Undeploying the Connector	23
Appendix A: Azure Event Log Categories	Y
Creating a Map File	Y
To create a Map File:	Z
Updating Map Files	AA
Diagnostic, Audit, and Sign-In Log Categories	AA
Activity Log Categories	AB
Active Directory Log Categories	AC
Appendix B: Troubleshooting	AD
Errors during Deployment	AD
Connecting Errors	AD
Parsing Errors	AD
Sharing Logs for Troubleshooting	AE
Using a Private IP	AE
Event Hubs and/or Storage Account.	AE
Function Apps	AE

Azure Datacenters with Stamps (Scale Units) without Premium V2 VMs	AF
AppService Plan with Basic Pricing Tier Created on a Stamp That Does not Support Premium V2 VMs	AF
Send Documentation Feedback	AH

Introduction

Microsoft Azure is an ever-expanding set of cloud services to help your organization meet your business challenges. It is the freedom to build, manage, and deploy applications on a massive, global network using your favorite tools and frameworks.

Cloud applications are complex with many moving parts. Logging data can provide insights about your applications and help you:

- Troubleshoot past problems or prevent potential ones
- Improve application performance or maintainability
- Automate actions that would otherwise require manual intervention

Azure logs are categorized into the following types:

- **Control/management logs** provide information about Azure Resource Manager CREATE, UPDATE, and DELETE operations. For more information, see [Azure Activity Logs](#).
- **Data plane logs** provide information about events raised as part of Azure resource usage. Examples of this type of log are the Windows event system, security, and application logs in a virtual machine (VM) and the [diagnostics logs](#) that are configured through Azure Monitor.
- **Processed events** provide information about analyzed events/alerts that have been processed on your behalf. Examples of this type are [Azure Security Center Alerts](#) where [Azure Security Center](#) has processed and analyzed your subscription and provides concise security alerts. For more information, see [Azure Security Logging and Auditing](#).

Related Azure Services

The following services are used when working with Azure Monitor Event Hub connector:

- **Azure Resource Manager:** Azure Resource Manager is the deployment and management service for Azure. It provides a management layer that enables you to create, update, and delete resources in your Azure subscription. You use management features, like access control, locks, and tags, to secure and organize your resources after deployment. For more information, see [Azure Resource Manager](#).
- **Azure App Service plan:** In App Service, an app runs in an App Service plan. An App Service plan defines a set of compute resources for a web app to run. These compute resources are analogous to the server farm in conventional web hosting. One or more apps can be configured to run on the same computing resources (or in the same App Service plan). For more information, see [Azure App Service Plan Overview](#).

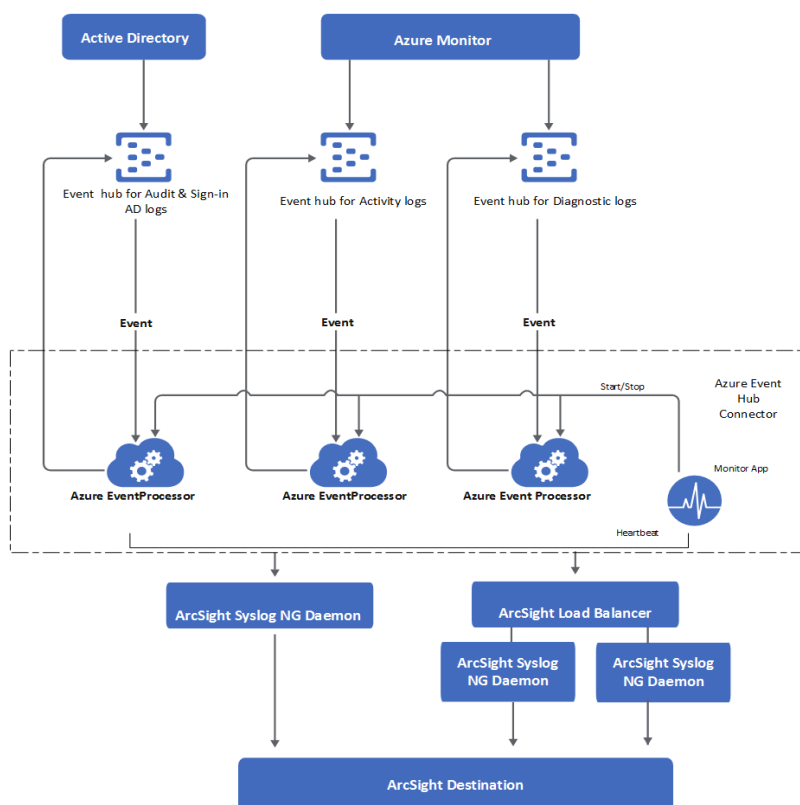
- **Azure Functions:** Azure Functions allows you to run small pieces of code (called "functions") without worrying about application infrastructure. With Azure Functions, the cloud infrastructure provides all the up-to-date servers you need to keep your application running at scale. For more information, see [An introduction to Azure Functions](#).
- **Storage account:** An Azure storage account contains all of your Azure Storage data objects: blobs, files, queues, tables, and disks. The storage account provides a unique namespace for your Azure Storage data that is accessible from anywhere in the world over HTTP or HTTPS. Data in your Azure storage account is durable and highly available, secure, and massively scalable. For more information, see [Storage Account Overview](#).
- **Azure Event Hubs:** Azure Event Hubs is a big data streaming platform and event ingestion service. It can receive and process millions of events per second. Data sent to an event hub can be transformed and stored by using any real-time analytics provider or batching/storage adapters. For more information, see [Azure Event Hubs – A big data streaming platform and event ingestion service](#).

The Azure Monitor Event Hub connector helps you monitor the activities on Microsoft Azure Cloud services.

This connector collects events and logs from Azure Active Directory and Azure Monitor, normalizes the events to Common Event Format (CEF), and then sends them to either ArcSight Syslog NG Daemon SmartConnector (SmartConnector) or to ArcSight Load Balancer. If the events are sent to ArcSight Load Balancer, these, are consequently sent to Syslog NG Daemon SmartConnector.

Understanding Data Collection

The following diagram provides a high-level overview of how the Azure Monitor Event Hub connector collects and sends data to ArcSight's destination.



The Azure Monitor Event Hub connector collects the following event logs from Active Directory and Azure Monitor in Azure:

- **Active Directory Logs**

- **Audit logs:** Provides records of system activities for compliance.
- **Sign-in logs:** Provides information related to user logins.

Note: To export Active Directory sign-in logs, you must have one of P1 or P2 premium editions of Azure Active Directory.

- **Azure Monitor Logs**

- **Activity logs:** Provides data related to write operations that were performed on resources in your subscription.
- **Diagnostic logs:** Provides data related to operations performed by your resource.

On installation, the connector first creates event hubs for both Active Directory and Azure Monitor. The connector then automatically configures the above mentioned log types to be forwarded to the following event hubs: Active Directory, Activity, and Diagnostics. For detailed information about event hubs, see the Azure documentation.

The Azure Event Processor collects logs in JSON format and then converts these to CEF format.

The Azure Event Processor then forwards these CEF events to an ArcSight Syslog NG Daemon SmartConnector or Load Balancer through a secured communication channel using TLS 1.2.

The connector establishes a TLS 1.2 connection by accepting a server certificate from ArcSight Syslog NG Daemon SmartConnector or ArcSight Load Balancer.

The Monitor App continuously monitors the heartbeat of the Syslog NG Daemon SmartConnector or Load Balancer to ensure that it is up and running to receive events. If the Syslog NG Daemon SmartConnector or Load Balancer is down due to an unexpected shutdown of the machine or network issues, this connector stops further processing of events from the event hub. The unprocessed events are sent back to the event hub to avoid data loss. Once the Syslog NG Daemon SmartConnector or Load Balancer is up and running, the connector continues to send the events to the Syslog NG Daemon SmartConnector. However, the Monitor App will not monitor Syslog NG Daemon SmartConnectors connected to the Load Balancer.

The Syslog NG Daemon SmartConnector then sends the events to the ArcSight destination.

Deploying the Connector

This section provides information about deploying the connector to collect and forward events from Azure Cloud Services to a Syslog NG Daemon SmartConnector or to a Load Balancer, then the events can be sent to an ArcSight destination.

["Prerequisites" below](#)

["Deploying the Connector" on page 15](#)["Deploying the Connector" on page 15](#)

["Post-Deployment Configurations" on page 18](#)

Prerequisites

["Certified Platforms for Azure Event Hubs Deployment " below](#)

["Setting User Permissions in Azure " below](#)

["Opening Ports " on page 13](#)

["Enabling Windows PowerShell to Run Scripts" on page 14](#)

Certified Platforms for Azure Event Hubs Deployment

- **Operating System:** : Microsoft Windows Server 2012 , 2016 and 2019 (in the cloud with Azure).

Note: Virtual machines are required for deployment, upgrade and undeployment. Users may choose to delete their VM after deployment.

Setting User Permissions in Azure

In Azure, users must be associated with a subscription to provide them with access to resources such as virtual machine, storage account, virtual network, and so on.

Therefore, you must determine the subscription you want to use for the connector and add users to the required subscription. You must also assign users to a role, to define their permission to perform tasks. With Azure defined roles, only a user with the **Owner or Contributor** role, can deploy and configure the connector. Users can also create a custom roles with the following permissions:

"Microsoft.Resources/subscriptions/resourceGroups/delete",

"Microsoft.Resources/subscriptions/resourceGroups/write",

"Microsoft.Resources/subscriptions/resourceGroups/read",

"Microsoft.Web/serverfarms/Delete",

"Microsoft.Web/serverfarms/Write",
"Microsoft.Web/serverfarms/Read",
"Microsoft.ServiceBus/namespaces/authorizationRules/listkeys/action",
"microsoft.web/sites/functions/write",
"microsoft.web/sites/functions/read",
"microsoft.web/sites/functions/delete",
"Microsoft.Web/sites/config/Read",
"microsoft.web/sites/config/delete",
"Microsoft.Web/sites/config/Write",
"Microsoft.Web/sites/stop/Action",
"Microsoft.Web/sites/start/Action",
"Microsoft.Web/sites/restart/Action",
"Microsoft.Web/sites/Delete",
"Microsoft.Web/sites/Write",
"Microsoft.Web/sites/Read",
"Microsoft.Storage/storageAccounts/fileServices/shares/write",
"Microsoft.Storage/storageAccounts/fileServices/shares/read",
"Microsoft.Storage/storageAccounts/fileServices/shares/delete",
"Microsoft.Storage/storageAccounts/fileServices/read",
"Microsoft.Storage/storageAccounts/fileServices/write",
"Microsoft.Storage/storageAccounts/blobServices/containers/read",
"Microsoft.Storage/storageAccounts/blobServices/containers/delete",
"Microsoft.Storage/storageAccounts/blobServices/containers/write",
"Microsoft.Storage/storageAccounts/blobServices/write",
"Microsoft.Storage/storageAccounts/blobServices/read",
"Microsoft.Storage/storageAccounts/write",
"Microsoft.Storage/storageAccounts/read",
"Microsoft.Storage/storageAccounts/delete",
"Microsoft.EventHub/namespaces/eventhubs/Delete",
"Microsoft.EventHub/namespaces/eventhubs/read",
"Microsoft.EventHub/namespaces/eventhubs/write",

"Microsoft.EventHub/namespaces/Delete",
"Microsoft.EventHub/namespaces/read",
"Microsoft.EventHub/namespaces/write",
"Microsoft.Storage/storageAccounts/listkeys/action",
"Microsoft.EventHub/namespaces/authorizationRules/delete",
"Microsoft.EventHub/namespaces/authorizationRules/write",
"Microsoft.EventHub/namespaces/authorizationRules/read",
"Microsoft.EventHub/namespaces/authorizationRules/listkeys/action",
"Microsoft.EventHub/namespaces/eventHubs/consumerGroups/Delete",
"Microsoft.EventHub/namespaces/eventHubs/consumerGroups/read",
"Microsoft.EventHub/namespaces/eventHubs/consumerGroups/write",
"Microsoft.Insights/DiagnosticSettings/Read",
"Microsoft.Insights/DiagnosticSettings/Delete",
"Microsoft.Insights/DiagnosticSettings/Write",
"microsoft.aadiam/diagnosticsettings/delete",
"microsoft.aadiam/diagnosticsettings/read",
"microsoft.aadiam/diagnosticsettings/write",
"Microsoft.EventHub/namespaces/eventHubs/authorizationRules/regenerateKeys/action",
"Microsoft.EventHub/namespaces/eventHubs/authorizationRules/listkeys/action",
"Microsoft.EventHub/namespaces/eventHubs/authorizationRules/delete",
"Microsoft.EventHub/namespaces/eventHubs/authorizationRules/write",
"Microsoft.EventHub/namespaces/eventHubs/authorizationRules/read",
"Microsoft.Authorization/roleAssignments/delete",
"Microsoft.Authorization/roleAssignments/write",
"Microsoft.Authorization/roleAssignments/read",
"microsoft.web/sites/slots/config/delete",
"Microsoft.Web/sites/slots/config/Write",
"Microsoft.Web/sites/slots/config/Read",
"microsoft.web/sites/functions/properties/read",
"microsoft.web/sites/functions/properties/write",
"microsoft.web/sites/extensions/write",

```
"microsoft.web/sites/extensions/read",  
"microsoft.web/sites/extensions/delete",  
"Microsoft.Web/sites/slots/config/list/Action",  
"Microsoft.Web/sites/config/list/Action",  
"Microsoft.Insights/LogProfiles/Read",  
"Microsoft.Insights/LogProfiles/Delete",  
"Microsoft.Insights/LogProfiles/Write"
```

To set user permissions:

1. Log in to Microsoft Azure as a user with administrator privileges.
2. To add users for this connector:
 - a. Click Azure Active Directory Users > All Users > New User.
 - b. Update user details.
 - c. Select Directory Role
 - d. Assign one of the following roles: Global administrator or Security administrator.
3. Click All services > Subscriptions and select the subscription you want for this connector.
4. Set the user you created in Step 2 as the **Owner or Contributor or Custom role** of the selected subscription:
 - a. Click Access Control (IAM) > Add.
 - b. Add the user who will deploy the connector and assign the role **Owner or Contributor or Custom role**.
 - c. (Conditional) Each connector supports a single subscription. If you have multiple subscriptions and you want to monitor the services under all your subscriptions, repeat the procedure in all of them. Multiple subscriptions require the use of multiple connectors.

Opening Ports

You must ensure that the ports on the server on which you installed the Syslog NG Daemon SmartConnector is accessible from Azure. The procedure to open ports varies based on whether you have installed Syslog NG Daemon SmartConnector on a virtual machine or not.

Opening Ports on a Non-Virtual Machine

If you installed Syslog NG Daemon SmartConnector on a physical, non-virtual machine, ensure that the ports on which you installed it are accessible to Azure.

Opening Ports on a Virtual Machine

If you have installed the Syslog NG Daemon SmartConnector on a virtual machine in Azure cloud, ensure that the ports on which you installed Syslog NG Daemon SmartConnector are open in both Azure and the virtual machine.

To open inbound ports on Azure:

1. Log in to Microsoft Azure as a user with administrator privileges.
2. Click Virtual Machines > Virtual machine name > Networking > Add inbound port.
3. Enter the same port or port range on which you installed the Syslog NG Daemon SmartConnector.
4. Update other fields and click Add.

To open ports in the virtual server:

1. Log in to the virtual Microsoft Windows Server machine.
2. Open Microsoft Windows Server Firewall.
3. Click Inbound Rules > New Rule > Port > Next > TCP > Specific local ports
4. Enter the same port or port range on which you installed the Syslog NG Daemon SmartConnector.
5. Click Next > Allow the connection > Next > Profile > Next.
6. Name the rule,
7. Click Finish.

Enabling Windows PowerShell to Run Scripts

To deploy the connector, you must run a script in Windows PowerShell. Ensure that Windows Powershell is enabled to run scripts on the machine where you want to deploy the connector. This procedure needs to be done only once on the machine.

PowerShell scripts are now signed in Azure Event Hub SmartConnectors. This allows users to run them in security-enabled environments with an execution policy set to either RemoteSigned or AllSigned. For more information, see [PowerShell Execution Policies](#).

Note: Signed scripts can still run in unrestricted environments.

To enable Windows PowerShell to run scripts:

1. Upgrade the Windows PowerShell version to 5.0 or later.
2. Click Start and search for Windows PowerShell. Right-click Windows PowerShell and click Run as administrator.
3. Check the current script execution policy:

```
Get-ExecutionPolicy
```

4. If the current script execution policy is Restricted, change the script execution policy to Unrestricted

```
Set-ExecutionPolicy unrestricted
```

Enter Yes to All when prompted.

5. Run the Get-ExecutionPolicy command to ensure that PowerShell is now Unrestricted.
6. Install the Azure Resource Manager modules from the PowerShell Gallery:

```
Install-Module -Name AzureRM -AllowClobber
```


For more information about installing, see the Azure documentation.
7. When prompted to install NuGet, enter Yes.
8. When prompted to install modules from the untrusted sites (PowerShell gallery), enter Yes to All.

You should now be able to run scripts in PowerShell.

Deploying the Connector

Deploying the connector will automatically deploy and configure the required components in your Azure Cloud.

When you deploy the connector against a subscription, you can monitor the events emitted from the services registered to the subscription. If you have multiple subscriptions and you want to monitor the services under all your subscriptions, you must deploy this connector against each of the subscriptions separately. The connector gets deployed directly into the Azure cloud and you do not need to set up a virtual machine in the cloud to deploy the connector.

To deploy the connector:

1. On the machine from where you want to deploy the connector, download arcsight-azure-monitor-eventhub-connector-8.0.0.zip.
2. Extract the arcsight-azure-monitor-eventhub-connector-8.0.0.zip files to the desired location.
3. Configure the application properties of this connector:
 - a. Edit the app.properties file.
 - b. (Conditional) Modify the value of FunctionAppName1 and FunctionAppName2 to change the name of the Function Apps. The default names of the Function Apps are arcsight-cloudfunctions and arcsight-monitor-functions.
 - c. Specify the connectorhostname by entering the IP address or hostname of the Syslog NG Daemon SmartConnector or the Load Balancer.

- d. Specify the `connectorport` of the Syslog NG Daemon SmartConnector or the Load Balancer.
- e. Update the `keyStoreFileName` and `keyStorePassword` parameters with the keystore file name and password in the Syslog NG Daemon SmartConnector or Load Balancer. The `keyStoreFileName` and `keyStorePassword` are used by the event hub connector application running on Azure to establish a TLS connection over SSL with the client Syslog NG Daemon SmartConnector.

Note: Copy the keystore file from Syslog NG Daemon SmartConnector or the Load Balancer to the cloud. To access the keystore file, log in to Azure and click Storage Accounts > *<storage account name>* > Files > Storage container > *<function app name>* > `certs` folder.

After you deploy the connector, upload your keystore certificate. For more information about uploading your certificate, see “Customizing the Connector”.

Caution: : Do not use the existing resource groups in your Azure environment because this resource group will be deleted when you uninstall this connector.

- f. Specify a unique `storageaccountname`.
- g. Specify a unique `Eventhubnamespace`.
- h. Specify the `Service Plan`. You can specify either `Consumptionplan` or `Appserviceplan`.
 - A Consumption plan is a serverless plan and allows you to scale automatically.
 - An App Service plan handles a fixed event load.

Specify the following only if you are entering `Appserviceplan`:

`servicePlanName` as `ArcSightPlan`

`servicePlanTier`

`servicePlanNumberOfWorkers`

`servicePlanWorkerSize`

For more information about specifying the `Service Plan`, see Azure documentation.

- i. Specify the location based on the locale of the resources you want to monitor.
- j. Save the file.

Note: Back up the `app.properties` file because you would need to refer to these configurations during uninstallation.

4. Open Windows PowerShell as Administrator and run the following command:

<extracted path>\DeployFunction.ps1

5. When prompted, log in to Microsoft Azure as a user with required privileges for the subscription you want to use with Azure
6. When prompted, select the required subscription.
7. Verify the deployment as described in “Verifying the Deployment in Azure”.
8. Configure the Syslog NG Daemon SmartConnector or Load Balancer with an ArcSight destination to view and monitor events coming from Azure Cloud Services. For more information, see the ArcSight Syslog NG Daemon SmartConnector Load Balancer documentation.
9. Complete the procedures listed in “Post-Deployment Configurations”. For more information, see the ArcSight Syslog NG Daemon SmartConnector or Load Balancer documentation.
10. (Conditional) Configure the Load Balancer.

Verifying the Deployment in Azure

Successful deployment of this connector does the following:

1. Installs two Azure functions in your Azure suscription: <arcsight cloud function app name> and <arcsight monitor app name>. These functions are used to collect events from Azure event hubs and monitor the health of the connection downstream. To view the functions in Azure, click Function Apps.
2. The install script automatically streams events from the audit logs, sign-in logs, and activity logs. For diagnostic logs, you must manually add diagnostic settings to configure streaming of these logs. For more information, see Step 7 in “Streaming Diagnostic Logs”.
3. Uploads the application settings listed in the app.properties file to Azure. This enables you to add or modify properties from Azure instead of modifying the app.properties file and redeploying the Connector. To view the properties in Azure, click Function Apps > <function app name> > Application Settings. For more information about modifying these properties, see “Customizing the Connector”.
4. Uploads the map files to Azure and the Azure Monitor Event Hub Connector can convert JSON events to a CEF format. To view the certificates in Azure, click Storage Accounts > <Storage account name> > Files > Storage container > <function app name> > certs folder. The default name of the storage account is emitterarcsightstorage.
5. Uploads the map files to Azure convert JSON events to CEF. To view these map files, click Storage Accounts > Storage account name> > Files > Storage container > function app name> maps folder.
6. Creates an Active Directory application called <arcsight monitor app name> and assigns the Azure Website Contributor role to this application

7. Creates a resource group called <arcsight functions group name>. This resource group manages the resources of this connector. The default name of the resource group is "arcsightfunctions-group".
8. Creates an Azure storage account called <storage account name>. This storage account stores the connector certificate, function logs, and the parser files.

Post-Deployment Configurations

You must configure the connector after deployment. To configure, you must log in to Azure as a user with required privileges for the subscription you want to use with Azure and perform the following:

["Additional Configurations for App Service Plan" below](#)

["Streaming Diagnostic Logs" below](#)

["Updating Keystore Certificate" on the next page](#)

Additional Configurations for App Service Plan

On the App Service Plan, function apps are designed to go to an idle state after a default timeout period. Therefore, you must manually configure the function apps to stay connected even if events are not streamed during the timeout period.

To configure the function apps to stay connected:

1. Click **Function Apps**<Function Name> **Application Settings**< **General Settings**.
2. Update to: **"Always On"**.

Ensure that you do this for both <arcsight cloud function app name> and <arcsight monitor function app name>.

Streaming Diagnostic Logs

The install script automatically streams events from the audit logs, sign-in logs, and activity logs. For diagnostic logs, you must manually add diagnostic settings to configure streaming of these logs. For information about adding diagnostic settings, see Azure documentation. While you are adding diagnostic settings, perform the following:

To stream diagnostic logs:

1. Select **Azure Home > Monitor > Diagnostic Settings**.
2. Select the event hub (the default event hub name is: eh-emitter-arcsight).
3. When the list of configured diagnostics is displayed, click **Edit** on the desired

diagnostic to be updated.

- a. Click Add, to monitor a new resource.
4. From the Diagnostic settings window, mark the Stream to an event hub check box (if not marked) or select the event hub.
5. On the Select event hub window:
 - a. From the the Select event hub namespace drop-down list, specify <name of event hub namespace>.
 - b. From the Select event hub name drop-down list, select insights-diagnostics-logs.
 - c. From the Select event hub policy name drop-down list, select ArcSightAccessKey.
6. Click OK.
7. On the Diagnostic settings window, select the logs you want to stream.

Updating Keystore Certificate

The Syslog NG Daemon SmartConnector includes a default keystore. During deployment, this default keystore is associated with a newly created storage account. You must associate your keystore with the new storage account to prevent errors.

To update keystore certificate:

1. Rename the desired keystore certificate as remote_management.p12, which is the file name of the default keystore certificate so that the Azure connector identifies the custom keystore.
2. Log in to Microsoft Azure as a user with required privileges for the subscription you want to use with Azure.
3. Select All Services > Storage Accounts.
4. In the Storage Accounts window, select <storage account name>.
5. In the Services tab, select Files.
6. Select the <storage container name> function.
7. Select the displayed folder.
8. In the fileshare of the storage account, select the certs folder. The default certificate, remote_management.p12, is displayed.
9. Delete this default certificate or overwrite the existing file.
10. Upload the certificate, remote_management.12.
 - a. Click Upload.
 - b. In the URL field, browse to the desired location and select remote_management.p12.
 - c. Click Upload.

11. Restart both the function apps:
 - a. Click Function Apps > <arcsight cloud function app name> > Restart.
 - b. Click Function Apps > <arcsight monitor app name> > Restart.

Note: After restarting the function apps, the Azure Connector is restarted along with the certificate uploaded.

Customizing the Connector

You can customize the connector properties as required.

To customize the connector:

1. Log in to Microsoft Azure as a user with required privileges for the subscription you want to use with Azure.
2. Click Function Apps > <arcsight cloud function app name> or <arcsight monitor function app name> > Platform Features > Application Settings.

Important: Do not modify any of the settings other than those listed in this procedure as this may cause unpredictable performance or even outages.

3. (Conditional) To modify the connector port and connector name of the Syslog NG Daemon SmartConnector or Load Balancer:
 - a. Update the Connector Port field.
 - b. Update the Connector Hostname field with the IP address or hostname.

Note: Ensure that you do this for both <arcsight cloud function app name> and <arcsight monitor function app name>.

4. (Conditional) Set the logging level.

Select <arcsight cloud function app name> or <arcsight monitor function app name> as mentioned in Step 2. By default, the logging level is INFO. To increase the logging level, enter DEBUG or TRACE in the Logging level field.
5. (Conditional) You can send the connector logs to a storage account. However, this consumes cloud storage

In the logging.storage.enabled field, enter true. The connector now sends logs from the function app to the storage account every 15 minutes.

To stop sending logs to the storage account, enter false in the logging.storage.enabled field.
6. Click Save.

Performance Scaling

You might need to modify your deployment or change certain configuration to improve the performance.

Your Azure pricing plan also affects performance scaling. A Consumption plan scales automatically and an App Service plan handles a fixed event load. A Consumption plan automatically creates Function App instances to scale up the load. For more information about the event load handled in a App Service plan, see the [Azure Documentation](#).

Configuring Load Balancer

In environments where the event load is more than what can be handled by a single Syslog NG Daemon SmartConnector, you can configure Load Balancer to handle large event loads. For more information about configuring Load Balancer, see [ArcSight Load Balancer documentation](#).

Upgrading the Connector

You can only do a binary upgrade of the connector. A binary upgrade of the connector upgrades the connector and also enables you to continue using the components created during deployment. You will not lose any custom settings.

To upgrade the connector:

1. On the machine from where you want to upgrade the connector, download `arcsight-azure-monitor-eventhub-connector-8.0.0.zip`.
2. Extract the `arcsight-azure-monitor-eventhub-connector-8.0.0.zip` files to the desired location.
3. Configure the `app.properties` file. For more information, see Step 3 in “Deploying the Connector”. Ensure that you specify the same Function App names that you specified during deployment.
4. Open Windows PowerShell as Administrator and run the following command:
`<extracted path>\DeployFunction.ps1`
5. When prompted, log in to Microsoft Azure as a user with required privileges for the subscription you want to use with Azure.
6. When prompted, select the required subscription.
7. When the script prompts you, select one of the following:
 - Y to do a binary upgrade.
The script first checks whether there is an existing installation of the connector in the cloud and then upgrades only the binary files without affecting the configuration settings.
 - N to do a fresh deployment of the connector.
The script deploys the connector and overrides all custom settings with default settings.

Undeploying the Connector

Undeploying the connector deletes the Active Directory application created during deployment, deletes the resource group, and all the associated components such as storage account and event hubs created during deployment.

To undeploy the connector:

1. Open Windows PowerShell as Administrator and run the following command:
`<deployed path>\UndeployFunction.ps1`

2. When prompted, log in to Microsoft Azure as a user with required privileges for the subscription you want to use with Azure.
3. When prompted, select the required subscription.
The script displays the components that will be removed and prompts you for permission to undeploy the Azure connector.
4. Enter Yes when prompted.
The script undeploys the Azure connector.

Appendix A: Azure Event Log Categories

Azure event logs such as activity logs and diagnostic logs are emitted in JSON format. The connector collects these event logs, converts these to CEF using mapping files, and sends these to Syslog NG Daemon SmartConnector or Load Balancer. Every JSON field is mapped to the appropriate CEF key. Each event log type has various categories and each log category has its own schema. Azure logs have schema for various log categories. With the help of these logs schema, the source fields (in JSON) are mapped to appropriate CEF keys.

The connector currently includes mapping files for several log categories of activity, audit, sign-in, and diagnostic logs. The Azure documents do not have the schemas for a few categories. Therefore, the mappings for these categories are not available in the connector. Such events are sent unparsed to the Syslog NG Daemon SmartConnector or to the Load Balancer, and then forwarded to the ArcSight destination.

Following tables list the categories for mappings supported by the Azure connector. The mappings are done using the schemas provided in the Azure documents.

This destination sends events in Common Event Format (CEF) through a Kafka broker to Microsoft Azure Event Hub (the Event Hub must enable a Kafka endpoint). For more information, see the *SmartConnector User's* guide.

Creating a Map File

About

In map files, attributes are used to map CEF fields. There are three types of logs:

- Active Directory
- Activity
- Diagnostic

Azure logs display the content as attribute-value pairs, separated by a comma. Attributes and values are separated by a colon.

For example:

```
"<attribute1>":"<value1>", "<attribute2>":"<value2>"
```

Procedure

To create a Map File:

1. Name your map file, following the format below:
`<log type>_<value of attribute "category" in the log file>.map`
2. Map the mandatory fields: "header.EventName", "header.DECID", "header.Severity".

The format is [category=<log type>:<attribute category>]

For example:

header.DECID, operationName

header.Severity, Level

header.EventName, operationName

3. Map the JSON Event fields with the corresponding CEF field. For more information about CEF fields, see the *Implementing ArcSight Common Event Format (CEF) guide*.

The format is JSON Event Field, CEF Field.

For example:

category, cat

properties/id, cs2Label

Note: JSON Event fields may contain a backslash or not. The attribute flags on the example above, are the main attributes, therefore, mapped without backslash. Sub attributes must be back slashed.

An Azure Event Log looks like this:

```
{
  "records": [
    {
      "time": "2018-10-17T13:35:53.2803858Z",
      "resourceId": "/tenants/5178620e-2208-456f-a21a-41df534180fe/providers/Microsoft.aadiam",
      "operationName": "Delete user",
      "operationVersion": "1.0",
      "category": "AuditLogs",
      "tenantId": "5178620e-2208-456f-a21a-41df534180fe",
      "resultType": "Success",
      "resultSignature": "None",
      "durationMs": 0,
      "callerIpAddress": "<null>",
      "correlationId": "15235063-7ec4-4a09-bade-17e595251300",
      "level": "Informational",
      "properties": {
        "id": "Directory_UFHU8_51444601",
        "category": "Core Directory"
      }
    }
  ]
}
```

4. Map the CEF fields.

Note: Fields not included on the list above, should be added as "Additional data".

Updating Map Files

1. Upload the new map files to Azure.
2. To view these map files:
 - a. Go to **Storage Accounts < Storage Account Name < Files < Storage Container < Function App Name < Maps.**

Diagnostic, Audit, and Sign-In Log Categories

Table A-1 Diagnostic Log Categories

Categories	Resource Type	Certified
GatewayLogs	Microsoft.ApiManagement/service	Yes
JobLogs	Microsoft.Automation/automationAccounts JobStreams	No
JobStreams	Microsoft.Automation/automationAccount	No
CoreAnalytics	Microsoft.Cdn/profiles/endpoints	No
PipelineRuns	Microsoft.DataFactory/factories	No
TriggerRuns	Microsoft.DataFactory/factories	No
Audit	Microsoft.DataLakeAnalytics/accounts	No
Requests	Microsoft.DataLakeAnalytics/accounts	No
Audit	Microsoft.DataLakeStore/accounts	No
Requests	Microsoft.DataLakeStore/accounts	No
Connections	Microsoft.Devices/IotHubs	No
DeviceTelemetry	Microsoft.Devices/IotHubs	No
C2DCommands	Microsoft.Devices/IotHubs	No
DeviceIdentityOperations	Microsoft.Devices/IotHubs	No
FileUploadOperations	Microsoft.Devices/IotHubs	No
Routes	Microsoft.Devices/IotHubs	No
D2CTwinOperations	Microsoft.Devices/IotHubs	No

Categories	Resource Type	Certified
C2DTwinOperations	Microsoft.Devices/IotHubs	No
TwinQueries	Microsoft.Devices/IotHubs	No
JobsOperations	Microsoft.Devices/IotHubs	No
DirectMethods	Microsoft.Devices/IotHubs	No
DataPlaneRequests	Microsoft.DocumentDB/databaseAccounts	No
ArchiveLogs	Microsoft.EventHub/namespaces	No
OperationalLogs	Microsoft.EventHub/namespaces	No
AuditEvent	Microsoft.KeyVault/vaults	No
WorkflowRuntime	Microsoft.Logic/workflows	No
NetworkSecurityGroupEvent	Microsoft.Network/networksecuritygroups	Yes
NetworkSecurityGroupRuleCounter	Microsoft.Network/networksecuritygroups	Yes
LoadBalancerAlertEvent	Microsoft.Network/loadBalancers	No
LoadBalancerProbeHealthStatus	Microsoft.Network/loadBalancers	No
ApplicationGatewayAccessLog	Microsoft.Network/applicationGateways	No
ApplicationGatewayPerformanceLog	Microsoft.Network/applicationGateways	No
ApplicationGatewayFirewallLog	Microsoft.Network/applicationGateways	No
OperationalLogs	Microsoft.ServiceBus/namespaces	No
QueryStoreRuntimeStatistics	Microsoft.Sql/servers/databases	No
QueryStoreWaitStatistics	Microsoft.Sql/servers/databases	No
Errors	Microsoft.Sql/servers/databases	No
DatabaseWaitStatistics	Microsoft.Sql/servers/databases	No
Timeouts	Microsoft.Sql/servers/databases	No
Blocks	Microsoft.Sql/servers/databases	No
Audit	Microsoft.Sql/servers/databases	No
Execution	Microsoft.StreamAnalytics/streamingjobs	No
Authoring	Microsoft.StreamAnalytics/streamingjobs	No
AzureFirewallApplicationRule	Microsoft.Network/AzureFirewalls	Yes
AzureFirewallNetworkRule	Microsoft.Network/AzureFirewalls	Yes

Activity Log Categories

Table A-2 Activity Log Categories

Categories	Resource Type	Certified
Action		Yes
Administrative		Yes
Alert		Yes
Delete		Yes
Recommendation		Yes
Security		No
ServiceHealth		Yes
Write		Yes

Active Directory Log Categories

Table A-3 Active Directory Log Categories

Categories	Resource Type	Certified
Signin		Yes
Audit		Yes

Appendix B: Troubleshooting

Errors during Deployment

If you receive an error message prompting you to register the subscription <subscription id> with Microsoft.Insights, register the microsoft.insights provider.

To register the resource provider:

1. Log in to Microsoft Azure as a user with administrator privileges.
2. Click **All Services < Subscriptions**.
3. Select the subscription you want for this connector.
4. Select **Resource Providers**.
5. Click **Register**.

Connecting Errors

Connection errors are displayed when:

The Syslog NG Daemon SmartConnector hostname and port are not reachable from Azure cloud. Ensure that the Syslog NG Daemon SmartConnector host and port are reachable from Azure cloud. Open the relevant ports. The certificate file is overridden during the deployment of the connector. Replace the remote connection management file in Azure with your remote connection management file. Click Storage Accounts > <Storage account name> > Files > Storage container > <function app name> > certs folder. Replace the remote_management.p12 file with your <customname>.p12 file.

Parsing Errors

Parsing errors are displayed if the event log categories are not supported by the connector. For a list of the supported categories, see Appendix A, "Azure Event Log Categories".

You can contact technical support in the following scenarios:

- If you want to change the default mappings.
- If you want to add a new log type.
- See parsing errors.

Sharing Logs for Troubleshooting

You may want to share logs with technical support for troubleshooting.

To share logs:

1. Log in to Microsoft Azure as a user with administrator privileges.
2. Set the logging level to **TRACE** or **DEBUG**. For more information, see Step 4 in ["Customizing the Connector" on page 21](#).
3. From the **Development Tools** menu, click **App Service Editor**.
4. Click **Go**.
5. On the new App Service Editor tab, select **Open Kudo Console** from the top drop down menu.
6. On the new tab, go to: **site > wwwroot > logs**.
7. Click download and send the logs to technical support.

Using a Private IP

You must upgrade to a Standard or Premium plan.

1. Add a new or an existing VNet to the following resources, created after deployment.
Event Hubs, Storage Account and Function Apps.

Event Hubs and/or Storage Account.

- a. From your **Event Hub** or **Storage Account**, click **Firewalls and Virtual Networks**
- b. Select **All Networks** or **Selected Networks** (in case you want limited networks to access your resource).
- c. Add the **Existing Virtual Network** or **Create New Virtual Network**.

Function Apps

- a. From **2 Function Apps < Networking, < VNet Integration**, add those to your VNet.
 - b. Add or remove network interfaces from your virtual machines, for more information, see [Add network interfaces to or remove network interfaces from virtual machines](#).
2. Enable the Service endpoints of the previously used subnets.

- a. From **Virtual networks Service**, select your **VNet < Subnets**.
 - b. Open all the subnets.
 - c. Select **All Service Endpoints** and save your changes.
3. Check if the Function Apps communicate to the destination (ArcSight Syslog NG Daemon, ArcSight Load Balancer, etc.) through the Private IP.
4. From **Development Tools < Console Tool**, execute the `tcpping` command to your VM via private IP.


```
tcpping host:port
host: private IP
port: you may use port 3389 or the port used in your Function Apps.
```
5. After successfully executing the command above, from **Function Apps < Application Settings**, check if the setting already exists or add a new one:


```
APP SETTING NAME: JAVA_OPTS
VALUE: -Djava.net.preferIPv4Stack=true
```
6. In the field **connectorhostname**, enter your Private IP.
7. Next, in the field **Port**, enter the port of your Private IP.
8. Restart **Function Apps**.

Note: The VNet integration preview is a preview, if it does not work, you can disable and enable the VNet integration or create another subnet.

Azure Datacenters with Stamps (Scale Units) without Premium V2 VMs

Only applications running on stamps that support Premium V2 scale units, possess the hardware required to use the VNET Integration (preview) feature.

AppService Plan with Basic Pricing Tier Created on a Stamp That Does not Support Premium V2 VMs

It is possible that your previous **AppService plan** (even for a basic tier) was created in a stamp that does support Premium V2, hence, it can use the VNET Integration feature. If the new version of your application was created in an AppService plan with a stamp that does not have Premium V2 VMs, you might not be able to see the VNET Integration feature.

Workaround:

1. Ensure your **AppService plan** was created in a stamp that supports VNET Integration. It is recommended to initially create the AppService Plan as Premium V2 and once created, scale it back down to Standard and then, use the VNET Integration feature.
2. If your **AppService plan** does not show the feature to scale up to Premium V2, you might not be able to create a new AppService plan in the same Resource Group of the Premium V2 pricing Tier. This happens because the **Resource Group** sometimes, decides on a particular stamp and creates all resources in there. If you experience this issue, try creating the AppService plan in a different Resource Group

Send Documentation Feedback

If you have comments about this document, you can [contact the documentation team](#) by email. If an email client is configured on this computer, click the link above and an email window opens with the following information in the subject line:

Feedback on Configuration Guide (SmartConnectors 8.0.0)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to Documentation-Feedback@microfocus.com.

We appreciate your feedback!